SeeR: Simulated Annealing-Based Routing in Opportunistic Mobile Networks

Barun Kumar Saha, Sudip Misra, *Senior Member, IEEE*, and Sujata Pal, *Member, IEEE*

Abstract—Opportunistic Mobile Networks (OMNs) are characterized by intermittent connectivity among nodes. In many scenarios, the nodes attempt at local decision making based on greedy approaches, which can result in getting trapped at local optimum. Moreover, for efficient routing, the nodes often collect and exchange a lot of information about others. To alleviate such issues, we present SeeR, a simulated annealing-based routing protocol for OMNs. In SeeR, each message is associated with a cost function, which is evaluated by considering its current hop-count and the average aggregated inter-contact time of the node. A node replicates a message to another node, when the latter offers a lower cost. Otherwise, the message is replicated with decreasing probability. Moreover, SeeR works based solely upon local observations. In particular, a node does not track information about other nodes, and, therefore, reduces the risk of privacy leaks unlike many other protocols. We evaluated the performance of SeeR by considering several real-life traces under plausible conditions. Experimental results show that, in the best case, SeeR can reduce the average message delivery latency by about 58 percent, when compared to other popular routing protocols.

Index Terms—Opportunistic mobile networks, routing, simulated annealing, privacy

1 Introduction

In OMNs, nodes have intermittent connectivity among themselves, which makes routing a challenging problem. Although a plethora of schemes are proposed for efficient message delivery in OMNs, the quest for the *holy grail* of routing solution still continues. In general, an optimal routing protocol should result in higher message delivery ratio, lower latency, and lower routing overhead. These performance metrics are interlinked, and obtaining the "best" tradeoff is difficult. Some contemporary routing protocols, for example, PRoPHET [1], offer high delivery at the expense of high overhead. Other protocols, such as Delegation Forwarding (DF) [2], result in low overhead but high delivery latency. Moreover, in reality, the nodes in OMNs often attempt at decision making based on locally available information using greedy heuristics, which are prone to getting trapped at local optimum. Motivated by this, in this work, we present a Simulated annealing-based Routing (SeeR) protocol for OMNs. In SeeR, our objective is to reduce the delivery latency, while maintaining good chances of message delivery, as the other state-of-the-art routing protocols. It may be noted here that from the network point of view, the percentage of messages delivered is perhaps the most important performance indicator. However, one should remember that routing services in OMNs would eventually be used by human users, who are sensitive to latencies [3], [4], [5]. Therefore, it is desirable that one looks for a routing scheme that minimizes message delivery latency while not incurring significant deterioration in message delivery percentage at the same time.

Many contemporary routing protocols, on the other hand, utilize different types of information (metadata) about the other nodes in order to improve efficiency. Storage of such destination-specific utilities is undesirable because it can lead to privacy leaks. For example, it might be possible to identify a set of users that the device of a given user frequently comes in contact with. SeeR reduces such privacy risks by relying solely upon the local observations available at nodes as well as by not exchanging such information with others.

Conceptually, the simulated annealing (SA) [6] framework–with its *cost*, *solution*, and *neighbor* components–is well-suited for modeling routing problems in OMNs as compared to other related approaches, for example, genetic algorithm (GA). Moreover, it is not only easy to formulate a problem with SA, but it has also been shown that SA-based approaches can outperform those based on GA [7], [8]. These factors motivate us to use SA in designing SeeR.

Fig. 1 illustrates the high-level design of SeeR. The "contact engine" of a node maintains an estimate of its average inter-contact time[1] (ICT). By combining a node's ICT, together with the hop-count of a message, the "routing engine" assigns a cost metric to each message stored in its buffer. A routing decision is taken by comparing this composite cost of a message with the corresponding cost offered by another node. It may be noted that not only these information are locally observable, but there is also no (or rather, constant) overhead in exchanging such information among the nodes in the OMN.

• B.K. Saha and S. Misra are with the Department of Computer Science and Engineering, Indian Institute of Technology Kharagpur, Kharagpur, West Bengal 721302, India. E-mail: {barun.kumar.saha, smisra}@sit.iitkgp.ernet.in.
• S. Pal is with the Department of Computer Science and Engineering, Indian Institute of Technology Ropar, Rupnagar, Punjab 140001, India. E-mail: sujata@iitrpr.ac.in.

Manuscript received 11 Sept. 2015; revised 8 Feb. 2017; accepted 13 Feb. 2017. Date of publication 23 Feb. 2017; date of current version 29 Aug. 2017.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TMC.2017.2673842

1. The mean contact time between a given node and any other node in the network. This is discussed in details in the latter Sections.

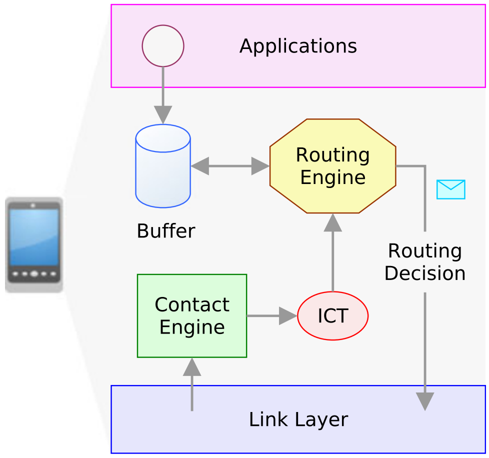


Fig. 1. Overview of the architecture of SeeR.

1.1 Contributions

The specific *contributions* of this work are summarized below.

- Identifying the minimum latency routing problem in OMNs and investigating the difficulty in solving it.
- Proposing SeeR, a SA-based routing scheme for OMNs, that works based solely on local observations, reduces average message delivery latency, and minimizes the risk of privacy leaks in terms of social contact profiling.

1.2 Organization

The remainder of this work is organized as follows. Section 2 reviews the state-of-the-art routing protocols, as well as privacy schemes, and highlights the scopes for improvements. Section 3 presents the minimum latency routing problem. Section 4 begins with a brief overview of SA followed by which the design of SeeR is discussed. In Section 5, we talk about the risks that storage of routing metadata can pose, and discuss how SeeR overcomes it. Simulation setup, benchmarks, and performance evaluation metrics are discussed in Section 6. The results of performance evaluation and related discussions are presented in Section 7. Finally, Section 8 concludes this work.

2 Related Work

Whether one is looking for mobility-assisted [9] or cooperation-based [10] message delivery in OMNs, having an efficient underlying routing scheme is a necessity. In the following, we focus only on multi-copy based message replication protocols, which is the *de facto* standard practice in OMNs, and do not consider single-copy based forwarding schemes (such as [11]).

In the absence of information about the global network performance or configuration, routing schemes for OMNs usually rely upon information available locally, as well as from neighbors, for their operation. The fundamental decision making problem concerned with routing in OMNs is, whether or not a message should be replicated to a given node. For example, Lindgren et al. [12] proposed PRoPHET, which uses the time-varying likelihood of meeting a node, referred to as delivery predictability, as the basis for the aforementioned decision making. Moreover, transitivity in contact patterns among the nodes is exploited. In PRoPHET, a node replicates a given message to a certain node only if the latter has better chances of meeting with the destination of the message. Pal and Misra [13] proposed Contact-based Routing (CBR), where, in the presence of multiple neighboring nodes, a transmitting node sends a copy of a message only to the neighbor that has the maximum frequency of encounter with the corresponding destination node. It was shown that CBR can reduce the average time that a message stays in a node's buffer. Apart from PRoPHET and CBR, there are large number of routing protocols that adopt greedy approaches for optimal decision making based on locally available information. Consequently, they suffer from a common risk–getting trapped at local optimum. For example, in PRoPHET, a transmitting node does not replicate a message to a node, if the delivery predictability of the latter is less than the former. However, in practice, it is quite possible that the ignored node can have a better delivery predictability for the destination node in the future, and therefore, the unperformed replication to that node could rather have been a move toward optimality. In general, these protocols often take suboptimal actions in their attempt to exploit contemporary local information. As a result, they often result in high delivery latency and overhead.

The DF protocol, proposed by Erramilli et al. [2], is of particular interest where message- and network-level properties are fused together. In particular, in DF, every message has a threshold, and every node has a routing quality metric. A node replicates a message to another node only if the quality of the latter is greater than the threshold of the message. Subsequently, the threshold is updated to the new quality metric. A direct consequence of such optimal decision making is a significant reduction in overhead. However, DF, in general, suffers from high delivery latencies.

Abdelkader et al. [14] noted that the minimization of hop-count resulted in relatively higher delivery ratio of messages than the optimization of other metrics. The social group-based routing (SGBR) [15] protocol was proposed accordingly. However, in practice, we have observed that the average hop-count along a message delivery path crosses four in relatively few cases. In fact, a minimum of two hops is required, unless direct delivery of a message by its source to the destination node is considered. As such, the aggregated benefit of minimizing hop-count may not be much, and therefore, we aim at minimizing the average latency instead.

Uddin et al. [16] proposed an energy-efficient routing scheme targeting OMNs formed in post-disaster scenarios. Based on encounters among the nodes, and the encounter graph so induced, the proposed routing scheme constructs an inter-contact graph. Variation in latencies is used to compute the delivery probability of a message by a given node. Accordingly, proportional number of message copies is replicated to the concerned node. However, as the authors themselves noted, approximating inter-contact latencies with a normal distribution requires large number of samples. Its direct impact is upon the storage requirement of the nodes. Moreover, Uddin et al. evaluated their scheme only with a synthetic post-disaster mobility model. In the recent years, however, OMNs have found applications in various other scenarios, such as cellular traffic offloading [17]. Such usage in urban contexts, with provisions for recharging devices, lacks in stringent energy constraints, but requires higher delivery ratio and lower latency.

Fan et al. [18], on the other hand, aimed to minimize the variance in the proportion of successful delivery of messages for all the source nodes–in contrast to network-wide aggregate message delivery, which is typically considered. To address the underlying unfairness, the proposed scheme assigns a priority value to each message generated by a node. The priority of a message is set to be inversely proportional to the delivery success rate of the corresponding node. The proposed scheme was found to improve fairness and reduce cost of routing. However, the effect upon message delivery latency remained unexplored. Moreover, the proposed scheme is not a routing protocol by itself, but rather a "plug-in" module to improve the fairness and reduce cost of a routing protocol. As such, certain drawbacks of the host routing protocol, as we note below, may not be mitigated by the scheme.

It may be noted that, PRoPHET [1], CBR [13], DF [2], and many other existing routing schemes, make use of per-node utilities. In particular, a utility value for each node in the OMN is maintained, based on which routing decisions are made. Moreover, such metadata are exchanged among the nodes. Apart from the storage and transmission overhead required for them, which, however, are typically ignored, there is another potential issue involved here. To illustrate, let us consider that the delivery predictability vector of a node using PRoPHET is available to a rogue third party. Therefore, theoretically, it would be possible to infer the set of other nodes frequently visited by that node (whose information is available) in real life. Of course, gaining information by decoding such values would depend upon the kind of utility stored, storage format, and addressing schemes. Nevertheless, a risk of privacy leak does exist, in general. Such a scenario, however, is unlikely for routing protocols that do not use per-node utilities, for example, Spray-and-Wait [19].

Privacy-assured communication in OMNs has received considerable attention spanning different aspects, such as location privacy [20], anonymous communication [21], and obfuscating identity of source nodes [22]. A fundamental form of privacy breach is *users' social contact profiling*, where per-node routing metadata is used to infer social ties of users. This aspect largely lacks addressing, except perhaps in [23], where the authors proposed randomizing contact graphs as a solution. However, Distl and Hossmann [23] noted the lack of scalability of their approach. Such a risk can, however, be neutralized if node-specific utilities are not maintained at the first place.

To synthesize, based on the survey of the existing state-of-the-art routing protocols, we find several broad areas of improvement.

1) Many routing protocols that exploit local information for optimal solution are prone to getting trapped in local optimum, thereby resulting in *suboptimal performance* of the concerned metrics. Avoiding such local optimum, wherever possible, is desirable.
2) Many routing protocols often tend to gather as much contextual information as possible, in order to provide better performance in terms of the number of messages delivered or other metrics under consideration. However, an open question in this context is–*can we do more with less information*?
3) Use and exchange of per-node utilities among the nodes can lead to *privacy risks*. In general, vulnerability to privacy leaks should be minimized to the extent possible.

To address these lacunae, in this work, we propose SeeR, that takes into consideration the above mentioned aspects. In the remainder of this work, we take a detailed look at SeeR.

3 The Routing Problem

In this section, we discuss the network model and related terms used subsequently in this work. We also formulate the minimum latency routing as an optimization problem and discuss its nonsolvability in practice.

3.1 Network Model and Assumptions

We represent an OMN with a set of $n+2$ nodes, $N = \{1, 2, \ldots, n, n+1, n+2\}$. We assume that these nodes can communicate only when they are within the transmission range of one another. In particular, we assume that no form of "global" network connectivity is available.

The set of messages created in the network is denoted by M. Let us consider that $m.src$, $m.dst$, and $m.ttl$, respectively, denote the source, destination, and time-to-live (TTL) of a message m, $\forall m \in M$. The TTL of a message is usually considered to be a large, but finite, value.

For the sake of completeness, we define the following two terms. A *contact* is defined as an event when two nodes in the OMN come within the transmission range of one another, and, therefore, are able to communicate. The time duration between the termination of a contact and beginning of a new contact between a given pair of nodes is termed as the *inter-contact time*. Let τ_i be the average ICT of any node $i \in N$. A discussion on computation of ICT values would be presented in Section 4.

3.2 Minimum Latency Routing

Let us consider that the nodes s and d, respectively, are the source and destination nodes of a single message m; $s, d \in N$. Let P be the set of all possible paths (a sequence of nodes) through which m can pass from s to d. For example, the message can be directly delivered to d, so that the path is $s||d$[2] or, via another intermediate relay node say, x, so that the path becomes $s||x||d$ or, via multiple relay nodes. Moreover, any permutation of the relay nodes is possible. Then, we have

$$|P| = \sum_{r=0}^{n} \binom{n}{r} r! = \sum_{r=0}^{n} n!/r!, \tag{1}$$

where $r!$ denotes the factorial of r. The summation in (1), referred to as sequence A00522,[3] is recursively expressed as $S(n) = nS(n-1) + 1$. Accordingly, the minimum latency routing problem can be expressed as

$$\begin{aligned} \underset{p \in P}{\text{minimize}} \quad & \ell_p = \sum_i \ell_{p_i, p_{i+1}} \\ \text{subject to} \quad & \sum_i \ell_{p_i, p_{i+1}} < m.ttl \\ & 0 \le i < |N|, \end{aligned} \tag{2}$$

2. Here, $||$ denotes the concatenation operator. By $x||y$ we indicate that the identifiers (addresses) of nodes x and y are concatenated together to form a path from x to y.

3. https://oeis.org/A000522

where p_i denotes the ith node of the path p, $\ell_{p_i,p_{i+1}}$ is the latency of the message to travel from ith node to the next, and ℓ_p denotes the delivery latency of the message along the path p. The constraint indicates that the delivery latency should be less than the TTL of the message. The objective function on considering multiple messages can be written in a similar manner.

The inter-node latencies in (2) can be substituted with the corresponding pairwise ICTs (in contrast to aggregated ICTs as considered in this work). However, the optimization problem is difficult to solve due to the fact that, in reality, accurate estimates of all the pairwise ICTs are unavailable. Moreover, even if the presence of a global observer is assumed–as considered in [15], the future ICT values are likely to be different from the current estimates. Furthermore, the exponential growth of the search space of all possible paths P makes the minimization practically infeasible. It may, however, be noted that with each hop, the reduced set of possible paths decreases. This is stated in Theorem 1.

Theorem 1. *Let P_i be the set of available message delivery paths after a message has traversed i nodes. Then, P_i is a strictly monotonically decreasing function of i, when loop-free message forwarding is considered.*

Proof. Let p_i be the path that a message takes to reach the ith node from the source. Let us consider that the message is next forwarded to node $j \in N$, so that j becomes the $(i+1)$th node along the path. We concatenate j to p_i so that $p_{i+1} = p_i || j$. Now, p_{i+1} was potentially available in P_i, but no more in P_{i+1}, since node j has already been appended to the contemporary path. So, $|P_{i+1}| = |P_i| - 1$. Moreover, since routing is loop-free, j cannot appear anymore in the message forwarding path in future, which reduces further candidates from the possible delivery path space. As a consequence, $|P_{i+1}| < |P_i|$. □

Theorem 1 essentially states that with every hop, the search space for message delivery paths decreases in size. In the following Section, we look at the proposed solution of the problem (2), using a SA-based approach. Since it is not possible to evaluate the original objective function, we approximate it with a cost function, based on locally available information at the nodes.

4 Simulated Annealing-Based Routing

In this section, we present a brief overview of SA, followed by a detailed discussion on the operations of SeeR. For the ease of readers, Table 1 summarizes the data structures maintained by SeeR at each node. The notations are suffixed with node identifiers to indicate node-specific values, wherever appropriate. Among these, SeeR requires exchange[4] of only $\hat{\tau}$ among the nodes (neighbors). These are discussed in details in the remainder of this section.

4. SeeR also involves exchange of summary vectors–a list of IDs of the messages contained by a node in its buffer–similar to many other routing schemes. When a node receives a summary vector from another node, it learns the messages that should not be replicated to the other. To summarize, ICT estimates and summary vectors constitute all data–apart from the actual messages–that are exchanged among the nodes using SeeR.

TABLE 1
Data Structures Maintained by SeeR

Data structure	Description
$\hat{\tau}$	Variable storing the estimated ICT of a node
$T_1, T_2, \ldots$	Variables storing time instants when the previous contact with nodes $1, 2, \ldots$ is terminated; required for computing $\hat{\tau}$
MLT = $\{(m_1, t_1), (m_2, t_2), \ldots\}$	Message Lookup Table; times indicate when a given record can be removed

4.1 Overview of Simulated Annealing

In algorithms with greedy heuristics, an agent (or node) takes decision, based on locally available information. Although such decisions are locally optimal, they cannot always be guaranteed to be globally optimal. Metaheuristic algorithms address this shortcoming by involving exploration and exploitation of the solution space.

SA [6] is a metaheuristic algorithm that accepts (*exploits*) locally optimal solutions. However, unlike a greedy approach, it also accepts inferior solutions (*explores*) probabilistically. Such a move helps to avoid getting trapped in the local optimum. SA is inspired by the physical process of annealing in metallurgy, where a metal is heated to a very high temperature. The so obtained molten metal is then cooled in a controlled and slow manner, so that the final resulting metal achieves a crystalline structure with less defects. In the recent past, SA has been used to solve wide range of optimization problems in wireless networks in different contexts, such as localization [24], mobility [25], and routing [26].

4.2 Routing Cost Function

SeeR uses three locally available observations to make efficient routing decisions–1) Estimated ICT of a node, 2) TTL of a message, and 3) Current hop-count of a message. In the following, we elaborate on these metrics.

We consider that each node $i \in N$ in the OMN maintains an exponential weighted moving average (EWMA) of its ICT value denoted by $\hat{\tau}_i$. In particular, let T_{ij} be the time instant when the previous contact of i with another node, say j, terminated. Then, at the next contact event with j at time instant, say t, the average ICT value of node i is updated as

$$\hat{\tau}_i = \alpha \times \hat{\tau}_i + (1 - \alpha) \times (t - T_{ij}), \quad (3)$$

where $\alpha \in (0, 1)$ determines how much weight should be given to the historical estimate of ICT. Subsequently, T_{ij} is set to t.

The residual TTL, $\rho(m, t)$, of a message $m \in M$ at any time instant t is determined as

$$\rho(m, t) = m.ttl - (t - m.createdAt), \quad (4)$$

where $m.createdAt$ denotes the time instant when m was created. Finally, let $h : M \times N \mapsto \mathbb{N}_{\geq 0}$ be a function that denotes the hop-count of a message m at a node i, $m \in M, i \in N$; $\mathbb{N}_{\geq 0}$ is the set of natural numbers including zero. For example, if i is the source of a message m, then

$h(m,i) = 0$. If i replicates the message to another node say, j, then $h(m,j) = 1$.

Based on these, we define a per-message cost function taking into account the current hop-count of the message and the average aggregated ICT of the node. Formally, the cost of having a message m at node i is given by

$$C(m,i) = \hat{\tau}_i \times [1 + h(m,i)]. \quad (5)$$

The hop-count function is incremented by unity in order to prevent the cost function from becoming zero at the source of a message. Let us now consider that node i, which has a message m, comes in contact with another node j, which does not carry that message. It may be observed that, if the message m is replicated to j, the hop-count of m at j increases by unity. In other words, the replication action allowed the message to travel a further hop, which increased the cost of its routing. Therefore, the cost function at j is computed as

$$C(m,j) = \hat{\tau}_j \times [2 + h(m,i)]. \quad (6)$$

Equations (5) and (6) give the cost of a message at two nodes. In the remainder of this section, we look at how these costs are utilized.

4.3 Simulated Annealing of Messages

Let θ_0 be the initial *temperature* of a message $m \in M$, which is assigned to it when it is generated. Also, let $\theta(m)$ be its current temperature. Beginning with θ_0, at every potential replication attempt, the temperature of a message decreases by a scale factor, γ, termed as the *cooling coefficient*. Therefore, at the kth replication attempt of m, its current temperature becomes $\theta(m) = \gamma^k \times \theta_0$. The value of γ is usually taken in between 0.85 and 0.99. Replication of a message is attempted at every contact until its current temperature exceeds a lower threshold, ε. It may be noted that, in general, $\theta(m_1) \neq \theta(m_2)$, where m_1 and m_2 are two different messages. Moreover, a change in $\theta(m_1)$ does not alter $\theta(m_2)$, and vice versa.

To state simply, the objective herein is to determine a subset of messages at any given node that can be replicated to the other node(s) during a given contact. In other words, given a set of messages, the objective translates into evaluating the fitness of each message carried by a given node for replication to others. We leverage the SA framework here, which essentially gives us the set of "best" messages to replicate. The detailed steps of this annealing process are described below.

4.3.1 Temporal Neighborhood Pruning

Let $N_i(m,t)$ be the temporal neighborhood of a message m at node i at any time instant t. Here, $N_i(m,t)$ denotes a set of nodes that have not already received m, and have the potential to deliver–or, replicate to another node at least–before the TTL of m expires. In particular, replicating m to a node whose average ICT is greater than its residual TTL, $\rho(m,t)$, does not constitute an efficient action. However, due to inherent uncertainty involved with the current estimate of ICT as well as its future values, we consider a slightly stricter condition here. Based on (4), we define $\rho'(m,t) = m.ttl - 2 \times (t - m.createdAt)$. Thus, ignoring TTL, ρ' decays twice as fast ρ with respect to time. The formulation of $\rho'(m,t)$ is simple and involves components required to evaluate $\rho(m,t)$. Moreover, resource-constrained devices may possibly replace the multiplication by 2 with bit-shift operation for efficiency. However, further hastening the decay in ρ' can be counterproductive due to elimination of a large number of candidate nodes. Based on these points of reasoning, we define the temporal neighborhood set of a message m at time instant t as $N_i(m,t) = \{x \in N | x \neq i, \hat{\tau}_x < \rho'(m,t)\}$.

Limited storage capacity of the nodes, however, can give rise to another problem, depending upon the traffic intensity and TTL of the messages. When a node's buffer is full, it would need to drop one or more message(s) to receive a new one. It might so happen that at a later instant of time, a node once again receives the same message that it had dropped earlier to free up buffer space. It cannot be guaranteed whether or not receiving that message again would improve its chance of delivery. However, such an action would definitely increase the number of replications in the OMN.

To avoid the above described redundancies in message replication, we assume that every node maintains a Message Lookup Table (MLT) that records the history of messages received. In particular, when a node receives a message from another node, it adds a record $\langle message\_id, expiry\_time \rangle$ into its MLT. Before two nodes actually begin transferring messages, they exchange their respective summary vectors. We assume that, based on such a summary vector and its MLT, a node informs[5] the other node the list of messages it is ready to receive.

The aforementioned process, however, leads to storage overhead. To cope with this, we assume that after every time period T_M, records from MLT corresponding to the messages whose TTL has expired are removed. It may be noted that although MLT bears similarity with the immunity list [27] to a little extent, they are different in multiple respects. First, the immunity list was proposed only to maintain records of delivered messages. Second, MLT is more advanced than the immunity list, and can autoclean over time. Finally, an MLT is only known by its host node–it is not exchanged with others unlike an immunity list.

4.3.2 Message Replication

Once the temporal neighborhood of a message is determined, the potential candidates are evaluated for their fitness of message delivery. Specifically, any node that is not currently in the temporal neighborhood is not considered for replication of the concerned message. Let us, once again, consider that a node i is contemplating the replication of a message m to another node j. Let us define the difference in the two cost functions as

$$\Delta C = C(m,j) - C(m,i). \quad (7)$$

Node i decides to replicate,[6] if the difference in cost is negative. In other words, such a replication would help the message to move from a higher cost (or energy) state to a lower cost state. However, when the difference in cost is

5. For example, if a node receives a summary vector containing message ID say, m, and finds a record for m in its own MLT, the node also includes m in the summary vector that it transmits.

6. The negative cost required for replication in (7) implies that $\tau_j \leq \frac{1+h(m,i)}{2+h(m,i)} \tau_i$. Thus, SeeR attempts to replicate messages to nodes having lower ICT values because such nodes have higher chances of coming in contact with others, on an average.

non-negative, node i replicates the message to j with an *acceptance probability* given by

$$A = \exp\left[\frac{-\Delta C}{\theta(m)}\right]. \tag{8}$$

Algorithm 1 summarizes the operations of SeeR. In particular, Algorithm 1 is executed by any node, when a connection up event with another node is triggered.

Algorithm 1. Actions of Node i on a Connection up Event with j

Inputs:

- M_i: Set of messages currently held by node i
- t: Current time

Output:

- Messages replicated to j

for $m \in M_i$ **do**
  $\theta(m) = \gamma \times \theta(m)$ `// Cooling schedule`
  if $\hat{\tau}_j > \rho'(m,t)$ **then**
    `// Neighborhood filtering`
    continue
  if $\theta(m) > \varepsilon$ **then**
    Compute $C(m,i)$ and $C(m,j)$ using (5) & (6)
    $\Delta C = C(m,j) - C(m,i)$
    Compute A using (8)
  r = Random number between 0 and 1
  if $\Delta C \leq 0$ *or* $r < A$ **then**
Replicate m to j

5 Insights into SeeR

In this section, we take a detailed look at the protocol state maintenance of SeeR and its implications upon privacy risks. We also quantify the storage and transmission overhead of SeeR and contrast with other contemporary routing protocols.

5.1 Routing Metadata and Privacy Aspects

As discussed in Section 2, a large number of routing schemes proposed for OMNs store and use information about the other nodes in the network. For example, PRoPHET maintains delivery predictability values for every other node, whereas CBR maintains frequency of contacts with others, and DF stores the ICT of nodes. Moreover, nodes using these routing protocols exchange such metadata among themselves when they come in contact.

In general, routing protocols can be categorized into two broad categories. The first category consists of schemes that continuously keep accumulating routing metadata, such as PRoPHET. The second category consists of schemes that periodically clean up accumulated routing metadata. As we shall see in this Section, SeeR belongs to the second category. The following theorem is stated in the context of the first category of protocols.

Theorem 2. *Let P be a routing protocol, wherein the nodes maintain and exchange information about the other nodes in an OMN. Let $R_P(t)$ be a set used by a node for storing (as key-value pairs) such metadata. Then, the size of the set, $|R_P(t)|$, is a monotonically increasing function of time t. Moreover, as $t \to \infty$, $|R_P(t)| = |N|$.*

Proof. Let us consider that a node in the OMN comes in contact with another node at a time instant $t + \Delta t$. Moreover, let K be the set of metadata that the former node receives from the latter. Then $R_P(t+\Delta t) = R_P(t) \cup K$. The set K can contain information about zero or more nodes not yet visited by the first node. If no such metadata about any new node is available, the former would (possibly) update the contents of its set so that the size remains same, i.e., $|R_P(t+\Delta t)| = |R_P(t)|$. On the other hand, if the set K has information about some not yet visited nodes, then $|R_P(t+\Delta t)| > |R_P(t)|$. Therefore, in general, $|R_P(t+\Delta t)| \geq |R_P(t)|$. Moreover, in the limiting case, a node has information about all other nodes in the OMN so that $\lim_{t\to\infty} |R_P(t)| = |N|$. □ □

Theorem 2 essentially states that, as time passes, the nodes in an OMN tend to accumulate information about every other node in it. However, for the sake of fairness, we should note that at any instant of time, such information available at a node may not be accurate–or rather, they are often likely to be stale information. Nevertheless, even with such approximation, data about others are available with the nodes.

Such per-node information can, however, put user's privacy at risk. In particular, Cunche et al. [28] used Wi-Fi fingerprinting to identify social links among users. Since OMNs work atop similar link layer protocols, such observations are of importance.

As an example, if we sort the delivery predictabilites maintained by a node using PRoPHET in descending order, we would be able to identify a set of device address[7] that the concerned node often comes in contact with. Using these device addresses, together with some social and network engineering, one may be able to infer the real-life users, who own those devices. Such *social contact profiling* is undesirable. Ideally, network operations should not put privacy of users at risk. At the same time, gathering some information is essential for efficient operation. Therefore, a tradeoff in this regard is desirable.

Such contact profiling risks, however, do not arise, when aggregate information (for example, total number of contacts and aggregate ICT) are used, rather than when *node-specific* metadata are used. If we know that the aggregate ICT of a device is τ, we can only infer that, on an average, the user of the device comes in contact with others every τ unit of time. However, even if node-specific information are maintained, the aforementioned risk can mitigated to some extent, at least if the nodes periodically cleanup metadata about the others. Theorem 3 is stated in this context.

Theorem 3. *Let Q be a routing protocol, wherein the nodes maintain, and reset every time period T_C, information about the other nodes in an OMN. Then, $|R_Q(t, T_C)| \leq |R_P(t)|$.*

Proof. It can be observed that $\lim_{T_C\to\infty} |R_Q(t, T_C)| = |R_P(t)|$. In other words, with infinitely large metadata cleanup period, Q degenerates into P. However, when T_C is finite and small, at the end of such a period, the metadata

7. Theoretically, link and network layers would likely have different addresses. Here, we assume that some form of mapping from one to the other is available. Moreover, such addresses need to be transmitted via wireless medium, and can be listened by others.

gathered by a node about others would be cleaned up, so that $|R_Q(t, T_C)| = 0$. Subsequently, the node would again engage in accumulating metadata until the period ends. Therefore, at any given time instant t, a node using protocol R would gather as much metadata, or even more, as protocol Q. So, in general, $|R_Q(t, T_C)| \leq |R_P(t)|$. $\square$

The question that immediately arises is–what are the implications of Theorem 3 for SeeR? Before going into it, let us briefly recollect how aggregate ICT is calculated by a node. Let us consider a node i that comes in contact with another node j for the first time. When the contact between i and j goes down, i initiates a timer for node j, and maintains it until it comes in contact with j again. The time elapsed in between, as reflected by the timer, gives a single ICT between i and j. This process is repeated for every contact up/down events with any other node, and their average is computed to get the ICT value. In particular, it takes a single (constant) unit of storage to maintain the aggregate ICT value using EWMA. However, in the worst case, a node may require to maintain $|N|$ such timers.

SeeR maintains and uses aggregate ICT to evaluate the cost function, and therefore, does not pose privacy risk. However, to maintain the aggregate ICT, SeeR requires temporary timers, as discussed above. We incorporate the observations from Theorem 3 into SeeR here. In particular, after every time period T_C, the contact engine of SeeR (Fig. 1) deletes all the ICT timers. As a consequence, if we look at the state maintained by SeeR at any node at any instant of time, we find only three pieces of information–1) the node's aggregate ICT, 2) some temporary timers indicating the time since last contact with the corresponding nodes, which is valid only for the current time period, and 3) a list of messages in MLT that are received by the node. Moreover, none of these information are shared with others. Thus, SeeR puts much less information at the risk[8] of exposure than many other contemporary routing protocols.

5.2 Storage and Transmission Overhead

We note that SeeR requires $O(|N|)$ storage for maintaining the ICT timers. This is consistent with similar other protocols, such as PRoPHET, CBR, and DF (when destination-specific utilities are used). However, unlike those protocols, the storage space required by SeeR for these timers is temporary and freed periodically.

The aforementioned protocols require $O(|N|)$ transmission overhead as well, since these information are exchanged among the nodes at every contact. SeeR, however, does not require such exchanges. Specifically, in SeeR, a node maintains only its own ICT. Moreover, when two such nodes come in contact, they exchange their respective estimates of average ICT, which is stored temporarily for relevant computations in Algorithm 1. Therefore, the resultant transmission overhead of SeeR becomes $O(1)$. For fairness, it should be noted that overhead involved in exchanging such metadata is often ignored due to their much less size than an actual message. However, when aggregated over a long period, they tend to consume a considerable chunk of bandwidth.

Additionally, for each message,[9] SeeR requires maintaining its current temperature. Since each message already has a lot of metadata associated with itself (such as source and destination addresses, creation time, and intermediate hops), inclusion of another field for temperature incur insignificant storage and transmission overhead.

6 Experimental Setup

In this section, we discuss the experimental setup used to evaluate the performance of SeeR. The Opportunistic Network Environment (ONE) [29] simulator was used for this purpose. We also discuss the benchmarks and performance evaluation metrics used in the process.

6.1 Mobility Scenarios

The following real-life connection traces and synthetic mobility model were used in our simulations.

- *Infocom'05* and *Infocom'06* [30]: These connection traces were collected during the INFOCOM conferences in 2005 and 2006, respectively. The traces were obtained based on the contact events logged by iMotes carried by 41 and 78 users, respectively. The transmission ranges of the nodes were 30 m. We simulated the contact events from the first 71 and 24 hours, respectively, from these two traces.
- *Truncated Levy Walk (TLW)* [31]: Extensive research has shown that human movement can be modeled using Levy Walk. The TLW model was developed to synthetically generate mobility patterns that closely resemble real-life human movement. We used an implementation of the TLW model, as described in [9]. We considered 50 nodes moving for 3 days in a terrain of size 1.5×1.5 sq. Km with transmission range 100 m. We took the ensemble average of results from 15 scenarios, and determined the corresponding 95 percent confidence interval. Only 15 scenarios were considered to reach a tradeoff between accuracy and simulation speed.

The use of these traces and mobility model allowed us to consider scenarios with diverse characteristics, such as ICT distribution, node density, and transmission ranges.

6.2 Parameter Values for Simulated Annealing

We experimented with different values for the parameters of SA used together in SeeR. Fig. 2b indicates an increasing trend of overhead ratio as the initial temperature of the messages, θ_0, increases. This is due to the reason that, with higher initial temperature, the SA process results in a greater scope for exploration. In other words, the number of instances where a message is replicated to another node with a higher cost function becomes large. Thus, from the perspective of overhead ratio, very high value of θ_0 is undesirable for SeeR.

8. Finally, it may be noted that even though SeeR does not explicitly exchange node-specific contact utilities, delivery path (list of intermediate nodes through which a message has passed) information available in message headers can still be used to estimate contact probabilities among the nodes of an OMN. Therefore, we suggest that while implementing SeeR, delivery paths should not be maintained in message headers.

9. We ignore the overhead due to MLT because, as we shall see later, the size of an MLT is significantly less, on an average.

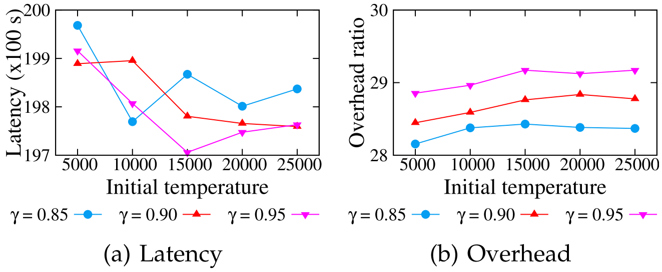


Fig. 2. Effects of initial temperature (θ_0) and cooling coefficient (γ) upon network performance using the Infocom'05 trace.

On the other hand, the average message delivery latency seemed to slightly decrease with an increase in initial temperature in the Infocom'05 scenario as shown in Fig. 2a. A clear trend, however, did not emerge for other scenarios. Based on these results, we narrowed down to the choice of $\theta_0 = 15,000$ and $\gamma = 0.95$ to settle down at a reasonable tradeoff between latency and overhead. These, along with other simulation parameters, are summarized in Table 2.

6.3 Benchmarks and Evaluation Metrics

The performance of SeeR was compared with that of the following routing protocols.

- *PRoPHET (version 2)* [1]: PRoPHET was considered for multiple reasons. First, it is a simple routing protocol that can result in delivery of greater number of messages. Second, it has been tested in real-life based on which, the version 2 has evolved. Finally, its complete technical specification is available in RFC 6,693.[10] The values of the transitivity and aging constants were taken as 0.9 and 0.999885791, respectively. Time units of 60 seconds were considered.
- *Delegation Forwarding* [2]: DF is a state-of-the-art routing protocol, which ensures high performance with low routing overhead. We used frequency of contacts as the quality metric for DF.
- *Contact-based Routing* [13]: CBR is a recently proposed routing algorithm, which chooses the neighbor with the maximum contacts as the best next hop.

The following performance evaluation metrics were used to contrast SeeR with others.

- *Message delivery ratio*: It is the ratio of the number of messages delivered to the number of messages created.
- *Average message delivery latency*: Delivery latency of a message is the time duration since its generation to until its delivery. The average latency is obtained by taking mean of delivery latencies of all the delivered messages.
- *Overhead ratio of message delivery*: The overhead is computed as the difference between the total number of message replicas and the number of messages delivered. The overhead ratio is the ratio of this difference to the total number of delivered messages.
- *Metadata accumulated*: A measure of the fraction of node in an OMN about whom some form of routing metadata is collected and exchanged.

10. https://www.rfc-editor.org/rfc/rfc6693.txt

TABLE 2
Simulation Parameters

Parameter	Value
Buffer size (MB)	50, 100, 150, 200, 250
Message TTL (hours)	8, 12, 16, 20, 24
Message generation interval (seconds)	25–35 (high), 600–1,200 (low)
Message size (MB)	0.5–1
α	0.6
θ_0	15,000.0
γ	0.95
ε	0.001
Contacts reset period (T_C), MLT reset period (T_M)	10–12 hours

- *Size of MLT*: It is measured by the number of records present in the MLT of a node, on an average. The physical storage requirements would be linearly proportional to the size of MLT.

An optimal routing protocol results in high delivery ratio, but low latency and overhead.

7 Results

In this section, we present the results of simulation-based performance evaluations and related discussions. Subsequently, we take a holistic view and reflect upon the efficiency of SeeR, in general.

7.1 Effects of Buffer Size

Figs. 3, 4, 5, and 6 show the delivery ratio, the average message delivery latency, and the overhead ratio obtained using

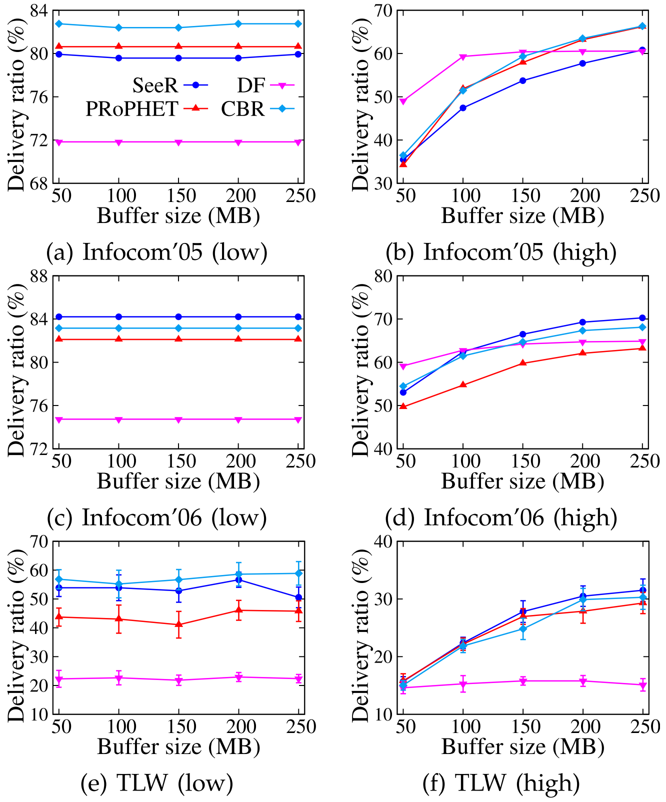
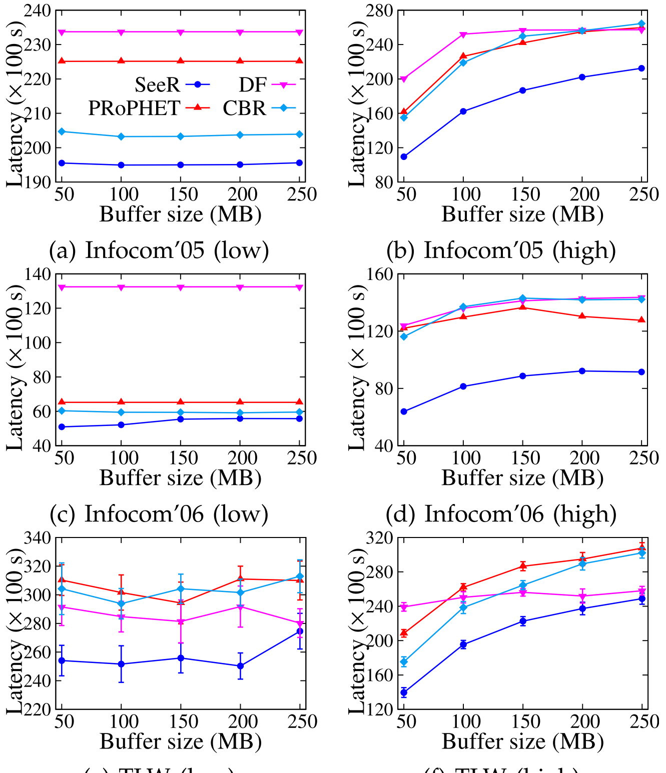
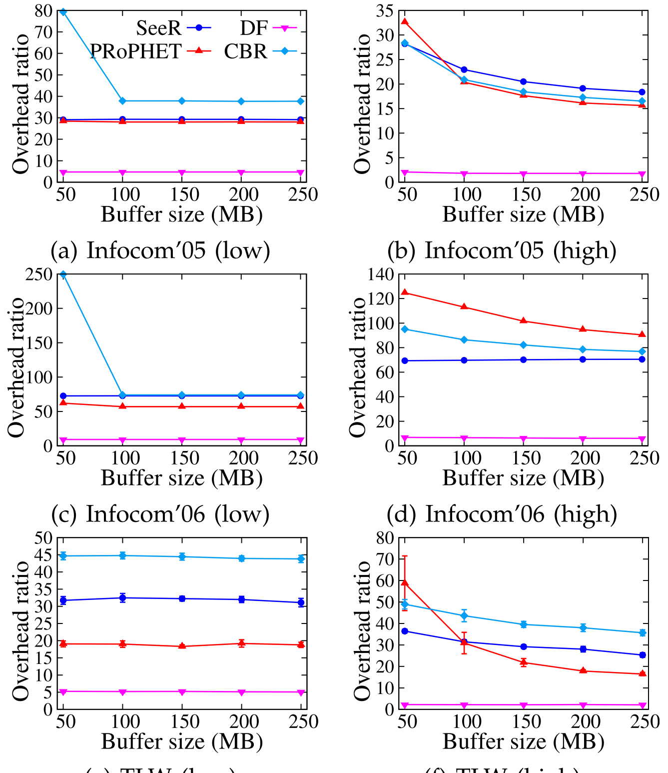


Fig. 3. Effect of buffer size on delivery ratio using different mobility scenarios and traffic intensities. TTL was taken as 20 hours.



(a) Infocom'05 (low) (b) Infocom'05 (high) (c) Infocom'06 (low) (d) Infocom'06 (high) (e) TLW (low) (f) TLW (high)

Fig. 4. Variation in average message delivery latency under different mobility scenarios and traffic intensities.

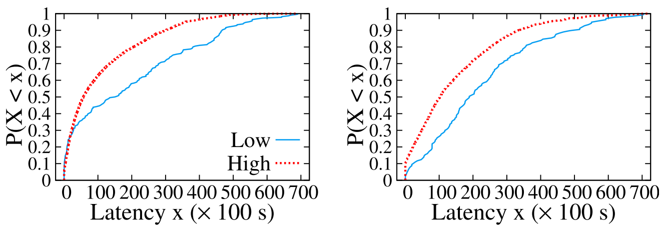


(a) Infocom'05 (low) (b) Infocom'05 (high) (c) Infocom'06 (low) (d) Infocom'06 (high) (e) TLW (low) (f) TLW (high)

Fig. 6. Variation in overhead ratio for different buffer sizes under diverse mobility scenarios and traffic intensities.

different routing protocols under diverse mobility scenarios, when the nodes had different buffer capacities. The Figure captions with "(low)" and "(high)", respectively, indicate results obtained using low and high traffic intensities, as summarized in Table 2.

Fig. 3 shows that the delivery ratio of messages obtained with SeeR was almost as good as the other three routing protocols considered here. As a sanity check, we observe that the overall message delivery ratio in scenarios with low traffic intensity were higher than with high intensities. In the Infocom'05 and Infocom'06 scenarios with low traffic intensity (Figs. 3a and 3c), SeeR, PRoPHET, and CBR resulted in very close delivery ratio of messages. When these scenarios were considered together with high traffic intensity (Figs. 3b and 3d), we find that, on increasing the buffer size, the difference between the delivery ratio of messages obtained using SeeR and other protocols narrowed down steadily. The only relative performance degradation was seen with high traffic in the Infocom'05 scenario (Fig. 3b), where all the protocols except DF fared poorly up to 100 MB buffer size.



(a) Infocom'05 (b) TLW

Fig. 5. Message delivery latency distribution using SeeR.

From Fig. 3, it can also be observed that with low traffic intensity, DF fared significantly poorer than the other routing protocols. Moreover, the performance of DF largely remained unaffected with changes in buffer size. This can can be accounted to the strict criteria that DF imposes upon message replication. SeeR and others, on the other hand, employed a relatively "relaxed" replication rule, and therefore, the competition among messages for buffer occupancy reduced on increasing buffer sizes. This helped in ensuring the delivery of larger number of messages.

Fig. 4 shows the average message delivery latency obtained in different scenarios. It can be observed that, when SeeR was used, the mean latency obtained was not only minimum, but also considerably less than the other routing protocols in most of the scenarios. In particular, in Fig. 4b, SeeR resulted in about 58 percent reduction, as compared to the maximum latency resulted in by DF. These observations are significant, since in real-life, it is desirable to have greater number of messages delivered with lower latencies. The Figure also shows that the average message delivery latency with DF is generally high.

An apparently strange observation in Fig. 4 is that, in some instances, the average latencies obtained using low traffic intensity were higher than the corresponding values obtained using high intensity. For example, in the Infocom'05 scenario in Fig. 4a, the average latency was about 20,000 seconds when the buffer size was 50 MB, whereas in Fig. 4b, the same latency reduced to about 11,000 seconds. This can be explained with reference to Fig. 5, which shows the distribution of message delivery latencies obtained using SeeR in the Infocom'05 and TLW scenarios. The buffer size was taken as 50 MB and TTL of the messages were 12 hour. It can seen

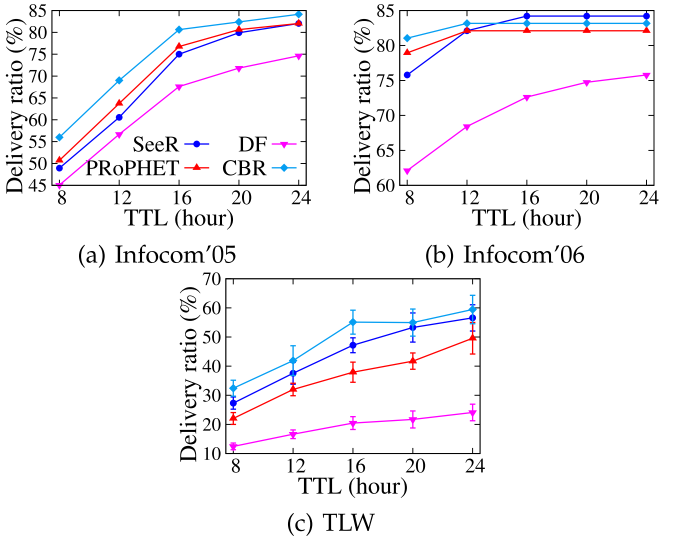


Fig. 7. Effect of TTL on delivery ratio using different mobility scenarios. Low traffic intensity was considered.

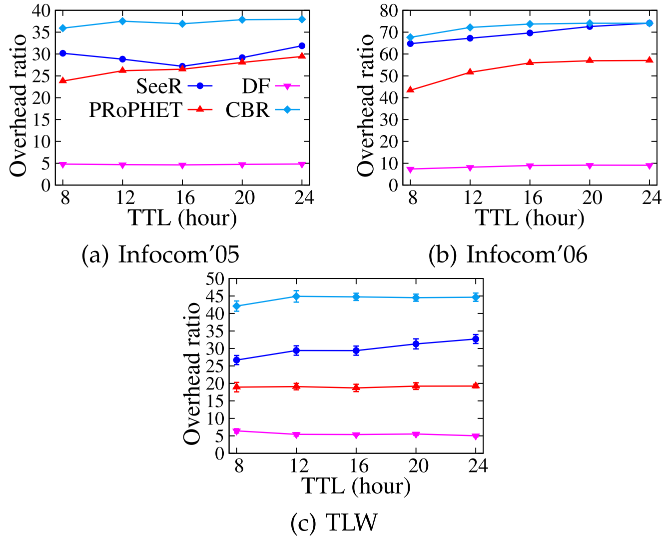


Fig. 9. Variation in overhead ratio for different buffer sizes under diverse mobility scenarios.

that the empirical cumulative distribution function (CDF) plots corresponding to the low traffic intensities move from 0 to 1 far more slowly than those for high intensities. In other words, the latency distribution for low traffic intensity was far more diverse, which not only resulted in wide confidence intervals in Fig. 4e, but also moved up the average value. Additionally, it should also be noted that the average message delivery ratio obtained using low traffic intensity was much higher than when high intensity was used.

Fig. 6 shows the overhead ratios for all the scenarios. Undoubtedly, DF not only resulted in the minimum overhead, but it was significantly low. Moreover, although CBR usually fared better than PRoPHET in terms of delivery ratio and latency, it did not show any clear trend with respect to overhead ratio. On the other hand, the overhead ratio obtained using SeeR was upper bounded by at least another protocol except in Fig. 6b. In other words, SeeR engaged in some unnecessary message replications in that particular scenario.

7.2 Effects of TTL

We now look at the performance of different protocols when TTL was varied. For the sake of brevity, we show only the results pertaining to low traffic intensity.

Fig. 7 shows that, an increase in TTL of the messages enhanced their chances of delivery in all scenarios for all the protocols. It can be observed that SeeR, in general, offered as good delivery ratio as the other state-of-the-art routing protocols. Moreover, the message delivery latency, on an average, was lower than the others by different extents (Fig. 8). Fig. 9 shows the lack of any pronounced effect on overhead ratio due to the variation in TTL. The overhead of SeeR, however, was less than that of CBR. Moreover, temporal neighborhood pruning helped SeeR to maintain the overhead below $|N|$.

7.3 Routing Metadata Accumulation

Fig. 10 shows the fraction of nodes about whom routing metadata were accumulated by SeeR and PRoPHET with

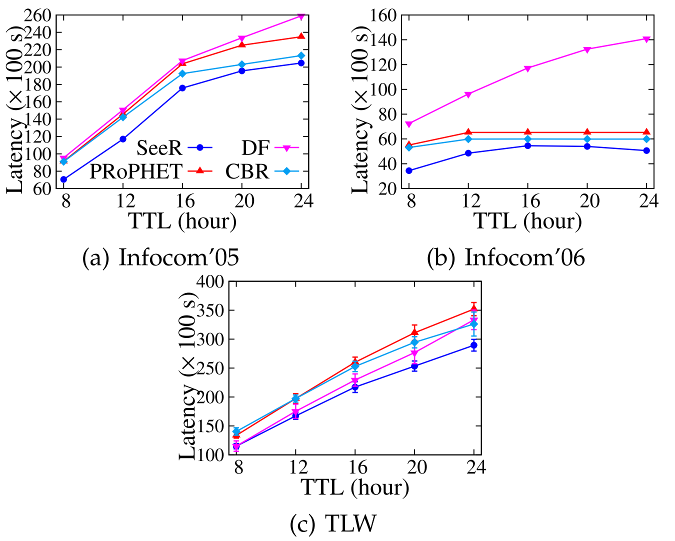


Fig. 8. Variation in average message delivery latency under different mobility scenarios.

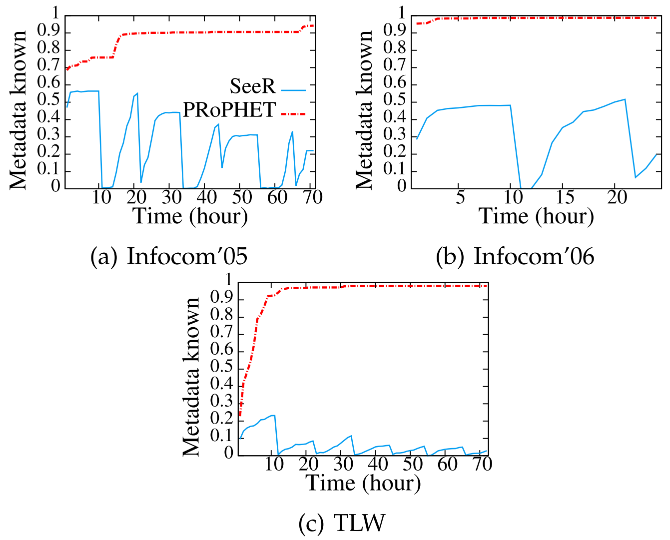


Fig. 10. Routing metadata accumulated for other nodes as a fraction of total number of nodes under different mobility scenarios.

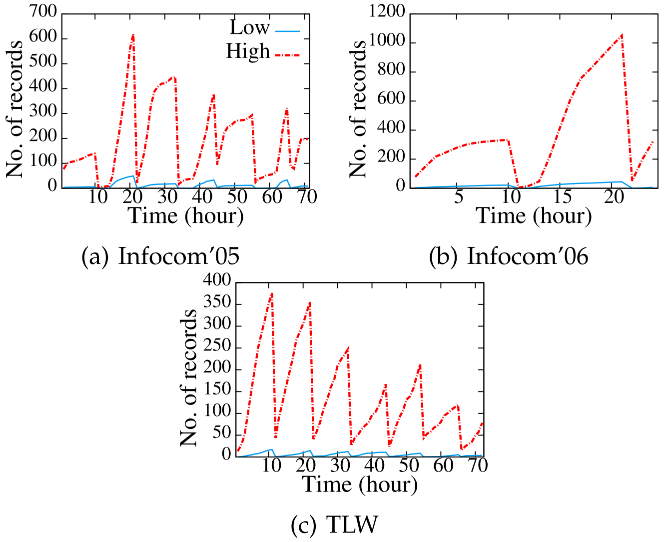


Fig. 11. Variation in the size of MLT with time. Here, TTL = 20 hour and buffer size = 150 MB.

respect to time. The results for PRoPHET shown here are equally valid for any other routing protocol that maintains and exchanges similar metadata among the nodes. The claims of Theorems 2 and 3 can be verified from these results.

In particular, it can be observed from Fig. 10a that when PRoPHET (or any other similar protocol) was used in the Infocom'05 scenario, information related to about 90 percent of the nodes, on an average, were available to any node in the OMN within the first 20 hours of simulation. In the Infocom'06 scenario (Fig. 10b), however, it took much less time to accumulate such data due to frequent contacts among the nodes. When contrasted, we find that SeeR accumulated significantly less metadata (temporary ICT timers) about the nodes in an OMN. The Fig. 10, in general, shows gradually-rising-up-and-suddenly-falling-down patterns formed as a result of accumulation of routing metadata and periodic reset. Thus, from a practical point of view, SeeR has much lower risk of exposing information collected about other nodes in an OMN.

7.4 Size of Message Lookup Table

Fig. 11 shows the variation in the size of MLT with respect to time under different mobility scenarios and traffic intensities (indicated by "Low" and "High" in the Figure's legend). With low traffic intensity, the size remained arguably very low, on an average. However, with high traffic intensity, the number of records in MLT spiked up to about 1,000. Let us look at the storage requirement by considering that message ID and expiry time each requires 8 bytes. Then, with 1,024 such records, for example, the size of an MLT, on an average, would be $1,024 \times (8+8)/1,024 = 16$ KB. This is insignificant, considering the facts that a message itself would require a few KBs and that the storage availability would be around hundred MBs. Therefore, the use of an MLT is practically feasible.

7.5 The Big Picture

A potential question that arises at this point is–can metaheuristics-based SeeR perform well in all scenarios? In the spirit of the "No Free Lunch Theorem", we discount such a claim. In reality, it is hard, if not impossible, to find a routing protocol for OMNs that performs well in every scenario. However, the big picture that emerges out of this work is that, when subjected to diverse mobility patterns and traffic intensities, SeeR can perform reasonably well. In general, the average message delivery latency of SeeR is much low and the overhead is within the acceptable bounds. Deviations in the message delivery ratio are typically within the 5 percent bound except for a few stray cases.

TABLE 3
Normalized Routing Performance

Routing scheme	Normalized delivery ratio	Normalized latency
(Buffer size: 50 MB)		
SeeR	0.72	1.00
PRoPHET	0.69	1.47
DF	1.00	1.83
CBR	0.74	1.41
(Buffer size: 250 MB)		
SeeR	0.91	1.00
PRoPHET	0.99	1.22
DF	0.91	1.21
CBR	1.00	1.24

In order to have a better insight, let us normalize the performance results obtained earlier in the following way. Let S be the set of routing protocols under consideration. Let $\delta(x)$ and $\ell(x)$, respectively, be the delivery ratio and delivery latency obtained using a routing protocol $x \in S$. Then, we define the normalized delivery ratio of any routing scheme x as: $\frac{\delta(x)}{\underset{s\in S}{max}\{\delta(s)\}}$. Similarly, the normalized delivery latency is expressed as: $\frac{\ell(x)}{\underset{s\in S}{min}\{\ell(s)\}}$. Table 3 shows the normalized results (correct up to two decimal places) pertaining to different routing protocols in the Infocom'05 scenario with high traffic intensity (50 and 250 MB buffer sizes; 1,200 minutes of TTL).

The Table shows that SeeR offers the minimum normalized message delivery among the four routing schemes considered. In the scenario with 50 MB buffer size, the normalized delivery ratio obtained using SeeR is 0.28 units less than the maximum value, which is offered by DF. However, the normalized latency of DF is 0.83 units more than that of SeeR. This underscores the fact that the delivery ratio-delivery latency tradeoff is tilted in favor of SeeR. Moreover, the lack of mutual information exchange among the nodes as well as refrain from the use of destination-specific utilities adds a thin "privacy guarding" layer to SeeR. When such prospects are considered altogether, they indicate toward the feasibility of the use of SeeR in real-life applications.

8 Conclusion

In this work, we presented SeeR, a SA-based routing protocol for OMNs. In SeeR, a cost function is defined for each message by considering its contemporary hop-count and average ICT of the node. SeeR exploits by replicating a message to a node with a lower cost function value. Additionally, it explores by probabilistically replicating a message to a node with higher cost. Moreover, in SeeR, a node solely relies upon its own local observations, and does not exchange any metadata with the other nodes, which limits the risk of privacy leaks.

This work has been a small attempt towards "doing more with less" routing information, and can be enhanced further in several ways. As an example, the integration of reputation aspects together with SeeR would be an interesting study. Additionally, alternative choice of cost metrics can be explored to further reduce the overhead. Moreover, energy budget of the nodes can be taken into consideration. Such improvements can take development of an efficient routing protocol for OMNs further ahead. For the benefit of the readers, we make the source code of SeeR available at https://github.com/barun-saha/one-simulator under an open source license.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their feedback and suggestions that has helped to improve the quality of this work.

REFERENCES

[1] S. Grasic, E. Davies, A. Lindgren, and A. Doria, "The evolution of a DTN routing protocol-PRoPHETv2," in *Proc. 6th ACM Workshop Challenged Netw.*, 2011, pp. 27–30.
[2] V. Erramilli, M. Crovella, A. Chaintreau, and C. Diot, "Delegation forwarding," in *Proc. 9th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2008, pp. 251–260.
[3] I. Arapakis, X. Bai, and B. B. Cambazoglu, "Impact of response latency on user behavior in web search," in *Proc. 37th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2014, pp. 103–112.
[4] A. Crescenzi, D. Kelly, and L. Azzopardi, "Impacts of time constraints and system delays on user experience," in *Proc. ACM Conf. Human Inf. Interaction Retrieval*, 2016, pp. 141–150.
[5] G. Anderson, R. Doherty, and S. Ganapathy, *User Perception of Touch Screen Latency*. Berlin, Germany: Springer, 2011, pp. 195–202.
[6] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Sci.*, vol. 220, no. 4598, pp. 671–680, May 1983.
[7] S. Zolfaghari and M. Liang, "Comparative study of simulated annealing, genetic algorithms and tabu search for solving binary and comprehensive machine-grouping problems," *Int. J. Prod. Res.*, vol. 40, no. 9, pp. 2141–2158, 2002.
[8] S. Kundu, M. Mahato, B. Mahanty, and S. Acharyya, "Comparative performance of simulated annealing and genetic algorithm in solving nurse scheduling problem," in *Proc. Int. MultiConference Eng. Comput. Scientists*, Mar. 2008, vol. 1, pp. 96–100.
[9] B. K. Saha, S. Misra, and S. Pal, "Utility-based exploration for performance enhancement in opportunistic mobile networks," *IEEE Trans. Comput.*, vol. 65, no. 4, pp. 1310–1322, Apr. 2016.
[10] S. Misra, S. Pal, and B. K. Saha, "Distributed information-based cooperative strategy adaptation in opportunistic mobile networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 3, pp. 724–737, Mar. 2015.
[11] E. P. C. Jones, L. Li, J. K. Schmidtke, and P. A. S. Ward, "Practical routing in delay-tolerant networks," *IEEE Trans. Mobile Comput.*, vol. 6, no. 8, pp. 943–959, Aug. 2007.
[12] A. Lindgren, A. Doria, and O. Schelén, "Probabilistic routing in intermittently connected networks," in *Proc. 1st Int. Workshop Service Assurance Partial Intermittent Resources*, Aug. 2004, vol. 3126, pp. 239–254.
[13] S. Pal and S. Misra, "Contact-based routing in DTNs," in *Proc. 9th Int. Conf. Ubiquitous Inf. Manage. Commun.*, 2015, pp. 3:1–3:6.
[14] T. Abdelkader, K. Naik, and A. Nayak, "Choosing the objective of optimal routing protocols in delay tolerant networks," in *Proc. Int. Comput. Eng. Conf.*, Dec. 2010, pp. 16–21.
[15] T. Abdelkader, K. Naik, A. Nayak, N. Goel, and V. Srivastava, "SGBR: A routing protocol for delay tolerant networks using social grouping," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 12, pp. 2472–2481, Dec. 2013.
[16] M. Y. S. Uddin, H. Ahmadi, T. Abdelzaher, and R. Kravets, "Intercontact routing for energy constrained disaster response networks," *IEEE Trans. Mobile Comput.*, vol. 12, no. 10, pp. 1986–1998, Oct. 2013.
[17] B. Han, P. Hui, V. S. A. Kumar, M. V. Marathe, J. Shao, and A. Srinivasan,"Mobile data offloading through opportunistic communications and social participation," *IEEE Trans. Mobile Comput.*, vol. 11, no. 5, pp. 821–834, May 2012.
[18] X. Fan, V. Li, and K. Xu, "Fairness analysis of routing in opportunistic mobile networks," *IEEE Trans. Veh. Technol.*, vol. 63, no. 3, pp. 1282–1295, Mar. 2014.
[19] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Spray and wait: An efficient routing scheme for intermittently connected mobile networks," in *Proc. ACM SIGCOMM Workshop Delay-Tolerant Netw.*, 2005, pp. 252–259.
[20] S. Zakhary, M. Radenkovic, and A. Benslimane, "Efficient location privacy-aware forwarding in opportunistic mobile networks," *IEEE Trans. Veh. Technol.*, vol. 63, no. 2, pp. 893–906, Feb. 2014.
[21] E. Papapetrou, V. F. Bourgos, and A. G. Voyiatzis, "Privacy-preserving routing in delay tolerant networks based on bloom filters," in *Proc. IEEE 16th Int. Symp. World Wireless Mobile Multimedia Netw.*, Jun. 2015, pp. 1–9.
[22] M. Radenkovic, A. Benslimane, and D. McAuley , "Reputation aware obfuscation for mobile opportunistic networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 1, pp. 230–240, Jan. 2015.
[23] B. Distl and T. Hossmann, "Privacy in opportunistic network contact graphs," in *Proc. IEEE 15th Int. Symp. World Wireless Mobile Multimedia Netw.*, Jun. 2014, pp. 1–3.
[24] M. Basheer and S. Jagannathan, "Localization of RFID tags using stochastic tunneling," *IEEE Trans. Mobile Comput.*, vol. 12, no. 6, pp. 1225–1235, Jun. 2013.
[25] R. Rao and G. Kesidis, "Purposeful mobility for relaying and surveillance in mobile ad hoc sensor networks," *IEEE Trans. Mobile Comput.*, vol. 3, no. 3, pp. 225–231, Jul./Aug. 2004.
[26] J. Toutouh, J. Garcia-Nieto, and E. Alba, "Intelligent OLSR routing protocol optimization for VANETs," *IEEE Trans. Veh. Technol.*, vol. 61, no. 4, pp. 1884–1894, May 2012.
[27] P. Mundur, M. Seligman, and G. Lee, "Epidemic routing with immunity in delay tolerant networks," in *Proc. IEEE Mil. Commun. Conf.*, Nov. 2008, pp. 1–7.
[28] M. Cunche, M.-A. Kaafar, and R. Boreli, "Linking wireless devices using information contained in Wi-Fi probe requests," *Pervasive Mobile Comput.*, vol. 11, pp. 56–69, 2014.
[29] A. Keränen, J. Ott, and T. Kärkkäinen, "The ONE simulator for DTN protocol evaluation," in *Proc. 2nd Int. Conf. Simul. Tools Techn.*, 2009, pp. 55:1–55:10.
[30] J. Scott, R. Gass, J. Crowcroft, P. Hui, C. Diot, and A. Chaintreau, "CRAWDAD data set cambridge/haggle (v. 2006–01-31)," Jan. 2006. [Online]. Available: http://crawdad.org/cambridge/haggle/, doi: https://doi.org/10.15783/C70011, Accessed on: Nov. 01 2016.
[31] I. Rhee, M. Shin, S. Hong, K. Lee, S. J. Kim, and S. Chong, "On the levy walk nature of human mobility: Do humans walk like monkeys?" *IEEE/ACM Trans. Netw.*, vol. 19, no. 3, pp. 630–643, Jun. 2011.

Barun Kumar Saha received the MS degree in research from the Indian Institute of Technology Kharagpur, India. He is working toward the PhD degree in the Indian Institute of Technology Kharagpur, India. Prior to that, he was in industry for about 2.5 years. His research interests include delay tolerant networks, mobile ad hoc networks, and the use of technology for advancing education. His research works have been published in several top notch journals and conferences. He has co-authored the book entitled *Opportunistic Mobile Networks: Advances and Applications* published by Springer. He is the lead developer of two Virtual Labs on Software Engineering and Advanced Network Technologies. He maintains a blog on DTN and the ONE simulator at http://delay-tolerant-networks.blogspot.com/, which is widely acclaimed in the community. Further details about him can be found at http://barunsaha.me

Sudip Misra received the PhD degree in computer science from Carleton University, Ottawa, Canada. He is an associate professor in the School of Information Technology, Indian Institute of Technology Kharagpur. His current research interests include algorithm design for emerging communication networks. He is the author of more than 230 scholarly research papers. He was awarded the IEEE ComSoc Asia Pacific Outstanding Young Researcher Award at IEEE GLOBECOM 2012, Anaheim, California. He was also the recipient of several academic awards and fellowships such as the Young Scientist Award (National Academy of Sciences, India), Young Systems Scientist Award (Systems Society of India), Young Engineers Award (Institution of Engineers, India), (Canadian) Governor Generals Academic Gold Medal with Carleton University, the University Outstanding Graduate Student Award in the Doctoral level with Carleton University, and the National Academy of Sciences, India Swarna Jayanti Puraskar (Golden Jubilee Award). He was awarded the Canadian Governments prestigious NSERC Post Doctoral Fellowship and the Humboldt Research Fellowship in Germany. He is the editor-in-chief of the *International Journal of Communication Networks and Distributed Systems*, the *Inderscience*, Switzerland. He has also been serving as the associate editor of the *Telecommunication Systems Journal* (Springer SBM), the *Security and Communication Networks Journal* (Wiley), the *International Journal of Communication Systems* (Wiley), and the *EURASIP Journal of Wireless Communications and Networking*. He has eight books published by Springer, Wiley, and World Scientific. He was also invited to deliver keynote/invited lectures in more than 30 international conferences in USA, Canada, Europe, Asia, and Africa. He is a senior member of the IEEE.

Sujata Pal received the BE degree in computer science from North Orissa University, India, the MTech degree in multimedia and software system from the West Bengal University of Technology, India, and the PhD degree from the Indian Institute of Technology Kharagpur. She received the Tata Consultancy Services (TCS) Research Scholarship for 4 years for pursuing the PhD program. She was awarded the prestigious Schlumberger Faculty for the Future Fellowship for two consecutive years (2015 and 2016). Currently, she is an assistant professor in the Department of Computer Science and Engineering, IIT Ropar, India. She was a post-doctoral fellow with the University of Waterloo, Canada, before joining IIT Ropar. Her research works have been published in high quality international journals, conferences, a book, and a book chapter. Her current research interests includes delay tolerant networks, opportunistic mobile networks, and mobile ad-hoc networks. She is a member of the IEEE.