Computer Networks 81 (2015) 201–219

Contents lists available at ScienceDirect

Computer Networks

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

ML-SOR: Message routing using multi-layer social networks in opportunistic communications

A. Socievole [a,*], E. Yoneki [b], F. De Rango [a], J. Crowcroft [b]

[a] Department of Informatics, Modeling, Electronics and Systems Engineering (DIMES), University of Calabria, 87036 Arcavacata di Rende (CS), Italy
[b] Computer Laboratory, William Gates Building, 15 JJ Thomson Avenue, Cambridge CB3 0FD, UK

ARTICLE INFO

Article history:
Received 17 January 2014
Received in revised form 5 January 2015
Accepted 18 February 2015
Available online 24 February 2015

Keywords:
Opportunistic network
Opportunistic routing
Online social network
Detected social network
Multi-layer network

ABSTRACT

Opportunistic networks are a generalization of DTNs in which disconnections are frequent and encounter patterns between mobile devices are unpredictable. In such scenarios, message routing is a fundamental issue. Social-based routing protocols usually exploit the social information extracted from the history of encounters between mobile devices to find an appropriate message relay. Protocols based on encounter history, however, take time to build up a knowledge database from which to take routing decisions. While contact information changes constantly and it takes time to identify strong social ties, other types of ties remain rather stable and could be exploited to augment available partial contact information. In this paper, we start defining a multi-layer social network model combining the social network detected through encounters with other social networks and investigate the relationship between these social network layers in terms of node centrality, community structure, tie strength and link prediction. The purpose of this analysis is to better understand user behavior in a multi-layered complex network combining online and offline social relationships. Then, we propose a novel opportunistic routing approach ML-SOR (Multi-layer Social Network based Routing) which extracts social network information from such a model to perform routing decisions. To select an effective forwarding node, ML-SOR measures the forwarding capability of a node when compared to an encountered node in terms of node centrality, tie strength and link prediction. Trace driven simulations show that a routing metric combining social information extracted from multiple social network layers allows users to achieve good routing performance with low overhead cost.

1. Introduction

The pervasive use of mobile phones and the social networking applications available on these devices have attracted particular interest in recent years, especially in the research area of infrastructure-less network architectures exploiting peer-to-peer opportunistic connectivity and social relations for content dissemination. In a world where individuals are becoming increasingly reliant on mobile communication in several aspects of their life, being unable to communicate can negatively affect both business and personal relationships. Consequently, when there is no suitable network architecture, an alternative system is necessary. Delay Tolerant Networks (DTNs) [1–3] were designed to allow communication in challenged scenarios where a fixed network infrastructure is not available, nodes often create sparse network topologies and the contacts between them are

* Corresponding author at: DIMES Cubo 41 C, Ponte P. Bucci, University of Calabria, 87036 Arcavacata di Rende (CS), Italy. Tel.: +39 0984494802.

E-mail addresses: socievolea@dimes.unical.it (A. Socievole), eiko.yoneki@cl.cam.ac.uk (E. Yoneki), derango@dimes.unical.it (F. De Rango), jon.crowcroft@cl.cam.ac.uk (J. Crowcroft).

http://dx.doi.org/10.1016/j.comnet.2015.02.016
1389-1286/

intermittent. DTNs use a *store-carry-forward* paradigm where the mobile node first stores the message, carries it while moving, and then forwards it to an intermediate node or to the destination. A network that routes packets using this approach is also called *opportunistic network* [4], because nodes forward messages during an encounter opportunity.

Many works on opportunistic routing focus on the best way to select the optimal relay node considering real-world social interactions to optimize message delivery. Studying the social relationships between individuals within the network, it is possible to better understand which encounters are useful to minimize unnecessary message forwarding. Commonly, the social relationships are extracted from Bluetooth, ZigBee or Wi-Fi encounters between mobile devices [5–10]. Protocols based on the social network *detected* through encounters, however, take time to build up a knowledge database from which to take routing decisions. Contact information changes constantly and it may produce suboptimal paths since it takes time to build the complete social behavior of each network node. Online social network ties (e.g. Facebook, Twitter, MySpace, and LinkedIn), on the contrary, are explicitly declared and represent more stable relationships. Also nodes' interests that are usually self-declared online represent a social dimension useful to build the social behavior of a node. For this reason, we believe that by designing routing metrics that combine the available offline social information with online social information, the chances of the message reaching its destination are maximized since the routing scheme has a "multi-layer knowledge" of a relay's social behavior [11,12].

The aim of this paper is to demonstrate that the use of online and offline social features extracted from multiple social networks is able to improve opportunistic routing. Although several forwarding schemes [13–19] using both online social network and detected social network information exist, we propose an approach exploiting more than two social network layers, previously introduced in [20]. In this paper, we start defining a multi-layer social network model combining the temporal social network detected through wireless encounters and other types of static social networks, and investigating the relationship between social network layers in terms of node centrality, community structure, tie strength and link prediction. Our knowledge of the structural differences between different social networks for the same set of individuals is limited. This is partly due to the difficulty in collecting human mobility data that record, simultaneously, individuals' movements and their social interactions at different layers. For this reason, we analyze the data of two experiments including mobility, online social network and interests of the participants that enable us to better understand participants' behavior at different social network layers. The aim of this analysis is twofold: firstly, it provides novel insights into the comparability of social networks and secondly, it is useful for understanding social dynamics on a multi-layer complex network which can be exploited for message forwarding. Then, we demonstrate that multi-layer social networks can improve opportunistic forwarding by proposing a Multi-Layer Social network based Opportunistic Routing (ML-SOR) protocol, in which a node forwards packets using a routing metric that combines social information extracted from multiple social network layers. We show the effectiveness of ML-SOR by comparing it to Epidemic routing [21], PRoPHET [22], Bubble Rap [5], H-Bubble Rap, a hybrid version of Bubble Rap computing centrality on a multi-layer social network, and PeopleRank [15]. Extensive simulations on the two experimental datasets show that ML-SOR can achieve message delivery ratio similar to Epidemic routing with significantly lower overhead cost.

The remainder of this paper is organized as follows. Section 2 provides a review of existing works on multi-layer social networks and on social-based routing protocols for opportunistic networks exploiting several types of social networks to drive routing decisions. Section 3 formally describes the multi-layer social network model adopted in our work. Section 4 describes the datasets used to test individuals' behavior at different social network layers and discusses the results of multi-layer social network analysis. Section 5 presents ML-SOR, our multi-layer social network based routing proposal. Section 6 describes how ML-SOR performance evaluation is organized. Section 7 presents the results of performance evaluation. Finally, Section 8 discusses the results and Section 9 concludes the paper.

2. Related work

In recent years, several social-based routing algorithms have been proposed for opportunistic networks. The plain fact that these networks are basically human-centered and follow the way humans come into contact, has led researchers to use concepts of Social Network Analysis (SNA) [23] like *community* detection [24,25] or *centrality* analysis [26–28] in the design of more efficient routing protocols. Most social-based routing protocols exploiting community structure and/or centrality (i.e., the contribution of network position to the importance of an individual in the network) compute these metrics on the social graph detected through real-world contacts between mobile devices. We refer to this proximity graph as *detected social network* (DSN). Only few works drive routing decision also using social information extracted from virtual or self-declared contacts. We call the social network based on these kind of contacts *online social network* (OSN). In the following sections, we describe the main social-based forwarding schemes for opportunistic networks, by dividing them into two categories; using only DSN or using both DSN and OSN for extracting social routing metrics. Since we consider more than two social network layers in our work, we also review the main works on multi-layer social networks.

2.1. Multi-layer social networks

In the real world, more than one kind of connections can exist between any pair of individuals. As a matter of fact, for this type of social networks there is not a unique word identifying them. Terms as *multi-layer network*, *multi-relational network*, *multidimensional network* and *multiplex network* are considered synonyms [29].

Bródka et al. [30] define a multi-layer social network as a set of single-layered social graphs where each graph has a unified and fixed set of nodes, and only the set of edges between nodes may vary. A similar model is proposed by Magnani and Rossi [31], where a *pillar multi-network* allowing different node sets for each layer and adding a node mapping function between layers is defined. In this work, the authors propose also a *ML-model* where many nodes from one social network layer can match to a single node in another layer.

A great effort has been made into the definition of multi-layer social metrics that consider all the different social dimensions. In [32], Hao et al. propose a measure of the influence of one layer on the other layers. Bródka et al. [33] focus on the shortest path problem proposing two approaches for the computation of shortest paths in multi-layer social networks. In [30], the same authors investigate the neighborhoods in multi-layer social networks by defining and analyzing cross-layer clustering coefficient, cross-layer degree centrality and various versions of multi-layer degree centrality.

2.2. Opportunistic forwarding using only DSN

Forwarding schemes belonging to this category commonly extract social information from Bluetooth, ZigBee or Wi-Fi interactions between mobile devices whose ubiquity permits the collection of user co-presence information and the identification of social ties grounded on real-world interactions. In [34], a large number of traces related to different human mobility environments are analyzed, finding that their inter-contact time distribution is heavy-tailed. Consequently, routing algorithms for opportunistic networks have to be tested under different mobility models than the random waypoint (RWP).

Bubble Rap [5] is a social-based protocol exploiting both community and centrality computed on DSN. It uses two centrality values that are associated to each node based on the node global popularity in the whole network and local popularity within its community or communities. The forwarding scheme uses these centrality values so that a message is transferred to nodes with higher global centrality values until the carrier node meets a node with the same community label as the destination node. In this case, the message is forwarded to nodes with higher local rankings until successful delivery.

In *Habit* [6], Mashhadi et al. define a social-based routing scheme where messages are forwarded to nodes that are interested in the specific content of the message. Messages are disseminated in a selection-based manner by taking into account a node's physical proximity and its social ties. A *regularity graph* is used to keep trace of when and how often two nodes come into contact, and an *interest graph* is used to build dissemination paths based on nodes interested in particular data. However, the data useful to build the interest graph is only disseminated to close neighbors belonging to the regularity graph.

The *SimBetTS* [7] routing protocol is another example of DSN-based protocol where a node forwards a message to an encountered node according to three social metrics: *betweenness* (the number of shortest paths on which a node lies), *similarity* (the number of ties that two nodes share), and *tie-strength* (the recency, duration and number of contacts between two nodes). During an encounter, the nodes exchange their lists of encountered nodes in order to locally calculate the betweenness utility, the tie strength utility and the similarity utility. Each node then examines the messages it is carrying and computes the overall utility value of each message destination. Messages are then forwarded to the node holding the highest overall utility for the message destination node.

In [8,9], the popularity of a node in DSN is used as routing criterion. In the former, popular nodes (called *hubs)* are those connected with many nodes in DSN and are characterized analyzing the history of encounters. In the latter, both the popularity of a node in DSN and the contact durations are used to perform a destination-unaware forwarding strategy. The interesting aspect of this work is that centrality is not computed based on the aggregated network contact graph but takes into account the dynamics of node mobility.

2.3. Opportunistic forwarding using both DSN and OSN

Routing schemes based on DSN examine encounters between mobile devices in order to optimize routing by forwarding messages to nodes which are encountered more often. However, social metrics computed on DSN, may miss some strong social ties between nodes, since it takes time to reconstruct a consistent social behavior for each node from an intermittent contact network. In such situations, OSN helps to better identify the social behavior of nodes and consequently, to improve the construction of forwarding paths.

In [13], Mtibaa et al. propose a Bluetooth-based mobile social network application deployed among a group of participants during a computer communication conference and show that the structure of the social graph constructed on self-declared friends helps to build forwarding paths in the contact graph, allowing two nodes to communicate over time using opportunistic contacts and intermediate nodes.

In *MobiClique* [14], Pietiläinen et al. leverage DSN and OSN so that users can move between them in a way that enhances both. MobiClique bootstraps the network using the users' Facebook profiles consisting of a unique user identifier, the friendlist and a list of groups (or networks) consisting of users sharing some common interests. During an encounter, if the two users are friends or share some interests, they are alerted and can choose to exchange data. In a similar work [16], Bigwood and Henderson present an opportunistic routing protocol, called *Social Role Routing* (SRR), that uses OSN information to bootstrap the opportunistic network. SRR exploits social network analysis technique of regular equivalence [35] that partitions nodes into classes in order to categorize nodes into roles. During the bootstrap phase of the network, each node stores a copy of a *role connectivity graph*, which has been previously computed using the OSN of the participating nodes, allowing them to compute the geodesic distance between roles. Message are then forwarded only to encountered nodes that are in the same role, or in a role adjacent to the destination's role.

PeopleRank [15] uses OSN information in a different way. The OSN graph (called *social graph*) edges can represent a friendship relationship or shared interests between a node pair. This information is used by the routing scheme to compute node rankings. This protocol is similar to the PageRank algorithm [36] used by Google search engine to measure the relative importance of a Web page within a set of pages. When two neighbor nodes in the OSN graph meet, they exchange two pieces of information: their current PeopleRank values describing their sociality and the number of OSN neighbors they have. Messages are then forwarded towards nodes having a higher PeopleRank value.

In SPRINT [18], OSN information is combined with contact history and predictions of future encounters. These aspects are used by each SPRINT node to compute utility values for its messages and perform social-based routing accordingly. Message utility is computed considering the freshness and the number of hops of the message, the delivery probability and the popularity of the carrier node, the future meeting, the social connection and the time spent between the carrier and the destination.

3. Multi-layer social network model

Physical encounters described through DSN are the oldest form of interaction which still plays a fundamental role in human sociality. However, today's social relationships are maintained through several layers of interactions, such as chats, emails, phone calls and online social networking websites. DSN and OSN described in the previous section represent two different social contexts. If we extend the number of social contexts and consider several social networks for a particular set of users, we obtain a multi-layer structure, representing the connections of a single user to other users on several autonomous layers. Two users might be connected on many layers at the same time – e.g. two users may be connected through Bluetooth network, Facebook, LinkedIn and Twitter networks – while other users may be connected on just one layer – e.g. like co-workers connected only through LinkedIn or friends only through Facebook. The result is a multidimensional complex structure where there are several social network layers and where users exploit different dimensions of sociality. In this paper, we define a multi-layer social network as in [30] and consider weighted graphs, where edge weights can be used to represent the strength of the relationship, similarly to [31].

Definition 1 (*Social Network Layer*). A social network layer L is a weighted graph $G\langle V, E\rangle$ with vertex set V corresponding to users on the social network and edge set $E \subseteq V \times V$ corresponding to social links between users.

Definition 2 (*Multi-Layer Social Network*). A multi-layer social network $MLSN = (L_1, L_2, \ldots, L_n)$ is a tuple where $L_i = G_i\langle V, E_i\rangle, i \in 1, \ldots, n$ are social network layers.

An example of multi-layer social network is shown in Fig. 1. As can be seen, the vertexes of the set {A,B,C,D,E,F,G,H,I} are connected on n different social network layers. On each layer, the set of vertexes has a particular connectivity pattern (set of directed edges) which is related to the social context representing that layer. Vertex C and vertex D, for example, have a unique connection on layer 2. The study of such a network is useful to understand the overall social behavior of users. By comparing node centrality, communities or other structural measures computed on the multi-layer network, it is possible to understand how much the single networks are complementary to each other or have a similar social function. In the following section, we study two particular multi-layer social networks extracted from experiments performed during scientific conferences and we present the results of the multi-layer social network analysis.

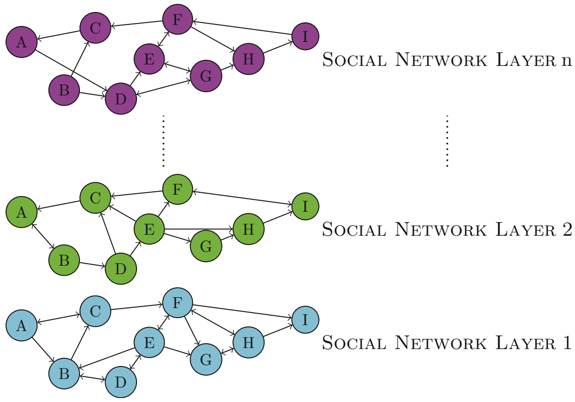

Fig. 1. A multi-layer social network.

4. Multi-layer social network analysis

In this preliminary study, we aim to illustrate how nodes behave at different social network layers. We start describing the datasets used to perform the analysis and the methodology adopted to create a multi-layer social network from temporal and static social networks. Then, we analyze some structural properties of the multi-layer social graph. The key contributions of this study can be summarized as follows:

- propose a novel methodology to create a multi-layer social network graph when the available social information is in the form of both temporal (e.g. DSN) and static social graphs (e.g. OSN);
- provide novel insights into the comparability of social networks;
- provide results about the relationship between social network layers which support our intuition that multi-layer social networks can be exploited for opportunistic routing.

4.1. Datasets

For our analysis, we use *Lapland* [37] and *Sigcomm* [38] datasets. Lapland dataset was collected during the ExtremeCom09 workshop in Padjelanta National Park (Sweden) and contains Bluetooth co-location data of 17

Table 1
Summary of the two datasets used for evaluation.

	Lapland	Sigcomm
DSN type	Bluetooth	Bluetooth
Radio range	10 m	[10–20] m
# of devices	17	76
Device type	iMote	Phone
Trace duration	399812 s	320593 s
Granularity	[120–600] s	120 ± 10 s
OSN type	Facebook	Facebook
Interests type	scientific	Facebook

conference attendees logged during 4 consecutive days. Participants were asked to carry iMotes with them detecting devices in proximity range (approximately 10 meters). In addition to mobility information, the dataset contains participants' Facebook friendlists and interests in terms of scientific topics.[1] Similarly to Lapland dataset, Sigcomm dataset was collected during a conference, the SIGCOMM 2009 conference held in Barcelona (Spain). This dataset, available from the CRAWDAD project [39], includes Bluetooth co-location data collected by the opportunistic mobile social application MobiClique and the social profiles (Facebook friends and interests) of 76 conference attendees.

In both datasets, the experimental devices logged all Bluetooth contacts between the nodes participating to the experiment using a periodic scanning every t seconds, where t is the experiment's granularity. These two experimental traces are representative of two different social environments in terms of node mobility. Moreover, considering OSN information, each Lapland mobile node has a Facebook profile while in Sigcomm, a small subset of mobile nodes does not have Facebook information (probably for privacy issues or simply because they do not use Facebook). On the contrary, considering interests, both in Lapland and in Sigcomm each node has at least one interest. Table 1 summarizes their main characteristics.

Fig. 2 summarizes the characteristics of Bluetooth co-location data in terms of contact duration and total number of contacts distributions. As can be observed, they follow both an approximate power law in each dataset. By looking at the complementary CDF of contact durations, 52% of Lapland contact durations last more than one hour, while only 4% last more than 3 h. In Sigcomm dataset, contact durations are shorter: only 5% of contact durations last more than 1 h. By looking at the number of contacts in Lapland dataset, we observe that 50% of the number of contacts is greater than 26, and 15% is greater than 50. In Sigcomm dataset, on the contrary, the number of contact opportunities between node pairs is significantly lower. Only 10% of the number of contacts is greater than 10. As far as the correlation between contact durations and the number of contacts is concerned, in Lapland dataset we found that contact duration is positively correlated to the number of contacts with a correlation coefficient of 0.991. For Sigcomm dataset, we found a correlation value of 0.621.

[1] The scientific topics of conference attendees have been extracted from their publications available online. Using each paper's keywords listed after the abstract, we produced a list of scientific interests for each participant.

4.2. Social network layers modeling

Based on the above social information, we extract different undirected social graphs that are used to model the multi-layer social network for each dataset. We use the participants' Facebook social network information to generate a *Facebook network* social graph, where an edge between two nodes exists if they are friends, and the participants' interests to generate an *Interest network* social graph where an edge between two nodes i and j measures the similarity $Sim(i,j)$ between them. Here, we use the Jaccard coefficient as similarity measure:

$$Sim(i,j) = \frac{|I_i \cap I_j|}{|I_i \cup I_j|} \tag{1}$$

where I_i and I_j are the sets of interests of node i and node j, respectively.

As far as Bluetooth co-presence data are concerned, we need to form a social graph as in the above cases. However, the modeling of a social graph from a time-varying structure is more complex, since contact patterns may change radically over time. The most obvious example is the day-time pattern: many contacts during the day and few at night. Considering that the DSN is a temporal graph while the other network layers (Facebook and Interest) are static graphs, we choose to model a multi-layer social network using only static graphs in order to simplify the comparison between layers. In that way, we are able to easily compare metrics belonging only to static graphs thus avoiding the comparison between temporal and static graph metrics. To this end, we choose to use a Joint Diagonalization (JD) technique [40] that is able to decompose the behavior, in times, of DSN in order to create average static graphs for each time. Each of these static graphs, called *mode*, is a representation of the most common propagation paths corresponding to a particular time interval. JD has been successfully used in different areas to track the evolution of systems via their eigenvectors and the application to SNA is quite recent. Given M samples of a network $\{A_1, A_2, \ldots, A_M\}$, JD produces an average matrix $\bar{A}$ of the samples. Specifically, it seeks an orthogonal matrix such that:

$$A_i = UC_iU^T \quad \forall i \tag{2}$$

If U corresponds to the eigenvectors of A_i then C_i is diagonal, however no matrix U exists where all C_i are diagonal (except for the trivial case in which all A_i are equal). Since JD aims at finding an average matrix representative of all samples, it seeks an average orthogonal matrix $\bar{U}$ which diagonalizes the given matrices A_i as much as possible. In particular, it seeks a matrix $\bar{U}$ such that the sum of the squares of off diagonal elements of $\sum_{i=1}^{M} C_i$ are minimized:

$$\bar{U} = \underset{U}{\operatorname{argmin}}\ off_2\left(\sum_{i=1}^{M} C_i\right) \tag{3}$$

where off_2 is the sum of the squares of off diagonal elements, called the *deviation* of A_i from $\bar{A}$, δ_i:

$$\delta_i = off_2(C_i) = \sum_{k \neq j} |C_i^{k,j}|^2 \tag{4}$$

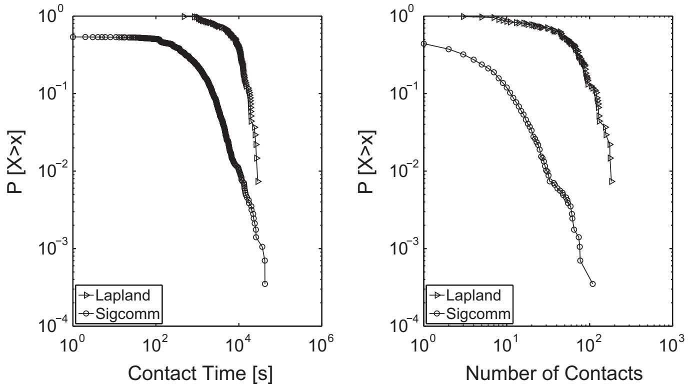

Fig. 2. Complementary CDFs of contact durations and the number of contacts.

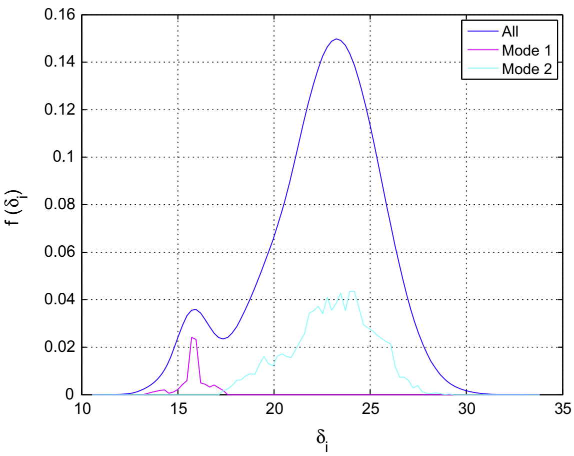

Fig. 3. Distribution of δ_i for Lapland Bluetooth contacts.

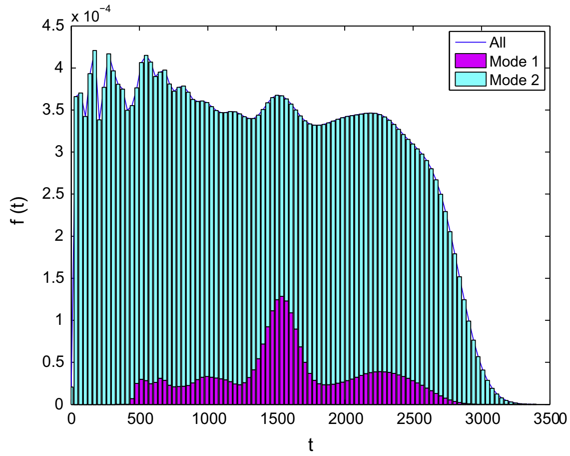

Fig. 4. Distribution of times for each network mode extracted from Lapland Bluetooth contacts.

where $C_i^{k,j}$ is the kth row and jth column of C_i. Given $\bar{U}$, an average sampling graph may be constructed as:

$$\bar{A} = \bar{U}\bar{C}\bar{U}^T \tag{5}$$

where $\bar{A}$ is a matrix in which each entry is the average weight of the link as observed by the samples in the network (in a least square sense) and $\bar{C}$ is the average of diagonals of A_i projected onto $\bar{U}$.

For each dataset's DSN, we generated 10,000 spanning trees as samples, starting from a random node with the messages starting at random times (uniformly distributed). Then, these trees were combined using JD in order to create an average sampling matrix $\bar{A}$. By examining the distribution of deviations from $\bar{A}$, δ_i (with $i = 1, \ldots, 10,000$), we found that the distribution is multi-modal both in Lapland and in Sigcomm datasets (Figs. 3 and 5). Two modes of operation, extracted through a Gaussian mixture model [41], summarize the most frequent propagation paths on the corresponding DSNs. Figs. 4 and 6 show the distribution of the sample start times. Lapland DSN has different modes of operation at different times. Mode 1 covers part of the times with low frequency values (i.e. network pattern occurring few times), while Mode 2 is the predominant one, being the first mode to occur and covering all the times with high frequency values. Similarly, Sigcomm DSN has Mode 2 as the predominant one. However, differently from Lapland DSN, Sigcomm Mode 1 covers a shorter time window with high frequency values (i.e. network pattern occurring many times in a limited time period).

Based on the social graphs extracted from Bluetooth co-presence data, Facebook friendlists and shared interests, Lapland and Sigcomm multi-layer social graphs will be composed by 4 layers: (1) DSN Mode 1 ($M1$), (2) DSN Mode 2 ($M2$), (3) Facebook network (FB), and (4) Interest network (Int). The structural analysis of these multi-layer networks will be presented in the following sections.

4.3. Comparison of node centrality

The identification of which nodes are more central than others is one of the most important tasks in social network analysis. The aim of this analysis on a multi-layer network is to understand how a particular centrality measure varies for a given node in each network. For each node and at

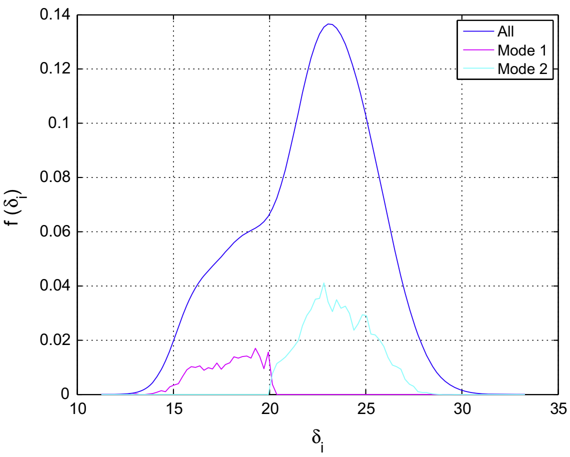

Fig. 5. Distribution of δ_i for Sigcomm Bluetooth contacts.

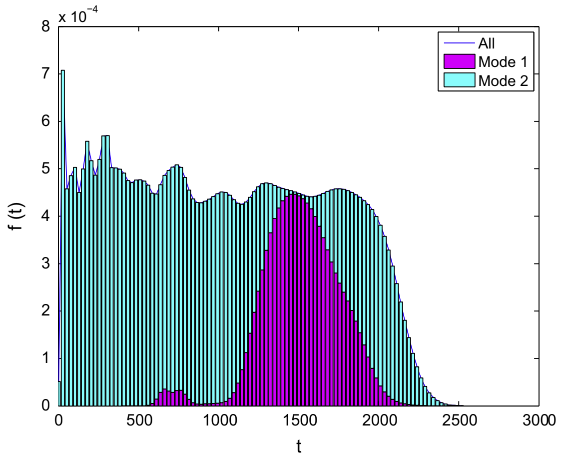

Fig. 6. Distribution of times for each network mode extracted from Sigcomm Bluetooth contacts.

each network layer we computed some fundamental centrality measures: degree, ego betweenness, closeness and eigenvector centrality. Degree centrality counts how many connections a node has and can be considered the most basic of all centrality measures. Ego betweenness [42] is another important measure quantifying the influence a node has over the flow of information between every pair of nodes in the network graph under the assumption that information flows over the shortest path between them. This measure is computed using just the one-hop adjacency matrix of a node, as opposed to the global adjacency matrix used for classical betweenness [27]. Closeness centrality [23] emphasizes the distance of a node to all others in the network. With this measure, it is possible to identify the nodes that could reach others quickly. Eigenvector entrality [43] is a particular centrality measure defined in a circular manner. The centrality of a node is proportional to the sum of the centrality values of all its neighbors. In other words, an important node can be characterized by its links to other important nodes.

In Table 2, the correlation between the values of a particular centrality measure computed among two different layers of Lapland dataset is shown. $\rho_{M1,M2}$ represents, for example, the correlation between a centrality measure in Mode 1 network and the same centrality measure in Mode 2 network. We note that, in general, there is no high correlation between centrality measures of the same type at different layers, except for closeness between Mode 2 network and Facebook network with a medium correlation value of 0.42. Looking at Sigcomm dataset results in Table 3, we find again low correlation values, except for Mode 2 network and Interest network that show medium correlation values on most centrality values (degree, closeness and eigenvector centrality) thus demonstrating that they are more similar. We conclude that even if there are some cases in which some layers show medium–low similarity in terms of a centrality metric, the results for both datasets show that in most cases the centrality of a node on Facebook network and Interest network layers do not predict its centrality on DSN.

4.4. Similarity among communities

In this section, we focus on groups by analyzing the communities at each network layer. We consider a set of community detection algorithms that are able to detect both non-overlapping and overlapping communities. The following summarizes their features.

- *Fiedler* clustering [44] is a spectral method splitting the graph into two disjoint communities. The eigenvector for the second smallest eigenvalue of a Laplacian matrix is called Fiedler vector and can be used for decomposing graphs into structural components.
- *Louvain* method [45] partitions the graph in disjoint communities and is based on a greedy optimization technique that attempts to optimize the modularity [46] of a partition of the network. As a first step, the method looks for small communities by optimizing modularity locally. As a second step, it aggregates nodes belonging to the same community and builds a new network whose nodes are the communities. These steps are repeated iteratively until a maximum of modularity is attained and a hierarchy of communities is produced.
- *n-CLIQUE* algorithm [47] analyzes the overlapping structure of communities. It finds all the maximal complete subgraphs such that the distance of each pair of nodes is not larger than n.
- *k-CLIQUE* or Clique Percolation Method (CPM) [48] is another approach finding overlapping communities where a community is defined as the union of all k-cliques (complete subgraphs with k nodes) that can reach each other through a series of adjacent k-cliques, where two k-cliques are said to be adjacent if they share $k-1$ nodes.

Since we are interested in measuring the similarity between communities belonging to different network layers, we use the *normalized mutual information* [49] measure. Given two networks A and B, the normalized mutual information is defined as follows:

Table 2
Correlation between centrality measures computed between couple of social network layers (Lapland dataset).

	$\rho_{M1,M2}$	$\rho_{M1,FB}$	$\rho_{M2,FB}$	$\rho_{M1,Int}$	$\rho_{M2,Int}$	$\rho_{FB,Int}$
Degree	−0.318	0.155	0.113	−0.12	0.358	0.244
Ego betweenness	−0.453	0.104	−0.106	−0.073	0.082	0.191
Closeness	−0.347	0.169	0.42	−0.212	0.272	−0.213
Eigenvector	−0.276	0.137	0.076	−0.206	0.392	0.213

Table 3
Correlation between centrality measures computed between couple of social network layers (Sigcomm dataset).

	$\rho_{M1,M2}$	$\rho_{M1,FB}$	$\rho_{M2,FB}$	$\rho_{M1,Int}$	$\rho_{M2,Int}$	$\rho_{FB,Int}$
Degree	−0.302	0.145	0.103	−0.111	0.343	0.233
Ego betweenness	−0.435	0.099	−0.086	−0.065	0.069	0.18
Closeness	−0.332	0.158	0.263	−0.199	0.418	−0.209
Eigenvector	−0.265	0.122	0.069	−0.195	0.379	0.196

$$NMI(A,B) = \frac{-2\sum_{i=1}^{c_A}\sum_{j=1}^{c_B} N_{ij}\log\left(\frac{N_{ij}N}{N_{i.}N_{.j}}\right)}{\sum_{i=1}^{c_A} N_{i.}\log\left(\frac{N_{i.}}{N}\right) + \sum_{j=1}^{c_B} N_{.j}\log\left(\frac{N_{.j}}{N}\right)} \tag{6}$$

where c_A is the number of communities in network A, c_B is the number of communities in network B, N_{ij} is the number of nodes in the intersection between community i from network A and community j from network B, N is the total number of nodes, and $N_{i.}$ and $N_{.j}$ are the number of nodes in community i of network A and community j of network B, respectively. $NMI(A,B)$ ranges between 0 and 1, where different communities have a mutual information of 0 and identical communities have a mutual information of 1. In [50], an extended version of this measure was defined for overlapping communities.

The results of the similarity values for Lapland and Sigcomm datasets are shown in Tables 4 and 5, respectively. For both datasets, we considered $n = 3$ with a minimum size of cliques $c\_size_{min} = 2$ for n-CLIQUE and $k = 3$ for CPM. In Lapland networks, all the algorithms are characterized by medium–low similarity values between communities belonging to different network layers, except for n-CLIQUE and CPM detecting the Mode 2 network and the Interest network as two identical communities. An interesting result is that all the algorithms, even if producing different community sets, have a similarity value computed between Mode 1 network communities and Facebook network communities which can be considered significative. In Sigcomm dataset, we also find medium–low similarity values. In addition, we observe that all algorithms show that the most similar communities are those related to Mode 2 network and Interest network. Considering the results obtained for both datasets, we conclude that there is no high similarity between layers in terms of communities, except for the most frequent network pattern in DSN and Facebook network resulting slightly similar.

Table 4
Similarity between Lapland communities belonging to different social network layers.

Communities	$NMI_{Fiedler}$	$NMI_{Louvain}$	$NMI_{n-CLIQUE}$	NMI_{CPM}
C_{M1}, C_{M2}	0.185	0.17	0	0
C_{M1}, C_{FB}	0.174	0.229	0.179	0.398
C_{M1}, C_{INT}	0.028	0.228	0	0
C_{M2}, C_{FB}	0.075	0.11	0	0
C_{M2}, C_{INT}	0.091	0.025	1	1
C_{FB}, C_{INT}	0.108	0.208	0	0

Table 5
Similarity between Sigcomm communities belonging to different social network layers.

Communities	$NMI_{Fiedler}$	$NMI_{Louvain}$	$NMI_{n-CLIQUE}$	NMI_{CPM}
C_{M1}, C_{M2}	0.103	0.098	0.032	0.018
C_{M1}, C_{FB}	0.127	0.119	0.012	0.039
C_{M1}, C_{INT}	0.123	0.028	0.021	0.235
C_{M2}, C_{FB}	0.199	0.359	0.188	0.298
C_{M2}, C_{INT}	0.097	0.224	0	0
C_{FB}, C_{INT}	0.158	0.323	0.037	0.215

4.5. Strong ties

Motivated by recent studies [51–54] demonstrating that mobile nodes encounter other online socially-connected nodes or nodes with common interests with high probability, we now focus on analyzing the strong ties on Lapland and Sigcomm DSN layers in order to find a matching, if any, with Facebook network and Interest network links. In this work, we consider as strong ties the links between DSN nodes having a high number of contacts [55] and hence, frequent interactions through which the transfer of information may arise. If we find a good matching between the DSN strong ties and the links on other social layers, we could evaluate if a DSN link is a strong ties just considering the presence/absence of links on other social layers and thus avoiding the computation of contact frequency. To this end, we computed for each node pair the total number of contacts had both in Mode 1 network and in Mode 2 network. Then, we computed the percentage of matchings between strong ties and Facebook friendships (Table 6), and between strong ties and Interest network links (Table 7). In particular, we computed the top-20 strong ties (i.e. the first 20 node pairs ordered for decreasing number of contacts) and similarly, the top-40 strong ties. Table 6 shows that a high percentage of strong ties both for Mode 1 network and Mode 2 network correspond to Facebook friendships in both datasets, especially when considering the strongest ties corresponding to the top-20 node pairs. Comparing the two DSN network modes, we observe that there is a better matching between Mode 2 network strong ties and Facebook links. This means that online social ties are a good indicator of the strong ties in the predominant DSN mode. Comparing the two datasets, we can further note that Lapland dataset is characterized by higher matching percentages. Analyzing Table 7, the lower values of matchings lead us to conclude that online social ties are better than shared interests for identifying strong ties.

4.6. Link prediction

We finally analyze the multi-layer social network from another point of view that will be useful for designing efficient opportunistic forwarding rules. The tie strength analyzed in the previous section evaluates already existing connections and could be exploited by a routing scheme for evaluating the future availability of an existing link. Note that due to the intermittent connectivity, an existing link to a central node, for example, could not be available. In this case, the tie strength measure will be useful for having an indication about links having a high probability to be activated. In this section, we study the usefulness of a multi-layer structure for predicting likely future links that do not already exist. In particular, our aim is to evaluate if additional social layers like Facebook network and Interest network layers are able to predict new links in the DSN.

Some previous studies explored the theory of measures based on common neighbors for predicting future links in co-authorship networks [56], networks of users' home pages in the World Wide Web [57] and human contact networks [58]. These studies assign a connection weight score to pairs of nodes based on their common neighbors and then produce a ranked list in decreasing order of score. They basically compute a measure of similarity between node pairs, relative to network topology in order to predict future links. The more two nodes are similar in terms of neighbors, the more they will be likely to have a future link. In co-authorship networks, for example, if two authors have many colleagues in common, they are more likely to come into contact themselves.

Our link prediction method follows the aforementioned approach. We form the graph $G'_{DSN}[t_0,t_1]$ where an edge between two nodes exists if they had at least one contact during the first day of the experiment. We refer to $[t_0,t_1]$ as the *training interval*. Similarly, we form $G''_{DSN}[t_2,t_{end}]$ as the graph containing the contacts had from the second day to the end of the experiment and we refer to $[t_2,t_{end}]$ as the *test interval*. Note that for this analysis we do not use Mode 1 network and Mode 2 network since we are interested in the temporal evolution of DSN. The link prediction algorithm we apply accesses (1) the network $G'_{DSN}[t_0,t_1]$, (2) the Facebook network graph denoted with G_{FB} and (3) the Interest network graph denoted with G_{Int} to output a list of edges not present in $G'_{DSN}[t_0,t_1]$ that are predicted to appear in $G''_{DSN}[t_2,t_{end}]$. We denote the generic training graph by $G_{train}\langle V,E_{old}\rangle$ and the graph $G'_{DSN}[t_2,t_{end}]$ on the test interval as $G_{test}\langle V,E_{new}\rangle$. E_{new} are the new interactions we are seeking to predict. Note that we test the prediction capabilities of Facebook network and Interest network graphs compared to the training graph $G_{DSN}[t_0,t_1]$. Hence, we consider as training graphs also Facebook network and Interest network graphs.

Table 6
Percentage of strong ties corresponding to Facebook network links.

	Lapland M1	Lapland M2	Sigcomm M1	Sigcomm M2
% of matchings (20 pairs)	70	80	65	70
% of matchings (40 pairs)	62.5	67.5	60	62.5

Table 7
Percentage of strong ties corresponding to Interest network links.

	Lapland M1	Lapland M2	Sigcomm M1	Sigcomm M2
% of matchings (20 pairs)	40	45	35	40
% of matchings (40 pairs)	50	52.5	40	40

Table 8
Link prediction performance of different training graphs. The values specify the factor improvement over random prediction.

	Lapland	Sigcomm
G'_{DSN}	43.4	38.8
G_{FB}	21.3	17.1
G_{Int}	41.6	31.5

The method used for link prediction is the Jaccard coefficient. This link predictor measures the similarity between nodes i and j belonging to G_{train} as

$$LP(i,j)=\frac{|N_i\cap N_j|}{|N_i\cup N_j|} \tag{7}$$

where N_i and N_j are the set of i's neighbors and j's neighbors, respectively. The link prediction algorithm outputs a ranked list L_P of pairs in $V\times V-E_{old}$; these are predicted new links in decreasing order of confidence. Using the ranked list, we determine the size of the intersection between this set of pairs and E_{new}.

To represent the predictor quality, we compute the factor improvement over a *random predictor* which simply predicts randomly selected pairs not present in the training interval. Table 8 shows the results obtained for the different training graphs. The Interest network layer is able to predict links with an accuracy comparable to that of the DSN graph representing the first day of contacts. On the contrary, Facebook network, even if outperforming the random predictor, has a performance sensibly lower than the Interest network. This experiment demonstrates that the interests shared between nodes create relationships similar to those in the DSN, thus allowing to predict links with the same accuracy of the prediction using part of DSN data. Similarly to co-authorship networks where common neighbor measures well predict future collaborations

[56], we found that scientific interests shared between nodes are able to accurately predict future encounters.

5. Opportunistic routing with multiple social network layers

The analysis performed in the previous section can be useful for designing forwarding rules that construct efficient paths based on relationships at different social network layers. Although limited to Lapland and Sigcomm datasets, our results mostly show low correlation between the values of a given centrality measure computed on different social network layers. However, we also found that Mode 2 network and Interest network layers show medium correlation values for closeness with respect to Lapland dataset, and for degree, closeness and eigenvector centrality with respect to Sigcomm dataset. Analyzing communities, the results mostly show medium–low similarity between communities belonging to different social network layers. However, we also found that for n-CLIQUE and CPM algorithms there is a perfect similarity between Mode 2 network and Interest network communities.

The above relationship between layers assessed through different types of structural analyses can be exploited by an opportunistic routing scheme for having a proper view of users' social dynamics when available information about a social dimension is partial. Because of the temporal dynamics of contacts among users, it takes time to infer the corresponding social behavior of nodes. DSN strong ties, for example, take time to be identified using the history of encounters. If the routing scheme has a partial view of a node's sociality given by part of DSN data, it may produce sub-optimal forwarding paths. Having assessed that Facebook ties are able to predict DSN strong ties, we can exploit Facebook network layer together with DSN layer in the proposed routing strategy. In addition, we can also exploit the Interest network layer that has shown to predict future DSN links with a factor improvement over random prediction comparable to that of the DSN graph representing the first day of contacts. The extraction of more complete social information to be used to identify the nodes that are best suited to forward information, aims at maximizing the chance of the message to reach its destination.

In this section, we present ML-SOR, a multi-layer social network based opportunistic routing scheme. Simulating real mobility traces, we will show that social information extracted from a multi-layer social network is able to improve opportunistic routing.

5.1. ML-SOR social metric

ML-SOR is based on a social metric which exploits three social dimensions: proximity, online friendships and interests. ML-SOR social metric is computed using a combination of three measures:

- **centrality** on DSN layer
- **tie strength** on OSN layer(s)
- **link predictor** on Interest network layer

Table 9
List of symbols used to define ML-SOR social metric.

List of symbols	
i	ith node
j	jth node
d	Destination node
t	tth time slot
T	Number of time slots
N	Number of neighbors of node i
M	Number of neighbors of node j
l	lth OSN layer
L	Number of OSN layers
e	Encounter event
C_{Degree}	Degree centrality
$C_{CDegree}$	Cumulative degree centrality
TS	Online tie strength
TS_{TOT}	Total online tie strength
LP	Link predictor (common neighbors)
CS	Centrality utility score
TSS	Tie strength utility score
LPS	Link predictor utility score
MLS	ML-SOR utility score

Table 9 lists the symbols used to define the social metric.

We consider centrality as one of the most important factors to choose a good message relay. In graph theory and network analysis, centrality quantifies the structural importance of a vertex within the graph. A central node has usually a stronger capability of connecting other network nodes. We therefore compute centrality at the DSN layer, where the corresponding social graph is leveraged through encounters between mobile devices. ML-SOR social metric computes node centrality for a node i, $C_{CDegree}(i)$, using a long-term cumulative estimate of degree centrality. Degree centrality basically quantifies the number of connections a node has. The advantage in using this measure is that it can be easily computed locally considering only a node's ego network. More specifically, ML-SOR computes the number of unique nodes seen throughout a specific time slot and then average this measure with a set of previous measures. Degree centrality for a node i during a time slot t is computed as follows:

$$C_{Degree}(i,t) = \sum_{j=1}^{N} e(i,j,t) \tag{8}$$

where

$$e(i,j,t) = \begin{cases} 1 & \text{if } i \text{ encounters } j \text{ during time slot } t \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

represents an edge between node i and a node j on the DSN graph corresponding to the time slot considered,[2] and N is the number of nodes in i's range. The cumulative degree, $C_{CDegree}(i)$, is then computed by averaging the node's degree values over a set of T time slots including the most recent time slot and all the previous ones:

$$C_{CDegree}(i) = \frac{1}{T}\sum_{t=0}^{T} C_{Degree}(i, T-t) \tag{10}$$

[2] Considering that the DSN graph is a temporal graph, we form a static graph for each time slot by amalgamating all contacts in that time interval.

In that way, ML-SOR provides a fully decentralized approximation for a node's degree centrality, which is easy to be computed.

Centrality described above is measured using the history of contacts and does not consider future links availability. Considering that the links in the network are time-varying, an existing link to a central node may not be highly available. We therefore include a *tie strength* indicator into ML-SOR social metric. This indicator is able to identify the links that have a higher probability to be activated and is measured by considering online social ties between the individuals carrying the mobile devices. This choice is driven by the consideration that a social tie between two users on online social networking websites, such as Facebook, Twitter[3] or LinkedIn, typically do not change over time. In Facebook, for example, "intermittent" friendships are highly improbable. Moreover, such friendships have been shown to be a good indicator of DSN strong ties in the previous section. Consequently, online ties can be considered a good measure of whether a link on DSN will be activated. ML-SOR calculates tie strength between node i and node j at OSN layer l as:

$$TS(i,j,l) = \begin{cases} 1 & \text{if } i \text{ and } j \text{ are connected at } layer\ l \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

The total tie strength between two nodes is the sum of the indicators measured at each OSN layer:

$$TS_{TOT}(i,j) = \sum_{l=1}^{L} TS(i,j,l) \tag{12}$$

where L is the total number of online social networking websites considered. Here, we hypothesize that more OSN layers are able to improve ML-SOR's knowledge about strong ties. Two users, for example, may not be friends on Facebook but be linked on Twitter thus having an online tie indicating a DSN strong tie on another OSN layer.

ML-SOR social metric takes into account a third measure useful to predict future encounters between two nodes. While tie strength can be used to indicate the future availability of an existing link in the contact network, a *link predictor* predicts likely future new connections. Here, the link predictor is computed on Interest network layer, where a link between two nodes exists if they have at least one interest in common. As shown in the previous section, examining common neighbors of a pair of nodes on Interest network layer, we can predict future encounters to which the transfer of information may arise. ML-SOR computes the link predictor $LP(i,j)$ of a possible future collaboration between node i and node j as a common neighbor measure based on Jaccard coefficient:

$$LP(i,j) = \frac{|N \cap M|}{|N \cup M|} \tag{13}$$

where M is the number of nodes in j's range.

For each measure, ML-SOR determines the utility score of node i for delivering a message to node d compared to node j as follows:

$$CS(i,j) = \frac{C_{CDegree}(i)}{C_{CDegree}(i) + C_{CDegree}(j)} \tag{14}$$

$$TSS(i,j,d) = \frac{TS_{TOT}(i,d)}{TS_{TOT}(i,d) + TS_{TOT}(j,d)} \tag{15}$$

$$LPS(i,j,d) = \frac{LP(i,d)}{LP(i,d) + LP(j,d)} \tag{16}$$

The ML-SOR social metric is given by the combination of the contributing score values as follows:

$$MLS(i,j,d) = CS(i,j)[1 + TSS(i,j,d) + LPS(i,j,d)] \tag{17}$$

As can be observed, MLS captures the relay significance of a node when compared to an encountered node across all social network layers, in terms of centrality, tie strength and link predictor. Note also that node centrality is considered as the predominant factor in message forwarding. Both tie strength and tie predictor utility scores are weighted with centrality utility score and then added to centrality utility score. In that way, tie strength and link predictor utility scores will reflect the centrality utility score (e.g. high, low or medium) between the sender node and the encountered node.

5.2. ML-SOR forwarding strategy

The forwarding process in ML-SOR is given by Algorithm 1. During a contact event between two nodes, these nodes exchange their centrality values, one or more lists of online social contacts (one list for each online social networking website) and a list of nodes with common interests. Each node then examines the messages it is carrying and computes the *MLS* social metric of each message destination. Messages are then forwarded to the encountered node if this node has a higher *MLS* value for the message destination node or if it is the destination. As can be observed, the two nodes only have to exchange few informations to be able to compute the ML-SOR metrics in a distributed fashion.

Algorithm 1. ML-SOR forwarding algorithm.

```
procedure ENCOUNTERNODE(j)
  exchangeCentralityValues()
  exchangeOnlineContactsLists()
  exchangeInterestNodeList()
  for every message m in message_buffer do
    d ← m.destination()
    myMLS ← computeMLScore()
    encounterMLS ← computePeerMLScore()
    if encounterMLS ≥ myMLS || j = d then
      forwardMessage(m,j)
    end if
  end for
end procedure
```

6. Evaluation

The performance of ML-SOR are evaluated through simulations carried out on the Opportunistic Network

[3] Here we consider a tie between a user A and a user B, if A follows B and vice versa.

Environment (ONE) simulator [59], which is a specific tool for simulating mobile opportunistic networks. First, we focus on the analysis of ML-SOR social metric by showing the benefit of combining utility scores computed on several layers and the effect of using different centrality measures denoting user's importance within the DSN layer. Then, we compare ML-SOR to other well-known opportunistic routing schemes. For all these tests, we consider Lapland and Sigcomm traces described in Section 4. For both datasets we consider a multi-layer social network formed by the following layers: Bluetooth contact network (DSN layer), Facebook network (OSN layer) and Interest network.

6.1. Centrality measures in comparison

ML-SOR uses a social metric which is based on user degree centrality computed on DSN layer. Node centrality is just one of the measures accounting for user position/importance in social networks. The existing centrality measures can be divided based on the way in which they are computed in three main groups: degree, shortest paths and rank [60]. We therefore test on ML-SOR social metric two other user position measures, ego betweenness centrality and *node position* [60–62], representative of the shortest paths and rank groups, respectively. Even if there are other measures representative of these groups (e.g. closeness centrality for shortest-paths group), we choose the measures that we consider easy to implement in a distributed way.

As described in Section 4.3, ego betweenness centrality measures the influence a node has over the information flow between every pair of nodes in the ego network graph under the assumption that information flows over the shortest path between them. This centrality metric can be computed efficiently in a distributed way since only local information is required at each node. If A is the adjacency matrix for the ego network, $A^2[1-A]_{i,j}$ gives the number of shortest paths (geodesics of length 2) joining i to j and the sum of the reciprocal of the entries gives the ego betweenness.

Node position is a type of eigenvector centrality taking into account the position of the other nodes. In particular, node position for a node x considers both the value of node positions of its neighbors as well as their activity in relation to x:

$$NP(x) = (1-\epsilon) + \epsilon(NP(y_1)C(y_1 \rightarrow x) + \cdots + NP(y_m)C(y_m \rightarrow x)) \tag{18}$$

where ϵ is a constant coefficient from the range $[0,1]$ denoting the influence the neighborhood has on x, $\{y_1, y_2, \ldots, y_m\}$ are the m neighbors of x, and $C(y \rightarrow x)$ is a *commitment function* from the range [0, 1] reflecting the strength of the relationship between y and x. In the simulations, we compute the commitment function as the ratio between the number of connections from y to x and the total number of connections from y to all its encountered nodes. The initial values of node position were established to 1 for all nodes, while for ϵ we considered three levels of neighborhood influence: $\epsilon = 0.1$ (low), $\epsilon = 0.5$ (medium) and $\epsilon = 0.9$ (high).

6.2. Routing algorithms in comparison

There are many forwarding methods in the literature for opportunistic networks but we cannot compare ML-SOR with all of them, hence we choose those that we consider most relevant. We test three benchmark algorithms, Epidemic routing [21], PRoPHET [22] and Bubble Rap, together with two forwarding schemes using both DSN and OSN, *H-Bubble Rap* that we designed as a hybrid version of Bubble Rap and PeopleRank.

Epidemic routing can be considered a reference for other routing methods, since it determines an upper bound for message delivery and a lower bound for end-to-end delay. This method is characterized by a flooding-based strategy for which when two node encounter, they exchange all of their messages. In such way, messages spread like viruses by pairwise contacts between two nodes.

PRoPHET is another well known protocol in opportunistic networks and it is commonly used, as Epidemic, in comparisons due to its contact-based nature and good routing performance. It is a probabilistic routing method that calculates a metric, named *delivery predictability*, based on contact histories. A node that is carrying a message, relays it only to a node with higher deliver predictability.

As previously described, Bubble Rap is a social-based routing method which exploits node centrality and communities as routing metrics. We choose this method to compare ML-SOR to another social-based forwarding strategy where centrality is identified as the metric with a dominant impact on routing and for having a comparison with a well-known social protocol using only a social layer. We also implemented a hybrid version of Bubble Rap, H-Bubble Rap, in order to obtain another social-based protocol extracting social information from a multi-layer social network. H-Bubble Rap and Bubble Rap forwarding algorithms are the same. The only difference is that Bubble Rap local centrality and global centrality metrics are replaced with *MLS* metric computed with local $C_{CDegree}$ and *MLS* metric computed with global $C_{CDegree}$, respectively.

Aside from H-Bubble Rap, we compare ML-SOR to another protocol using both DSN and OSN. We choose *PeopleRank* since it uses a social graph where a link between two nodes is present either if they share interests or are online friends. Thus, it uses three social dimensions as ML-SOR.

6.3. Performance metrics

We use the following metrics to compare these algorithms: **delivery ratio** (the ratio of the number of delivered packets to the number of all packets), **overhead cost** (the number of packets transmitted across the air divided by the number of unique packets created), **average latency** (the average time it takes a packet to be delivered) and **average hop count** (the average number of hops a packet requires to reach destination). In the results, we plot all

Table 10
Values for the simulation parameters.

	Parameter	Value
Network	Buffer size	2000 MB
	Message[a] size	1 kB
	Inter-message creation interval	1800 s
PRoPHET	P_{init}	0.75
	β	0.25
	γ	0.98
Bubble Rap	C-Window duration	6 h
	C-Window # of windows	5
	K (K-Clique)	5
PeopleRank	Damping factor	0.8
ML-SOR	Time slot	6 h
	T	5

[a] Each message is exchanged between randomly selected source–destination pairs.

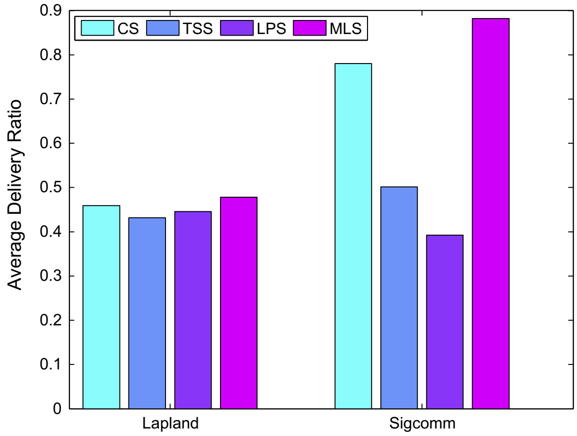

Fig. 7. Routing performance of ML-SOR utility scores.

these metrics as a function of TTL: the maximum time a message can stay in the system after its creation. TTL has been chosen since it is a fundamental parameter for studying the ability of a routing protocol to find the necessary number of relays within a certain time. In Table 10, we specify the common simulation parameters for all the simulations.

7. Simulation results

In this section, we discuss the results obtained after performing the simulations. The first experiment evaluates the average delivery ratio of each utility score composing ML-SOR social metric. The second experiment evaluates ML-SOR average delivery ratio when other centrality measures are used for computing the utility score defined on DSN layer. The third and the fourth experiments illustrate ML-SOR routing behavior compared to the other routing algorithms in Lapland and Sigcomm datasets, respectively, having different characteristics in number of nodes, duration, mobility patterns and social dynamics.

7.1. Evaluation of each ML-SOR utility score

In the first experiment, we evaluate the routing performance of each ML-SOR utility score and the benefit of combining the three utility scores in order to improve message delivery. Fig. 7 shows the average delivery ratio for Lapland and Sigcomm datasets after simulating several scenarios with different TTL values. As can be observed, when evaluating centrality (CS), tie strength (TSS) and link predictor (LPS) utility scores, forwarding based on DSN centrality is characterized by the highest average delivery ratio both in Lapland and in Sigcomm. This result confirms the importance of node centrality in message forwarding and hence, in finding good next hops. However, TSS and LPS show good delivery performance considering that are metrics that are not based on physical encounters and only on virtual ties, especially in Lapland dataset. As observed in Section 4, both tie strength computed on Facebook network layer and link prediction computed on Interest network give better results in Lapland datatset than in Sigcomm dataset. The advantage in using TSS and LPS is more clear when they are combined with CS. We can note that the overall delivery ratio increases, especially in Sigcomm dataset where ML-SOR achieves 88% of delivery ratio while routing based only on CS achieves only 78% of delivery ratio. In Lapland dataset, on the contrary, the performance improvement is lower since the higher duration of the trace and the higher link density cause the generation of more messages within the network, so a higher delivery ratio is harder to achieve with devices having a limited data memory. We observed that nodes' buffers are often full thus causing many message drops and consequently, a lower delivery performance.

7.2. Evaluation of different centrality measures on DSN

The goal of the second experiment is to evaluate centrality measures different from degree centrality in ML-SOR social metric. In particular, we compare ML-SOR delivery performance to two variants: $ML-SOR_{EBC}$ and $ML-SOR_{NP}$ using ego betweenness centrality and node position, respectively. Similarly to the previous experiment, we evaluate the average delivery ratio in both datasets after simulating several scenarios with different TTL values. Analyzing ego betweenness, Fig. 8 shows that in Lapland dataset, $ML-SOR_{EBC}$ achieves average delivery ratio slightly lower than ML-SOR based on degree centrality (45% vs. 47%). In Sigcomm dataset, on the contrary, ego betweenness has the lowest average delivery ratio, achieving only 61% of delivery. Considering that in Lapland dataset the average diameter of DSN is significantly smaller than Sigcomm DSN diameter, ego betweenness, which is based on shortest paths of length 2, is more effective in the smaller network. As far as node position is concerned, $ML-SOR_{NP}$ with $\epsilon = 0.9$ performs the best compared to the other two coefficients measuring neighborhood influence ($\epsilon = 0.1$ and $\epsilon = 0.5$) in Lapland dataset. It achieves 46% of average delivery ratio. Thus, node position results in a routing performance very similar to $ML-SOR_{EBC}$ and ML-SOR. In Sigcomm dataset, where the network is larger, we observe that $ML-SOR_{NP}$ is more influenced by the choice of the ϵ value. Here, setting a medium value for the neighborhood influence, we obtain a percentage of average delivery ratio comparable to ML-SOR (86% vs. 88%). In general, we conclude that rank-based node position performs better than ego betweenness and for some parameter settings is able to achieve average delivery ratio comparable to ML-SOR ratio. Consequently, node position can be considered a good candidate as user importance/centrality metric in opportunistic routing. However, the initial choice of the parameter ϵ in node position influences delivery performance, thus making degree centrality, which is simpler and easier to compute, a preferable choice.

7.3. Comparison between ML-SOR and other routing schemes

Lapland dataset. By analyzing delivery ratio showed in Fig. 9, we observe that all algorithms deliver more packets to the destinations when the TTL increases. However, as the TTL becomes high the increment in the delivery ratio is marginal, since the capacity of the network to forward packets becomes the performance bottleneck. Epidemic routing outperforms all the other protocols with the highest delivery ratio, achieving 61% of message delivery. We can observe that its overhead cost having a value of 15 on average is also very high because of the large amount of message replicas injected into the network. That is why an opportunistic protocol with a high delivery capability, as in the case of Epidemic routing, but with a lower cost would be the right choice in order to save energy. As can be seen, PRoPHET is a good candidate, since it reduces overhead cost, with a delivery ratio slightly lower than Epidemic routing. Moreover, PRoPHET outperforms all social-based protocols in terms of message delivery. Adding probabilities to the decision making, as in the case of PRoPHET, works better than social information in this mobility scenario. In terms of overhead cost (Fig. 10), however, PRoPHET costs much more than social-based protocols. On the contrary, multi-layer social information included by ML-SOR in the forwarding decision reduces notably overhead. ML-SOR shows the lowest overhead cost while maintaining a delivery ratio which is about 10% less than PRoPHET's delivery ratio. ML-SOR, even if outperforming the other social-based schemes in terms of message delivery, shows delivery ratios lower than Epidemic routing and PRoPHET. Analyzing the most frequent contact patterns in Lapland DSN, we conclude this difference with Epidemic routing and PRoPHET is due to the network contact patterns' dynamics that are not very suitable for social-based schemes in general. Lapland DSN network is often dense thus leading Epidemic routing to find quickly a path to destination. Also in PRoPHET, contact network dynamics lead to compute contact probabilities that result in effective routing paths. As far as the other social-based schemes are concerned, H-Bubble Rap is able to outperform PeopleRank both in terms of delivery ratio and delivery cost. This confirms that our social metric exploiting strong ties and link prediction is able to find routing paths better than PeopleRank that uses also friendships and interests within its routing metric. Observing Bubble Rap's delivery ratio and overhead cost, we conclude that protocols exploiting multiple social

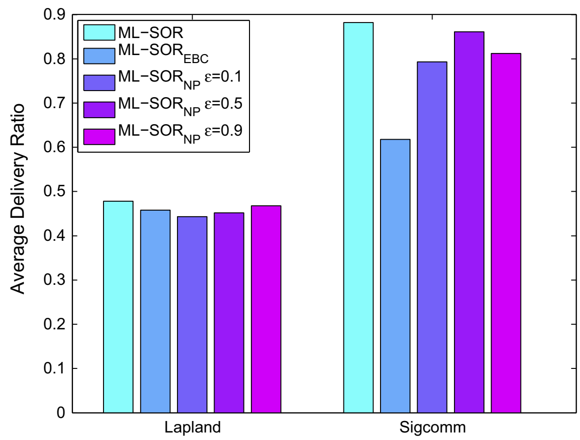

Fig. 8. ML-SOR routing performance for different centrality measures on DSN layer.

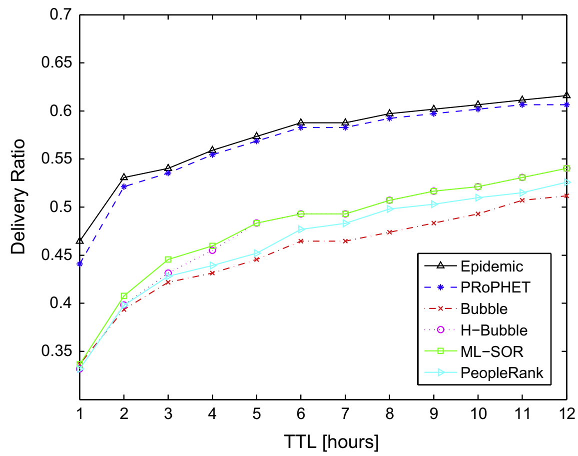

Fig. 9. Delivery ratio as a function of message TTL (Lapland dataset).

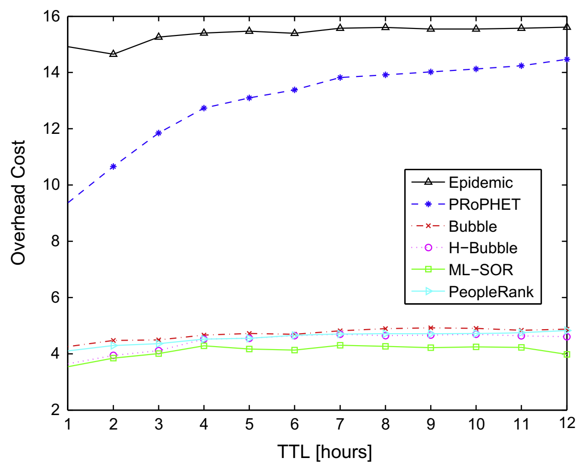

Fig. 10. Overhead cost as a function of message TTL (Lapland dataset).

network information are able to select more efficient paths in terms of delivery ratio and overhead ratio.

By looking at average latency in Fig. 11, for low TTLs (1 h and 2 h) all protocols show a similar average latency. As TTL is increased, Epidemic routing and PRoPHET are able to deliver messages faster than the other protocols. The reason is that they replicate more packets than the other algorithms, as can be seen from their overhead costs, thus reducing delivery delay. H-Bubble Rap, on the contrary, even if it transmits more messages than ML-SOR, has an higher average latency for each TTL value, while PeopleRank performs slightly better. However, both H-Bubble Rap and PeopleRank are outperformed by ML-SOR showing a lower delivery ratio. As expected, Bubble Rap, due to the higher number of messages transmitted, is able to reduce the end-to-end delay with respect to the other social-based schemes.

Fig. 12 compares the algorithms in terms of average hop count. This metric is interesting to analyze since it reveals the social distance between sources and destinations. Epidemic routing has the highest average hop count with a value that is around 2.7. PRoPHET shows a lower hop count, with an average value of around 2.3, while social-based routing protocols such as Bubble Rap, H-Bubble Rap and ML-SOR have lower values since they wait for the right relays to forward messages. As can be observed, H-Bubble Rap and ML-SOR deliver messages by using paths with an average number of hops lower than PeopleRank and Bubble Rap. These results confirm that forwarding strategies which exploit multiple social contexts, and in particular the schemes exploring tie strength and link prediction, are able to reach the destination of the message within less hops.

Sigcomm dataset. Fig. 13 shows the delivery ratio for the second dataset. When compared to Lapland dataset, Sigcomm overall delivery ratio is higher, with values achieving more than 95% of message delivery. Since the number of nodes is higher, the possibilities of forwarding and delivery are higher as well. Moreover, the network contact pattern that we observed from Sigcomm modes are able to better balance network traffic avoiding the overloading of message buffers that may result, as in the case of Lapland dataset, in lower delivery performance. Among all algorithms, Epidemic routing has again the highest delivery ratio. However, for most TTLs, ML-SOR and H-Bubble Rap perform the same as Epidemic routing. Differently from Lapland dataset, the protocols based on the ML-SOR metric are able to achieve Epidemic routing delivery and this is due to Sigcomm DSN modes that are less dense and more socially-structured thus resulting in better forwarding paths. We can further observe that within this dataset, PeopleRank and PRoPHET perform worse, even if for high TTLs (10, 11 and 12 h) they achieve Epidemic routing performance. Once again, as for Lapland dataset, Bubble Rap shows the lowest delivery ratio, achieving around 83% of message delivery. This means that ML-SOR social metric is able to improve the performance of Bubble Rap, both in the case of ML-SOR which does not consider communities to drive routing decisions and of H-Bubble Rap which is community-based.

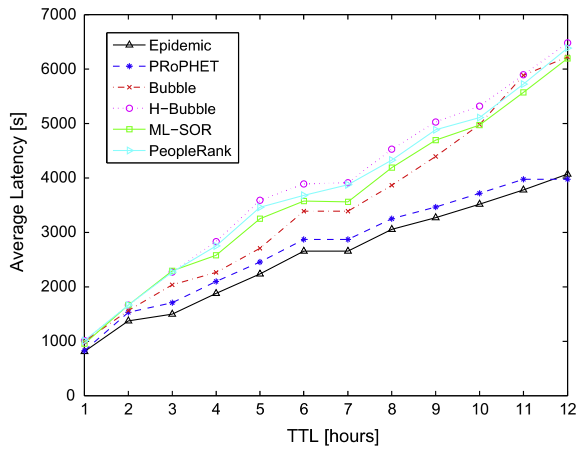

Fig. 11. Average latency as a function of message TTL (Lapland dataset).

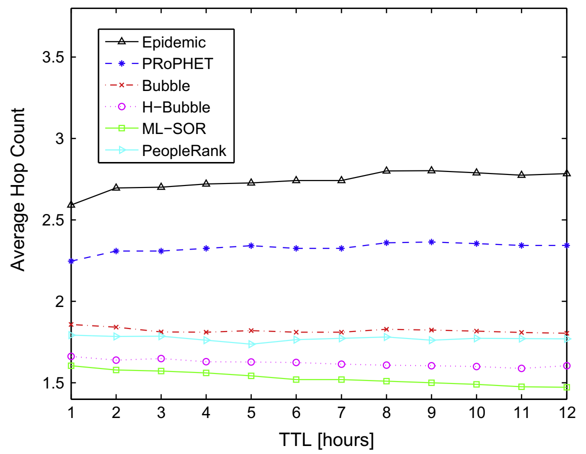

Fig. 12. Average hop count as a function of message TTL (Lapland dataset).

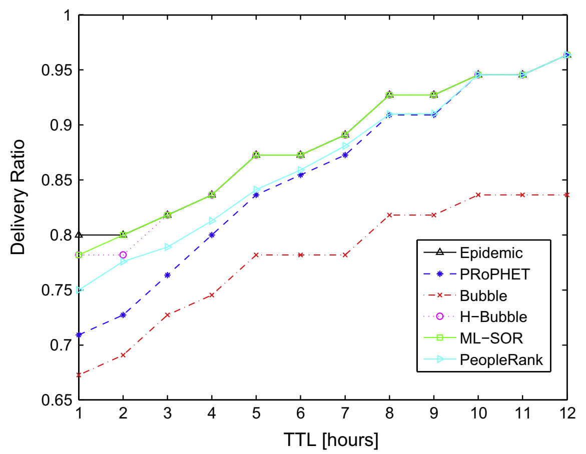

Fig. 13. Delivery ratio as a function of message TTL (Sigcomm dataset).

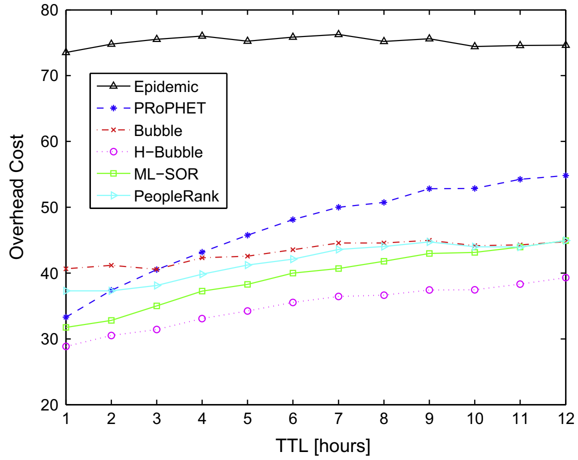

Fig. 14. Overhead cost as a function of message TTL (Sigcomm dataset).

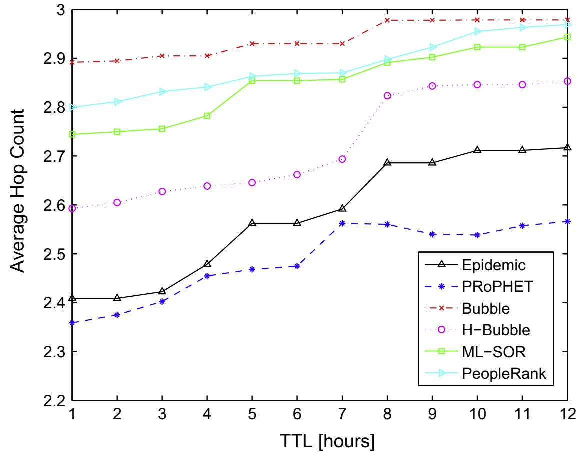

Fig. 16. Hop count as a function of message TTL (Sigcomm dataset).

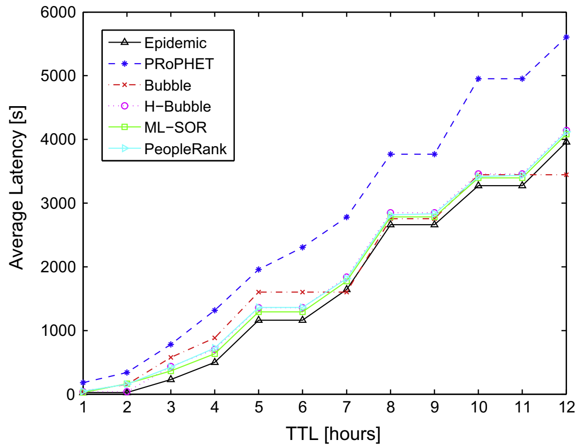

Fig. 15. Average latency as a function of message TTL (Sigcomm dataset).

In terms of cost (Fig. 14), we can observe that among the algorithms considered Epidemic routing has the highest cost. Again, this is because Epidemic routing generates many message replicas. In contrast, the other five algorithms are more greedy in replication, especially the social-based schemes. ML-SOR and H-Bubble Rap confirm that the use of the multi-layer social metric not only gives better performance than Bubble Rap that uses only contact network social information and PeopleRank that uses multiple social dimensions, but is also able to find more effective relays than PRoPHET. We can further note that H-Bubble Rap has the lowest overhead cost. In Sigcomm scenario, where we found communities better structured than in Lapland dataset, we observe that the multi-layer social network metric combined with community detection is more effective than ML-SOR.

As we can see from average latency in Fig. 15, among all protocols Epidemic routing performs the best except for TTL set to 12 h where Bubble Rap performs better, while PRoPHET is characterized by the worst performance. PRoPHET behavior is completely different from Lapland dataset, where its average latency was similar to Epidemic routing. Here, also social-based schemes outperform PRoPHET. As for delivery ratio, Epidemic routing, H-Bubble Rap and ML-SOR performance is very similar. In particular, the multi-layer social metric of ML-SOR and H-Bubble Rap works quite well producing a latency lower than Bubble Rap and PeopleRank latency for most TTLs.

In Fig. 16 the average hop count is shown. Differently from Lapland dataset, social-based strategies show average hop counts higher than Epidemic routing and PRoPHET. However, H-Bubble Rap and ML-SOR outperform Bubble Rap and PeopleRank. As can be further observed, H-Bubble Rap shows an hop count lower than ML-SOR hop count. This result demonstrates, as perviously seen for overhead cost, that in this dataset the multi-layer social network metric combined with community detection is more effective.

8. Discussion

ML-SOR is a social-based algorithm exploiting multiple social networks to perform routing in mobile opportunistic networks. ML-SOR has been designed in a distributed manner and drives routing decisions exchanging only small amount of social information. Taking advantage of multiple social contexts, it reduces notably the number of transmissions and achieves a satisfactory delivery performance compared to Epidemic routing, especially in Sigcomm dataset. In Lapland dataset, even though Epidemic routing and PRoPHET have high delivery ratios, ML-SOR outperforms both these schemes and the other social-based schemes in terms of overhead cost. That is why maximizing delivery ratio is not necessarily an indication that a routing protocol performs better than a protocol that does not. Considering that energy consumption is a fundamental issue in opportunistic networks [63], the reduction of unnecessary message forwarding will conserve energy and hence improve network performance.

There are some important aspects considered in this work that need further investigation. Node centrality at DSN layer is one of these aspects. In ML-SOR, node centrality is considered as the predominant factor in message forwarding. In Section 7.2, we evaluated different centrality measures showing the effectiveness of degree centrality. However, we did not analize the effect of combining multiple centrality measures at DSN layer (e.g. two, three or more measures on DSN layer). We argue that if we choose a proper set of m centrality metrics, we expect to have more information about user importance useful for message forwarding. However, the m metrics should be chosen to be as independent or as orthogonal as possible, otherwise less information is reflected by dependent metrics. In general, since most of the centrality metrics are computed from the adjacency matrix of DSN, we may expect that most metrics are dependent.

9. Conclusions

In this paper, we have presented ML-SOR, a novel opportunistic routing protocol that uses a multi-layer social network to select nodes to act as message relays. First, we have proposed a methodology to build a multi-layer social network graph from a temporal DSN graph describing wireless contacts and other static social network graphs describing virtual relationships. By analyzing two experimental traces containing multiple social networks for the same set of users, we have found medium–low correlation between centrality metrics at different layers and medium–low similarity between communities belonging to different layers. On the contrary, Facebook links have shown to be good indicators of DSN strong ties and Interest network links good predictors of future DSN links. Then, we have evaluated the routing performance of ML-SOR. Our results, although limited to two datasets, describe the behavior of our protocol in two environments where node mobility and the amount of social information are different. We have compared ML-SOR to Epidemic routing, PRoPHET, Bubble Rap, H-Bubble Rap which has been defined as a hybrid version of Bubble Rap using the same metric of ML-SOR, and PeopleRank. Testing Lapland and Sigcomm datasets, we have shown that by combining multiple social information messages can be delivered with high probability while keeping overhead ratio very small. Additionally, in Sigcomm dataset, delivery performance may be achieved equal to Epidemic routing but with significantly reduced overhead cost. Our plans for future work include the analysis of other datasets with different connectivity patterns and social network layers such as Twitter or LinkedIn, in order to gain further validation of our forwarding scheme.

Acknowledgements

The research leading to these results has been partially funded by the European Union's FP7 Program under the project eCOUSIN (318398). Annalisa Socievole was partially supported by European Commission, Calabria Region and European Social Fund 2007/2013 – ARUE grant. The authors wish to thank Damien Fay, Jérôme Kunegis and the anonymous reviewers for their insightful suggestions.

References

[1] K. Fall, A delay-tolerant network architecture for challenged Internets, in: Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, SIGCOMM '03, ACM, New York, NY, USA, 2003, pp. 27–34, http://dx.doi.org/10.1145/863955.863960.

[2] V. Cerf, S. Burleigh, A. Hooke, L. Torgerson, R. Durst, K. Scott, K. Fall, H. Weiss, Delay-tolerant Networking Architecture, RFC4838, April 2007.

[3] A. Socievole, F. De Rango, C. Coscarella, Routing approaches and performance evaluation in delay tolerant networks, in: Wireless Telecommunications Symposium (WTS), 2011, 2011, pp. 1–6. http://dx.doi.org/10.1109/WTS.2011.5960870.

[4] L. Pelusi, A. Passarella, M. Conti, Opportunistic networking: data forwarding in disconnected mobile ad hoc networks, IEEE Commun. Mag. 44 (2006) 134–141.

[5] P. Hui, J. Crowcroft, E. Yoneki, Bubble rap: social-based forwarding in delay-tolerant networks, IEEE Trans. Mob. Comput. 10 (2011) 1576–1589.

[6] A. Mashhadi, S. Ben Mokhtar, L. Capra, Habit: leveraging human mobility and social network for efficient content dissemination in delay tolerant networks, in: IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks Workshops, 2009. WoWMoM 2009, 2009, pp. 1–6. http://dx.doi.org/10.1109/WOWMOM.2009.5282467.

[7] E. Daly, M. Haahr, Social network analysis for information flow in disconnected delay-tolerant MANETs, IEEE Trans. Mob. Comput. 8 (2009) 606–621.

[8] S. Ahmed, S.S. Kanhere, HUBCODE: message forwarding using hub-based network coding in delay tolerant networks, in: Proceedings of the 12th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, MSWiM '09, ACM, New York, NY, USA, 2009, pp. 288–296, http://dx.doi.org/10.1145/1641804.1641853.

[9] K. Xu, V. Li, J. Chung, Exploring centrality for message forwarding in opportunistic networks, in: Wireless Communications and Networking Conference (WCNC), 2010 IEEE, 2010, pp. 1–6. http://dx.doi.org/10.1109/WCNC.2010.5506553.

[10] F. De Rango, F. Monteverdi, Social and dynamic graph-based scalable routing protocol in a DTN network, in: International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS), 2012, IEEE, 2012, pp. 1–8.

[11] S. Gaito, G.P. Rossi, M. Zignani, Facencounter: bridging the gap between offline and online social networks, in: Proceedings of 8th International Conference on Signal Image Technology and Internet Based Systems (SITIS), 2012, pp. 768–775.

[12] A. Socievole, S. Marano, Exploring user sociocentric and egocentric behaviors in online and detected social networks, in: 2nd Baltic Congress on Future Internet Communications (BCFIC), 2012, pp. 140–147. http://dx.doi.org/10.1109/BCFIC.2012.6217994.

[13] A. Mtibaa, A. Chaintreau, J. LeBrun, E. Oliver, A.-K. Pietilainen, C. Diot, Are you moved by your social network application?, in: Proceedings of the First Workshop on Online Social Networks, WOSN '08, ACM, New York, NY, USA, 2008, pp 67–72, http://dx.doi.org/10.1145/1397735.1397751.

[14] A.-K. Pietiläinen, E. Oliver, J. LeBrun, G. Varghese, C. Diot, Mobiclique: middleware for mobile social networking, in: Proceedings of the 2nd ACM Workshop on Online Social Networks, WOSN '09, ACM, New York, NY, USA, 2009, pp. 49–54, http://dx.doi.org/10.1145/1592665.1592678.

[15] A. Mtibaa, M. May, C. Diot, M. Ammar, Peoplerank: social opportunistic forwarding, in: INFOCOM, 2010 Proceedings IEEE, 2010, pp. 1–5. http://dx.doi.org/10.1109/INFCOM.2010.5462261.

[16] G. Bigwood, T. Henderson, Bootstrapping opportunistic networks using social roles, in: 2011 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2011, pp. 1–6. http://dx.doi.org/10.1109/WoWMoM.2011.5986139.

[17] F. De Rango, A. Socievole, A. Scaglione, S. Marano, Novel activity-based metrics for efficient forwarding over online and detected social networks, in: 2013 9th International Wireless Communications and Mobile Computing Conference (IWCMC), IEEE, 2013, pp. 1–6.

[18] R.I. Ciobanu, C. Dobre, V. Cristea, SPRINT: social prediction-based opportunistic routing, in: Proceedings of 7th IEEE WoWMoM workshop on Autonomic and Opportunistic Communications, AOC 2013, Madrid, Spain, 2013, pp. 1161–1166.

[19] A. Socievole, F. De Rango, S. Marano, Face-to-face with Facebook friends: using online friendlists for routing in opportunistic

networks, in: IEEE 24th International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC), 2013, 2013a, pp. 2989–2994. http://dx.doi.org/10.1109/PIMRC.2013.6666659.
[20] A. Socievole, E. Yoneki, F. De Rango, J. Crowcroft, Opportunistic message routing using multi-layer social networks, in: Proceedings of the 2nd ACM Workshop on High Performance Mobile Opportunistic Systems, ACM, 2013, pp. 39–46.
[21] A. Vahdat, D. Becker, Epidemic Routing for Partially-Connected Ad hoc Networks, Technical Report, Duke University, 2000.
[22] A. Lindgren, A. Doria, O. Schelen, Probabilistic routing in intermittently connected networks, in: Service Assurance with Partial and Intermittent Resources, Springer, 2004, pp. 239–254.
[23] S. Wasserman, K. Faust, Social Network Analysis: Methods and Applications, Cambridge Univ. Press, 1994.
[24] M. Newman, Detecting community structure in networks, Eur. Phys. J. B – Condens. Matter Complex Syst. 38 (2004) 321–330.
[25] M.E.J. Newman, M. Girvan, Finding and evaluating community structure in networks, Phys. Rev. E 69 (2004) 026113-1–026113-15.
[26] G. Sabidussi, The centrality index of a graph, Psychometrika 31 (1966) 581–603.
[27] L.C. Freeman, A set of measures of centrality based on betweenness, Sociometry 40 (1977) 35–41.
[28] P. Bonacich, Power and centrality: a family of measures, Am. J. Sociol. 92 (1987) 1170–1182.
[29] P. Bródka, P. Kazienko, Multi-layered Social Networks, 2012, arXiv preprint arXiv:1212.2425.
[30] P. Bródka, P. Kazienko, K. Musiał, K. Skibicki, Analysis of neighbourhoods in multi-layered dynamic social networks, Int. J. Comput. Intell. Syst. 5 (2012) 582–596.
[31] M. Magnani, L. Rossi, The ml-model for multi-layer social networks, in: International Conference on Advances in Social Networks Analysis and Mining (ASONAM), 2011, IEEE, 2011, pp. 5–12.
[32] J. Hao, S. Cai, Q. He, Z. Liu, The interaction between multiplex community networks, Chaos (Woodbury, NY) 21 (2011) 016104.
[33] P. Bródka, P. Stawiak, P. Kazienko, Shortest path discovery in the multi-layered social network, in: International Conference on Advances in Social Networks Analysis and Mining (ASONAM), 2011, IEEE, 2011, pp. 497–501.
[34] 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 (2007) 606–620.
[35] D.R. White, K.P. Reitz, Graph and semigroup homomorphisms on networks of relations, Soc. Netw. 5 (1983) 193–234.
[36] S. Brin, L. Page, The anatomy of a large-scale hypertextual web search engine, Comput. Netw. ISDN Syst. 30 (1998) 107–117.
[37] E. Yoneki, F.B. Abdesslem, Finding a Data Blackhole in Bluetooth Scanning, ExtremeCom, 2009.
[38] A.-K. Pietilainen, C. Diot, CRAWDAD Data set thlab/sigcomm2009 (v. 2012-07-15), 2012. <http://crawdad.cs.dartmouth.edu/thlab/sigcomm2009>.
[39] J. Yeo, D. Kotz, T. Henderson, Crawdad: a community resource for archiving wireless data at dartmouth, SIGCOMM Comput. Commun. Rev. 36 (2006) 21–22.
[40] D. Fay, J. Kunegis, E. Yoneki, Centrality and mode detection in dynamic contact graphs; a joint diagonalisation approach, in: IEEE/ACM ASONAM, 2013.
[41] G. McLachlan, D. Peel, Finite Mixture Models, vol. 299, Wiley-Interscience, 2000.
[42] M. Everett, S.P. Borgatti, Ego network betweenness, Soc. Netw. 27 (2005) 31–38.
[43] P. Bonacich, Factoring and Weighting Approaches to Status Scores and Clique Identification, 1972. <http://ci.nii.ac.jp/naid/10016154506/en/>.
[44] M. Fiedler, A property of eigenvectors of nonnegative symmetric matrices and its application to graph theory, Czech. Math. J. 25 (1975) 619–633.
[45] V.D. Blondel, J.-L. Guillaume, R. Lambiotte, E. Lefebvre, Fast unfolding of communities in large networks, J. Stat. Mech: Theory Exp. 2008 (2008) 10008.
[46] M.E. Newman, Modularity and community structure in networks, Proc. Nat. Acad. Sci. 103 (2006) 8577–8582.
[47] C. Bron, J. Kerbosch, Algorithm 457: finding all cliques of an undirected graph, Commun. ACM 16 (1973) 575–577.
[48] G. Palla, I. Derényi, I. Farkas, T. Vicsek, Uncovering the overlapping community structure of complex networks in nature and society, Nature 435 (2005) 814–818.
[49] L. Danon, A. Diaz-Guilera, J. Duch, A. Arenas, Comparing community structure identification, J. Stat. Mech: Theory Exp. 2005 (2005) 09008.
[50] A. Lancichinetti, S. Fortunato, J. Kertész, Detecting the overlapping and hierarchical community structure in complex networks, New J. Phys. 11 (2009) 033015.
[51] R.I. Ciobanu, C. Dobre, V. Cristea, Social aspects to support opportunistic networks in an academic environment, Ad-hoc, Mobile, and Wireless Networks, Springer, 2012, pp. 69–82.
[52] R.-I. Ciobanu, R.-C. Marin, C. Dobre, V. Cristea, Interest-awareness in data dissemination for opportunistic networks, Ad Hoc Netw. (2014).
[53] A. Mei, G. Morabito, P. Santi, J. Stefa, Social-aware stateless routing in pocket switched networks, IEEE Trans. Parallel Distrib. Syst. (2014) 1.
[54] A. Socievole, F. De Rango, A. Caputo, Wireless contacts, Facebook friendships and interests: analysis of a multi-layer social network in an academic environment, in: IFIP/IEEE Wireless Days Conference, 2014.
[55] P.V. Marsden, K.E. Campbell, Measuring tie strength, Soc. Forces 63 (1984) 482–501.
[56] D. Liben-Nowell, J. Kleinberg, The link-prediction problem for social networks, J. Am. Soc. Inform. Sci. Technol. 58 (2007) 1019–1031.
[57] L.A. Adamic, E. Adar, Friends and neighbors on the web, Soc. Netw. 25 (2003) 211–230.
[58] A. Socievole, F. De Rango, S. Marano, Predicting links in human contact networks using online social proximity, in: Predicting Real World Behaviors from Virtual World Data, Springer, 2014, pp. 85–102.
[59] A. Keränen, J. Ott, T. Kärkkäinen, The ONE simulator for DTN protocol evaluation, in: Proceedings of the 2nd International Conference on Simulation Tools and Techniques, Simutools '09, ICST, Brussels, Belgium, Belgium, 2009, pp. 55:1–55:10. http://dx.doi.org/10.4108/ICST.SIMUTOOLS2009.5674.
[60] K. Musiał, P. Kazienko, P. Bródka, User position measures in social networks, in: Proceedings of the 3rd Workshop on Social Network Mining and Analysis, ACM, 2009, p. 6.
[61] P. Brodka, K. Musial, P. Kazienko, A performance of centrality calculation in social networks, in: International Conference on Computational Aspects of Social Networks, 2009. CASON'09, IEEE, 2009, pp. 24–31.
[62] P. Kazienko, K. Musiał, Mining personal social features in the community of email users, in: SOFSEM 2008: Theory and Practice of Computer Science, Springer, 2008, pp. 708–719.
[63] A. Socievole, S. Marano, Evaluating the impact of energy consumption on routing performance in delay tolerant networks, in: 8th InternationalWireless Communications and Mobile Computing Conference (IWCMC), 2012, pp. 481–486. http://dx.doi.org/10.1109/IWCMC.2012.6314251.

Annalisa Socievole is a Research Fellow in the Department of Informatics, Modeling, Electronics and Systems Engineering (DIMES) at University of Calabria (Italy). She received a PhD in Systems and Computer Science Engineering in February 2013 and a master's degree in Telecommunications Engineering in July 2009, both from the University of Calabria. From October 2011 to April 2012 she has been a visiting PhD student in the Systems Research Group of Cambridge Computer Laboratory (UK). From November 2013 to June 2014 she spent a Post-doc period in NAS Group at TU Delft (The Netherlands) to carry out the research project CONTACTO. Her research interests include Opportunistic Networks, DTNs and Complex Networks.

Eiko Yoneki is an EPSRC Research Fellow in the Systems Research Group of Cambridge Computer Laboratory in United Kingdom. She has received her PhD in Computer Science from the University of Cambridge on 'Data Centric Asynchronous Communication' in 2007. Her research interests span distributed systems, networking and databases, including complex networks and parallel data-flow programming. She holds several EPSRC and EU FP7 grants and the current research focus is the exploration of new abstractions for supporting the design and implementation of robust, secure, and heterogeneous large-scale graph data processing in distributed systems.

Floriano De Rango is an Assistant Professor in the Department of Informatics, Modeling, Electronics and Systems Engineering (DIMES) at University of Calabria (Italy). He received a degree in Computer Science Engineering in October 2000, and a PhD in Electronics and Communications Engineering in January 2005, both at University of Calabria. From January to October 2000 he worked in the Telecom Research LAB C.S.E.L.T. in Turin as visiting student. From March to November 2004 he was visiting researcher at the University of California, Los Angeles (UCLA). From November 2004 until September 2007 he was Research Fellow in the DEIS Department at University of Calabria. His interests include Satellite Networks, IP QoS architectures, Adaptive Wireless Networks, Ad Hoc Networks and Pervasive Computing.

Jon Crowcroft has been the Marconi Professor of Communications Systems in the Computer Laboratory since October 2001. He has worked in the area of Internet support for multimedia communications for over 30 years. Three main topics of interest have been scalable multicast routing, practical approaches to traffic management, and the design of deployable end-to-end protocols. Current active research areas are Opportunistic Communications, Social Networks, and techniques and algorithms to scale infrastructure-free mobile systems. He graduated in Physics from Trinity College, University of Cambridge in 1979, gained an MSc in Computing in 1981 and PhD in 1993, both from UCL. He is a Fellow the Royal Society, a Fellow of the ACM, a Fellow of the British Computer Society, a Fellow of the IET and the Royal Academy of Engineering and a Fellow of the IEEE.