

Fair Routing in Delay Tolerant Networks

Josep M. Pujol

Alberto Lopez Toledo

Pablo Rodriguez

Telefonica Research, Barcelona
e-mail: jmp@tid.es

Telefonica Research, Barcelona
e-mail: alopez@tid.es

Telefonica Research, Barcelona
e-mail: pablorr@tid.es

Abstract—The typical state-of-the-art routing algorithms for delay tolerant networks are based on best next hop hill-climbing heuristics in order to achieve throughput and efficiency. The combination of these heuristics and the social network structure leads the routing to direct most of the traffic through a small subset of *good* users. For instance, in the SimBet algorithm, the top 10% of users carry out 54% of all the forwards and 85% of all the handovers. This unfair load distribution is not sustainable as it can quickly deplete constraint resources in heavily utilized mobile devices (e.g. storage, battery, budget, etc.). Moreover, because a small number of users carry a significant amount of the traffic, the system is not robust to random failures and attacks.

To overcome these inefficiencies, this paper introduces FairRoute, a routing algorithm for delay tolerant networks inspired by the social processes of *perceived interaction strength*, where messages are preferably forwarded to users that have a stronger social relation with the target of the message; and *assortativity*, that limits the exchange of messages to those users with similar "social status". We compare the performance of FairRoute to the state-of-the-art algorithms by extensive simulations on the MIT reality mining dataset. The results show that our algorithm outperforms existing algorithms in the *de facto* benchmark of throughput vs. forwards. Furthermore, it distributes better the load; the top 10% carry out 26% of the forwards and 28% of the handovers without any loss in performance.

I. INTRODUCTION

Thanks to the increasing number of mobile devices with wireless capabilities [1], the possibility of communication without network infrastructure is becoming a reality. For example, pocket switched networks rely exclusively on users exchanging messages among themselves upon proximity encounters. Provided that a significant amount of traffic is composed of delay tolerant messages, this paradigm, known also as delay tolerant networks (DTN) [2], could contribute to significantly reduce infrastructure costs and increasing bandwidth by orders of magnitude [3].

Because of their potential benefits, there has been a significant body of work in DTNs, with emphasis on routing algorithms. Since the seminal Epidemic routing algorithm [4] was introduced, there has been a succession of algorithms aimed at maximizing throughput – the number of messages arriving to the destination – while minimizing the traffic overhead caused by the number of forwards. While Epidemic routing achieves the optimal throughput, it is very inefficient: multiple copies of a message are forwarded until the network is flooded with redundant messages. Although there have been

proposals to reduce the network overhead of flooding [5], most of the recent algorithms avoid message duplication. This single-copy strategy requires routing algorithms to implement a *next-best-hop* heuristic that forwards the messages to those nodes with a highest probability to deliver the message to its destination.

Some algorithms operate under the assumption that the network of contacts is known a priori [6]. However, most recent algorithms rely only on mobility [7], [8], [9], [10], [11]. The availability of good quality datasets with contact traces gathered from real human mobility patterns [12], [13], has spurred a breed of sophisticated routing algorithms that exploit the topological properties of social networks [14], [15] to compute the *next-best-hop* heuristics. For instance, SimBet [16] relies on social distance based on transitivity [17] and betweenness centrality [18]. Also, Bubble [19] takes advantage of the empirically observed community structure in social networks [20] to decide which heuristic is more likely to be effective depending on the social group of the destination.

II. MOTIVATION: FAIRNESS WHO?

Routing in DTNs requires the answer to two questions [21]: 1) how the best next hop is calculated; and 2) how the traffic load is distributed among users. Most of the existing work has focused on maximizing throughput and minimizing number of forwards in order to obtain efficient algorithms with high delivery ratios, but with the exception of [22], [23], [24], there has been no attention to scalability and reliability. However, the intrinsic nature of DTNs, based on both human interaction and mobile phones, makes those networks prone to failures.

Contact traces used by DTN routing algorithms are correlated to social networks [25], [14], and social networks are a particular case of complex networks [26], [27]. One of the characteristics of such networks is that they exhibit a *fat-tail* connectivity distribution, where few nodes have many connections whereas the majority have very few. Since messages are forwarded via contacts, it is inevitable that the most connected nodes carry the majority of the traffic, hence producing an unfair load distribution. This is true even if traffic is routed at random.

Unfortunately for DTNs, fairness can only worsen if heuristics are applied to the forwarding decisions as we will show next. Let us assume a forward algorithm that chooses the *next-best-hop* randomly. Then the message forwarding process is a

random walk over a network with the normalized adjacency matrix A . The probability of going from node i to node j is A_{ij} that is the inverse of i 's connectivity degree under the random routing assumption. On the other hand, j 's probability of receiving a message is $\frac{\sum_i A_{ij}}{\sum_j \sum_i A_{ij}}$. It is trivial to see that nodes with higher connectivity have a higher probability of receiving a message. The probability of finding a message in node j , which is equivalent to its traffic load, can be calculated as the first eigenvector of P^T , where $P_{ij} = A_{ij}(\sum_j A_{ij})^{-1}$ is the stochastic matrix corresponding to A . The eigen vector for adjacency matrices of heterogeneous connectivity distributions will be skewed towards the most connected nodes. Thus, the *natural* unbalance of the load distribution in complex networks appears. Furthermore, if the forwarding is not random, but informed by a heuristic that biases towards connectivity, the probability of the highly connected nodes to receive messages is inevitably increased, and therefore, the load distribution to become even more unbalanced.

In systems that display a similar complex network connectivity such as the Internet, road networks or airline traffic; the unbalance problem is solved by upgrading the resources of the bottleneck nodes, e.g., by installing faster switches, constructing more 8-lane highways or expanding the airports with more terminals. However, this solution cannot be implemented system-wide in DTNs because every single node belongs to a different administrative domain (i.e., individual users). In addition, the mobile phone market is fairly homogeneous in terms of resources consumption, such as battery-life. Therefore, unlike other systems, the *natural* unbalance due to the network structure cannot be compensated by assuming that all the bottleneck nodes will be upgraded.

Under these circumstances failures can occur by a variety of reasons: 1) nodes do not have enough resources to manage the load and fail [28], 2) important nodes are logic targets of attacks [29]; and 3) the mismatch between the incurred cost (total messages to forward) and the utility (own messages to forward) of the bottleneck nodes is a deterrent for their participation in the network, and these nodes are, ironically, the most needed [30]. These problems illustrate the need of a fair distribution of the traffic load in the routing algorithms of DTNs. As an example, fairness and load balance are recurrent topics in other areas such as wireless sensor networks or Internet [31], [32], [33].

The rest of the paper is organized as follows: in Section III we present FairRoute, our DTN routing algorithm. In Section IV we discuss the empirical dataset used in the simulation as well as the experimental design. Sections V – VII summarize and discuss the results of the experiment in terms of efficiency and throughput (Section V), fairness (Section VI) and goodness of the assortative load control (Section VII). Finally Section VIII presents the conclusions and future work.

III. FAIRROUTE ALGORITHM

In DTNs messages are forwarded from node to node upon proximity contacts. Because the trace of contacts is a social network, we turn into social science to design an effective

routing algorithm that overcomes the unfairness problem of existing algorithms. In particular FairRoute relies in *perceived interaction strength* and *assortativity* to guide the forward decision upon contact.

A. Interaction Strength at Different Time Scales

The perceived interaction strength, a concept developed from social influence [34] [35], represents the subjective assessment of the strength of a social tie between two individuals. The interaction strength can be used as an indicator of the likelihood of a contact to be sustained over time. FairRoute uses two different estimators of interaction strength that operate at different time scales: σ_{ij} that indicates the interaction strength between i and j in the short term; and λ_{ij} that indicates the interaction strength in a longer time scale. The strength of the ij interaction increases upon contact but decreases over time at an exponential rate r_σ and r_λ for the short term and long term interaction strength respectively. For such reason it is required that $r_\lambda \ll r_\sigma$.

When a contact between i and j takes place, the nodes update their perceived interaction strengths as follows:

$$\sigma_{ik} = \sigma_{ik} e^{-r_\sigma(t-t_i)} \quad \forall k \in N_i, \quad (1)$$

$$\lambda_{ik} = \lambda_{ik} e^{-r_\lambda(t-t_i)} \quad \forall k \in N_i, \quad (2)$$

$$(\sigma_{ij}, \lambda_{ij}) = (\sigma_{ij}, \lambda_{ij}) + (1, 1), \quad (3)$$

where N_i is the list of contacts of node i , t_i is the time of node i 's last contact (with any other node), and t is the current time. Upon contact, node i updates the exponential decrease of the perceived strength with all the nodes that have encountered in the past (N_i); increases the interactions strength with node j by 1 (both the long term and the short term), and finally, the time of last contact is updated ($t_i = t$).

We then define the *aggregated interaction strength* s_{ij} between nodes i and j as $s_{ij} = \lambda_{ij}(\lambda_{ij} - \sigma_{ij})$. Intuitively, the aggregated interaction strength is an indication of the frequency of long term interactions (proportionality to λ_{ij}), while penalizing spurious bursts of activity (proportionality to the difference between long and short time scales ($\lambda_{ij} - \sigma_{ij}$)).

Let us define u_{ijk} as node i 's perceived utility of node j to deliver a message to node k as

$$u_{ijk} = \frac{\lambda_{jk}(\lambda_{jk} - \sigma_{jk})}{\lambda_{jk}(\lambda_{jk} - \sigma_{jk}) + \lambda_{ik}(\lambda_{ik} - \sigma_{ik})}, \quad (4)$$

which represents *utility* of node j to deliver a message to k as seen by node i , normalized by the total utility. For values of $u_{ijk} > 0.5$, we expect the node j to do a better job than i delivering the message to k . The utility u_{ijk} is defined only when $\lambda_{ik} + \lambda_{jk} > 0$, otherwise we set it as zero.

Analogously, let u_{ij} be i 's perceived utility of node j to deliver a message to *any node*, defined as

$$u_{ij} = \frac{\sum_{k \in N_j} \lambda_{jk}(\lambda_{jk} - \sigma_{jk})}{\sum_{k \in N_j} \lambda_{jk}(\lambda_{jk} - \sigma_{jk}) + \sum_{k \in N_i} \lambda_{ik}(\lambda_{ik} - \sigma_{ik})} \quad (5)$$

Finally, node i will forward to j a message whose destination is k iff

$$\begin{cases} u_{ijk} > \frac{1}{2} & \wedge & (\lambda_{ik} + \lambda_{jk}) > 0 \\ u_{ij} > \frac{1}{2} & \wedge & (\lambda_{ik} + \lambda_{jk}) = 0 \end{cases} \quad (6)$$

Notice that in order to calculate the utilities u_{ijk} , users only exchange their perceived interaction strength on node k , but never exchange the full contact list N_j . In order to obtain N_j , a node i should probe j for every possible value of k in a short period of time (as the values u_{ijk} decay with time). Then it is very easy for j to identify such an attack and deny further communication with i .

B. Assortative-Based Queue Control

The heuristics in (6) do not completely achieve a balanced traffic distribution. The reason is that the routing decision is still a greedy maximization of the utility, with the forwarding biased towards the high connectivity nodes.

In order to counter this effect, we can again turn into sociology and observe the mechanisms by which people decide with whom they interact with. At the risk of being stereotypical, is it an empirically observed fact that the social status of someone's neighbours is a good indicator of his status. The reason is that since social interactions require resources that are limited, humans carefully choose with whom they spend their resources with, and tend to allocate them so that the individual's utility is maximized. In other words, people of the same kind tend to interact together, and tend to disregard interactions from individuals from a lower social status.

For example, a big shot professor would allocate her time to review preliminary work from an equal peer, but she is unlikely to do the same for a graduate student. This behavior, known as *assortativeness* – or homophily – is one of the driving factors on the way individuals interact with each other [36], [37], [38], [39]. Assortativity is in fact what makes social networks different from other complex networks [40].

To capture the assortativeness in our algorithm, we define *social status* of a node i in the DTN to be functionally equivalent to the size of the node's queue length Q_i . The queue length can be interpreted an indication that the node is often chosen to forward packets, and hence, is a measure of its popularity¹.

Since accepting to forward a message has a cost, nodes will only accept forwarding request only from those nodes of equal or higher status. Thus, the status of a node limits the number of nodes to whom it can delegate the cost of carrying the message. With the assortative-based queue control, a node i would forward a message directed k through j if any of the following conditions is met

$$\begin{cases} (u_{ijk} > \frac{1}{2}) & \wedge & (\lambda_{ik} + \lambda_{jk}) > 0 & \wedge & (Q_j \leq Q_i) \\ (u_{ijk} = 1) & \wedge & (\lambda_{ik} + \lambda_{jk}) > 0 & & \\ (u_{ij} > \frac{1}{2}) & \wedge & (\lambda_{ik} + \lambda_{jk}) = 0 & \wedge & (Q_j \leq Q_i) \end{cases} \quad (7)$$

¹Note that popularity does not necessarily means that the node is a high performer, but instead that the node is *perceived* to be very useful.

High status nodes will be able to forward messages faster due to their privileged position, whereas low status nodes will have to find alternative paths. Since contact and social networks have a diversity of paths between two nodes [41], [42], constrains in the forwarding introduced by the assortative-based control queue does not necessarily implies a reduction of throughput (Section V). On the other hand, it does have a positive impact on the fairness of the routing algorithm (Section VI). This mechanism of the queue size control to achieve load balance is analogous to the *back pressure* congestion control [31], which is applied to fields such as the Internet [33], [43], ATM and ethernet networks [44], and wireless sensor networks [32], among others.

C. Privacy Considerations

As social network-based routing algorithms become more sophisticated they require access to more information about users, and therefore, they raise rightful privacy concerns. For instance, Bubble and SimBet require access to the full list of contacts of a contact. While this is common practice in social networking sites, users of these systems explicitly aggregate their friends. This is not the case of DTNs, where contacts are aggregated upon proximity encounters. In addition, Bubble also requires access to the social group label the target user belongs to. Having access to whom and how often a user has been interacting with is a security issue, and can be misused, for example, for profiling. While the authors' argument is to trade the privacy off in return for performance, we believe that it is preferable to obtain performance without giving sensitive information away.

IV. CONTACT TRACE AND EXPERIMENTAL DESIGN

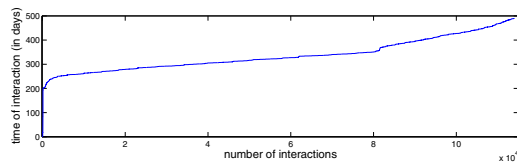
A. Contact Dataset

Contact trace datasets [12], [13] are crucial to evaluate the performance of routing algorithms in DTNs. While large scale dataset (e.g. city-wide) are not available [45], the existing datasets allow the benchmarking of social network based routing algorithms. In this paper we use the MIT reality mining dataset [13], which is widely used in the literature, thus allowing a fair comparison of FairRoute with existing DTN routing algorithm. The MIT dataset was collected in the MIT campus, that represents a general purpose scenario better than other existing datasets such as Infocom 2006 [10].

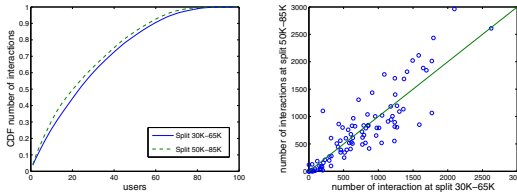
In the MIT dataset users carried a cell phone with a software that logged contacts by proximity using the bluetooth discovery function. From the dataset we extracted the list of observed contacts: $\langle i, j, t \rangle$ where i and j are users, and t is the time at which the interaction takes place. Some contacts might not last enough to allow data transfer between cell-phones. However, since we are only interested in patterns of human-to-human contact, we do not take this factor into consideration.

The MIT dataset provides contact trace information for 100 users over a period of over a year². Figure 1(a) shows the time at which each contact takes place; there are three different

²The first interaction is logged on 01/01/2004 and the last on 05/05/2005.



(a) Time (in days) at which the n-th contact takes place



(b) Cumulative Distribution of contacts by user in two overlapping subsets: 30K to 65K and 50K to 85K contacts. (c) Scatter plot of the number of interactions at split 30K-65K subsets. Each circle represents one of the 96 users that experience at least one contact

Fig. 1. Partition of the MIT reality mining project dataset in overlapping subsets

stages, the first 652 interactions took place over the first 7 months, which might correspond to the development of the system. In the following 5 months the system showed 79840 interactions. In the last 4 months there was a decrease of the activity and only 33577 contacts were collected.

We partition the MIT dataset into 16 overlapping subsets each consisting in 35K contacts and a relative offset of 5K contacts. The utility of this partition is twofold. First, it allows us to test the sensitivity of the algorithms against the existing variance of the contacts. Second, it minimizes the bias of the results toward the MIT dataset.

Figure 1(b) shows the comparison between the user’s cumulative distribution in two partially overlapping subsets (from 30 to 65 thousand-th contact and from 50 to 85 thousand-th contact). As expected, the aggregate contact distribution does not experience a lot of variance – as contacts take place between the same people in the same community. However, a look at the individual level (Fig. 1(c)) reveals that the number of contacts per user experiences a fair amount of variation. Note that both subsets have a 42% overlap.

B. Reference Algorithms

We evaluate the performance of FairRoute against well-known algorithms in the literature.

- Epidemic routing [4] is a flooding multi-copy algorithm often used as a baseline for comparison because, while extremely inefficient in the number of forwards, it is optimal in terms of throughput. Furthermore, Epidemic routing is very adequate when exploring fairness, because its load distribution is caused solely by the network topology (i.e., it does not apply any heuristic to direct the forwarding).
- PROPHET [46] is widely used as the reference algorithm in terms of efficiency (e.g. in [16], [19]). PROPHET uses the history of contacts to calculate the probability of

a node to deliver the message to the destination. Like FairRoute, it also takes the time between contacts into account to update its utility function. In our simulations, we use the parameters suggested by the author’s of $\{P_{init}, \beta, \sigma\} = \{0.75, 0.25, 0.99\}$.

- SimBet [16] combines a decentralized version of betweenness centrality [47] and the probability of future collaboration [17]. For the simulations, and following the authors, we set the parameter $\alpha = 0.5$.

The algorithm Bubble [19], which exploits the community structure found in social networks using a distributed clustering algorithm [48], was not included in our results because it requires the sender to know the address and the social group of the destination node, which is unfair to the other algorithms that only require the address. In addition, in the original paper, Bubble is not compared against the existing SimBet, but their comparison with PROPHET indicates that its performance is better than PROPHET but worse than SimBet.

C. Experimental Design

We run the four algorithms for each one of the 16 subsets of the MIT contact dataset. The first 5K contacts of the subset are used to bootstrap the different algorithms. This warm up period is used to avoid introducing artifacts in the results due to the heuristics’ different convergence time. After warm up phase, all the nodes send a message to every other node in the system, resulting in 9120 messages.

FairRoute’s algorithm parameters are set to $r_\lambda = 5 \times 10^{-4} h^{-1}$ and $r_\sigma = 5 \times 10^{-3} h^{-1}$. Which correspond to the an average inter-contact time of 2000 and 200 hours respectively. These values will depend on the characteristic inter-contact time of the contact network under scrutiny, for the MIT dataset they are extremely large because the contacts are very sparse (Fig. 1(a)). We also tested the parameter’s sensitivity. Perturbations up to one order of magnitude did not show qualitative changes on the results of the algorithm.

V. THROUGHPUT AND EFFICIENCY EVALUATION

Maximizing the throughput, i.e., increasing the number of messages that reach their destination, is the goal of any routing algorithm. However, minimizing the number of forwards is also crucial to limit the resource consumption due to traffic: battery-life, storage, bandwidth. For this reason the *efficiency*, or the ratio between throughput and number of forwards, in combination with the throughput, are generally accepted as the metrics of choice for the routing algorithms³.

Figure 2(a) shows the average throughput across the 16 subsets of the MIT dataset for all the algorithms. Table I shows the variance and numerical results summarized. The Epidemic routing yields the optimal throughput of 90% on

³We do not consider forwarding delay as a metric because of the sparsity of contacts in the MIT dataset. Inter-contact times can be up to days. Furthermore, the inter-contact time of different subsets do not have the same length and/or distribution since subsets are split by number of interactions. Because of the above analysis based on delay would have rendered inconclusive results.

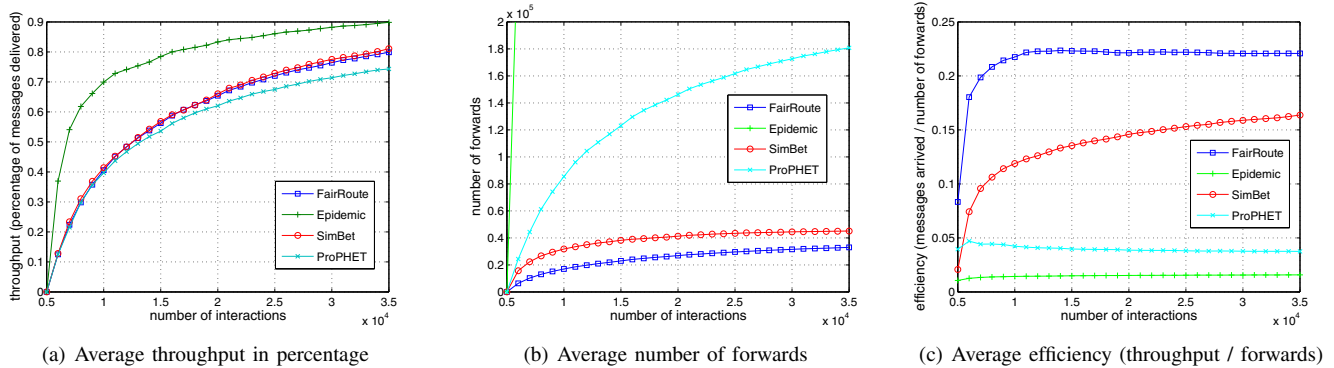


Fig. 2. Throughput and Efficiency Evaluation

average. SimBet and FairRoute deliver 81.1% and 79.9% of the messages respectively. PROPHET get a slightly worse performance with an average throughput of 74.5%.

While all the algorithms achieve asymptotically good throughput, it is important to note that epidemic is not only optimal in terms of throughput but also in delivery speed. As an example, if we consider the throughput after 10K contacts, the throughput is 36.4%, 12.7%, 12.5% and 12.4% for Epidemic, SimBet, PROPHET and FairRoute respectively. As we can see, all algorithms have a very similar performance but the epidemic that is clearly ahead of the rest.

However, as Figure 2(b) shows, FairRoute requires considerably less forwards. SimBet, PROPHET and epidemic require 1.37, 5.5 and 157 times more forwards than FairRoute.

We would like to mention that SimBet [16] uses a different definition for forwards. They do not count as a forward the final hop by which the message arrives to destination (the handover). Also, they count as one forward when different messages can be bundled in the same data transfer session. This metric is somewhat misleading in terms of the cost associated to a message. Handovers messages still have a cost that needs to be accounted for when it send via wireless. The same happens to messages that are delivered in the same data transfer; the time of the transfer and its toll on the resources will be multiplied by the number of messages contained in the bundle. With SimBet’s metric, FairRoute (with 11849 operations) still outperforms SimBet (with 13908 operations). Despite that, and for the sake of clarity, we take into account each individual message transfer as a forward.

Finally, Figure 2(c) displays the average efficiency. The most efficient algorithm is FairRoute followed by SimBet. PROPHET and epidemic routing fall behind due to their high costs in terms of number of forwards. FairRoute’s efficiency of 0.22 means that each message delivered to the destination requires, on average, 4.52 forward operations. Thus, assuming that the traffic generated by the routing algorithms is negligible compared to the traffic generated by the forwards, one 100K message results in a 452Kb of traffic to be handled by the participants of the DTN. In comparison, SimBet, PROPHET and Epidemic routing have a cost of 6.02, 26.3 and 62.5 times the cost of one forward operation respectively.

Algorithm	Throughput		Forwards		Efficiency	
	Avg (%)	StdDev	Avg	StdDev	Avg	StdDev
Epidemic	89.85	3.03	518790	18947	0.016	0.001
PROPHET	74.53	4.21	180790	26187	0.038	0.006
SimBet	81.08	4.74	45141	5389	0.166	0.022
FairRoute	79.90	3.94	32998	1492	0.221	0.007

TABLE I
AVERAGE AND STANDARD DEVIATION FOR THROUGHPUT, FORWARDS AND EFFICIENCY AT THE END OF THE SIMULATION.

Algorithm	top-1	top-10	top-20	Threshold
Epidemic	0.081	0.373	0.571	17
PROPHET	0.075	0.421	0.616	14
SimBet	0.134	0.539	0.706	9
FairRoute	0.046	0.275	0.452	24

TABLE II
DISTRIBUTION OF FORWARDS BY USERS: TOP- N CORRESPONDS TO THE FRACTION OF THE TOTAL NUMBER OF FORWARDS CARRIED OUT BY THE TOP N USERS. THRESHOLD IS THE NUMBER OF USERS THAT CARRY OUT 50% OF THE ALL FORWARDS. THE TOTAL NUMBER OF USERS IS 96.

A. Variance Across Subsets

As expected, the different subsets of the contact trace have a strong variance in throughput, number of forwards and efficiency (Table I). For instance, the throughput has a standard deviation of 3.03% for epidemic routing, 3.94% for FairRoute and 4.74% for SimBet. In the case of the number of forwards, the standard deviation accounts for differences of 11.9%, 4.5% and 3.7% for SimBet, FairRoute and epidemic routing respectively.

Such differences on the throughput and in the number of forwards clearly show the sensitivity of routing algorithms to the topology of the network of contacts. Despite the subsets belonging to the same social/contact network, and being clearly overlapped, the algorithms are sensitive to fluctuations in the contact patterns. We anticipate the need for large scale contact/social networks like [45] in order to explore the robustness of the routing algorithms for large-scale deployment.

VI. FAIRNESS EVALUATION

In the previous section we showed that FairRoute compares favorably to the state-of-the art DTN routing algorithms.

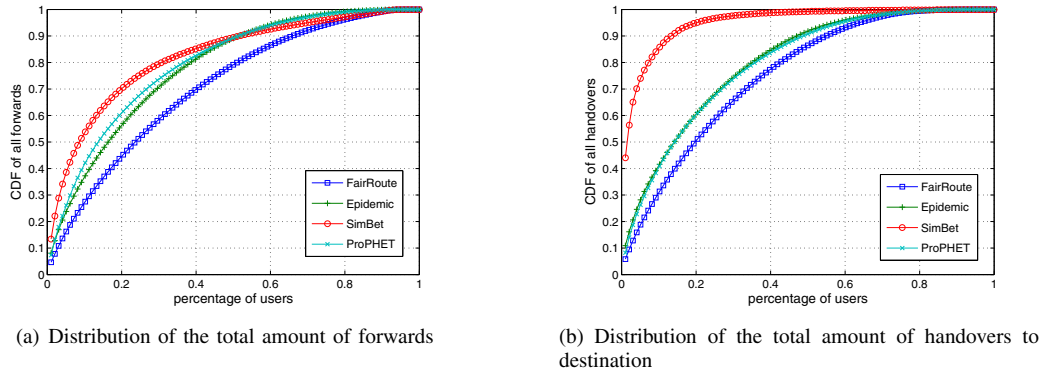


Fig. 3. Fairness evaluation

Compared to SimBet, FairRoute obtains a 33% increase in efficiency at a cost of just 1.2% loss in throughput. However, despite the good performance on the standard metric, the aim of FairRoute is neither throughput nor efficiency, but fairness.

Figure 3(a) shows the cumulative distribution of forwards for the four different algorithms. We can see that the *natural* load distribution due to the network topology – shown by the Epidemic routing – is already unbalanced: 50% of the traffic is handled by only 17 users, the top-1 user deal single handedly with 8.1% of the traffic in the system and the top-10 users deals with 37%.

While the perfectly fair load balance would be that 50% of traffic to be managed by 50% of users, it can only be achieved in a social network if there are no forwards but only handovers (i.e., each individual simply waits until it comes across the destination of the message – if ever). Since the perfect fairness is unrealistic, we set the baseline reference as the load distribution yield by the Epidemic routing, where the network topology is the only factor driving the load distribution.

As we discussed in Section II, the traffic load can only get worse for PROPHET and SimBET algorithms. In SimBet 50% of traffic goes through only 9 nodes, from which the top one handles 13.4%. PROPHET does better with 50% of traffic being managed by 14 nodes and the top one node 7.5% of the traffic load. This behavior can be extrapolated to all routing algorithms based solely on *best next hop* heuristics as these heuristics contribute to the already existing process of preferential attachment [26] of traffic due to the network topology. The fate of routing algorithms focused solely on best next hope heuristics is to increase performance (throughput and efficiency) at the expenses of an unfair traffic distribution.

FairRoute, on the other hand, exhibits a load distribution that is more fair than the *natural* distribution of the epidemic routing. The assortativeness-based control queue is able to distribute the traffic more evenly among all the users, and therefore, leverage the congestion problems and failures discussed in section III. In FairRoute, 50% of the traffic is handled by 25% of users, significantly better than the 9.3% of users for SimBet, the 14.5% of users for PROPHET and the

17.7% of the users for epidemic routing.

A. Query Size Dynamics and Robustness

Figure 4 displays the evolution of the queue size distribution over time for the SimBet, PROPHET and FairRoute algorithms. The white space between lines in the figure represents the fraction of messages in transit that has in its queue (Q_i) at every interaction. During the initial stage of training (first 5K contacts) there is no forwarding and each user’s queue contains exactly 95 messages.

In the case of SimBet, (Figure 4(a)), it is revealing to see that right after the warmup period, two of the users accumulate most of the messages in transit in their queues. This behavior is inadvisable in terms of robustness, as a departure or a failure of one of those users would result in a loss of an important fraction of the messages waiting to be delivered.

The behavior of the queues of PROPHET (Fig. 4(b)) is more robust than SimBet’s. The queue size distribution is quite homogeneously distributed among all users. The apparent discontinuity of the lines in the PROPHET is due to the high turn over of messages in the queue; PROPHET has one order of magnitude more forwards than SimBET and FairRoute. Consequently, messages exchange hands much more frequently and the queue distribution shows these fluctuations.

The FairRoute’s queue size distribution (Fig. 4(c)) is more balanced than the other algorithms. This is due to its assortative-based queue control mechanism, which, as we saw in Fig. 3(a), also makes the traffic load more balanced.

With the traffic load distribution (Fig. 3(a)) and the queue size distribution (Fig. 4) we can explore the effect of fairness on robustness with a simple analysis. Let us assume that an error happens during the forwarding process, and denote e_i as the probability of node i fails. Let $Q_{i,t}$ be the size of the queue of node i at time t . We define q_i as the average relative size of i ’s queue over time with respect to the total number of messages in the network, i.e., $q_i = \frac{\sum_t Q_{i,t}}{\sum_i \sum_j Q_{j,t}}$. Let us also denote the traffic load of node i as f_i , which is the fraction of forwards carried out by i . Finally, let us define the *damage* D as $D = \sum_i e_i q_i$, i.e., the average cost in the fraction of messages in transit lost due a node’s failure.

We consider three types of failures or errors:

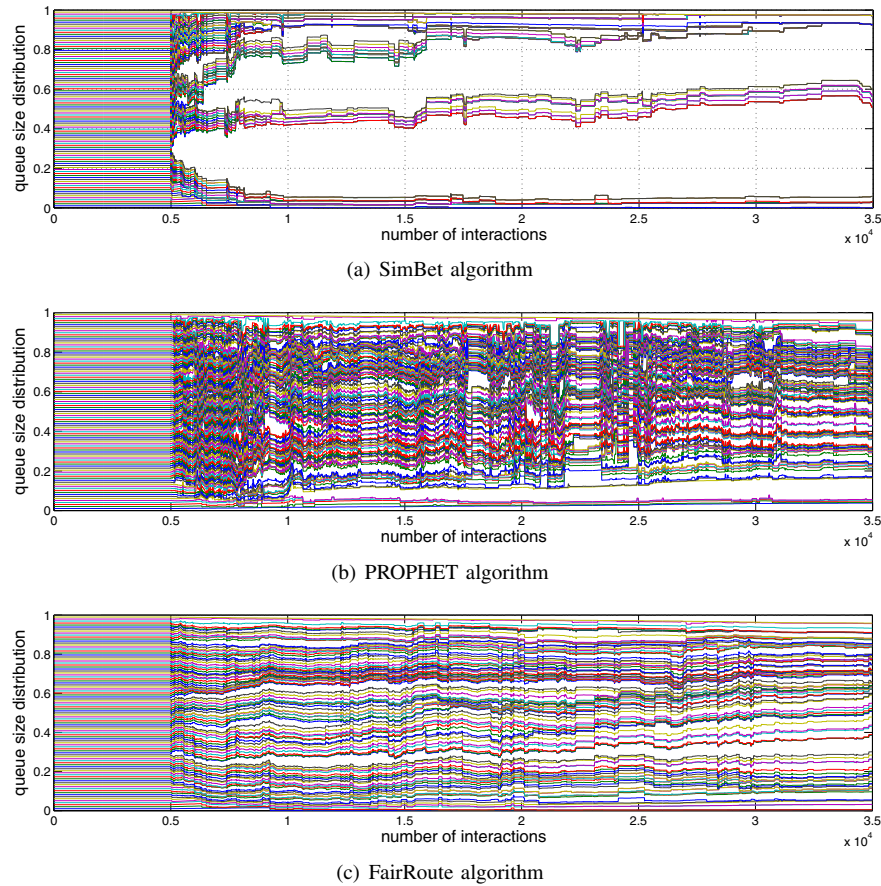


Fig. 4. Evolution of the queue size distribution for all nodes (stacked). The proportion of messages in transit stored in user’s i queue corresponds to the white space between to lines. The proportion are stacked so that it sums one. The first 5K contacts correspond to the warm up. In this period each user has 96^{-1} of the messages stored in the queue. After the 5K contact, the forwarding starts and the queue size distribution evolves over time. This plot correspond to the 10K-45K split of the MIT contact dataset.

- Random Error: the probability of failure is equally distributed between all nodes. Thus, $e_i = \frac{1}{96}$. Then, its trivial to see that the damage is $D = \frac{1}{96}$. A random error produces on average an loss of 1.04% of the messages regardless the routing algorithm.
- Load/cost error⁴: in the case of an error, the probability of failure of a node i is proportional to its traffic load f_i (resources are depleted proportionally to the traffic). Then, the average damage of *one* failure due to load is $D = \sum_i e_i q_i$, resulting in a damage of 5.81 for SimBet, 1.27 for PROPHET, and 2.21 for FairRoute. Also, note that the probability of individual errors increases with the traffic load, and so they will happen more often in algorithms that have more forwards. With PROPHET and SimBet performing 5.6 and 1.4 more forwards than FairRoute, the overall damage of load errors for FairRoute is considerably less than for the other algorithms.
- Attack: select the node that maximizes the damage ($argmax(q_i)$) and it will be 38.45%, 5.46% and 4.97% for SimBet, PROPHET and FairRoute. Unbalanced load

⁴The case of a node defecting because its perceived cost/benefit ratio is equivalent to the load error case.

Algorithm	top-1	top-5	top-10	Threshold
Epidemic	0.109	0.282	0.416	14
PROPHET	0.082	0.268	0.409	14
SimBet	0.441	0.739	0.858	2
FairRoute	0.058	0.188	0.313	20

TABLE III

DISTRIBUTION OF HANDOVERS BY USERS: TOP- N CORRESPONDS TO THE FRACTION OF THE TOTAL NUMBER OF HANDOVERS CARRIED OUT BY THE TOP N USERS. THRESHOLD IS THE NUMBER OF USERS THAT CARRY OUT 50% OF THE ALL HANDOVERS. THE TOTAL NUMBER OF USERS IS 96.

distributions are very fragile against attacks.

A more detailed analysis of robustness is outside the scope of this paper, at it has more complex ramifications in throughput and efficiency. Errors are not independent, and one failure can produce others in a cascade. However, the simple analysis performed above illustrates the fact that *fairness is not optional*, and hence, designers of DTN routing algorithms need to take fairness into consideration.

B. Dynamic and Static Heuristics

The subset of forwards that are also handovers have a special interest because they reveal another effect of the best

next hop heuristics on the routing behavior. Figure 3(b) and Table III show who are the users that perform the last deliver of a message to the final destination. For the epidemic routing, the top-5 users account for 28.8% of handovers. The same 5 users account for 26.80% of all handovers for PROPHET and 73.95% for SimBet.

These results illustrate one shortcoming of SimBet's heuristic: the delivery utility does not age over time and therefore the heuristic does not forget. If a node's utility values are very stable against the changes of user's contact patterns a hill climbing strategy might be caught in a local minima. Once there, it will not forward a message unless the actual destination is encountered. Thus, the most central nodes effectively become *mailmen*: the messages get to them very efficiently due to the high performance of SimBet's heuristics but once they reach the center of the network, the node will not deliver them to anyone but to the destination. Conversely, PROPHET and FairRoute's heuristics are dynamic and change according to the user's contacts fluctuation. The message will not be trapped in a local minima and therefore the handover distribution is similar to the handover distribution of the Epidemic Routing.

VII. EXPLORING FAIRROUTE'S ASSORTATIVENESS

In the last section, we saw that the assortative-based queue control mechanism used by FairRoute yields a load distribution that is more balanced than the *natural* load distribution that arises from the structure of a complex network. It is reasonable to question whether this control mechanism can be seamlessly implemented in other algorithms to increase their fairness.

Figure 5(a) depicts the cumulative distribution of the forwards when all the algorithms use the same assortativeness principle as FairRoute, as described in Section III-B. We can see that assortativeness has no effect in the epidemic, as it follows a flooding policy. However, SimBet and PROPHET do show an increase on their fairness, which is now better than the *natural* load distribution. The efficiency of the routing algorithms is also increased: from 0.038 to 0.077 for PROPHET and from 0.166 to 0.219 for SimBet as shown in Table IV.

This increase in fairness and efficiency comes with a penalty in throughput, reduced from 81.08% to 67.35% for SimBet and from 74.33% to 67.47% for PROPHET. The loss of throughput is due to the inability of the SimBet's and PROPHET's heuristics to find alternative paths once the best next hop node stops accepting messages. FairRoute, on the other hand, is able to explore multiple paths between nodes [41], [42] more effectively and to keep its throughput of 73.95%.

VIII. CONCLUSIONS AND FUTURE WORK

In this paper we presented FairRoute, an algorithm for routing in delay tolerant networks that achieves over 33% more efficiency than state-of-the art routing algorithms, without any loss in throughput. FairRoute achieves this performance by favoring peers that *appear* to be good candidates to deliver the message successfully to its target (*perceived interaction strength*), and by limiting communication with those nodes that have less importance in the network (*assortativity*).

Algorithm	Throughput		Forwards		Efficiency	
	Avg(%)	Std	Avg	Std	Avg	Std
Epidemic (A)	89.75	3.07	499509	18250	0.016	0.001
PROPHET (A)	67.47	4.62	78221	13575	0.077	0.014
SimBet (A)	67.35	4.62	28072	2180	0.219	0.018
FairRoute	79.90	3.94	32998	1492	0.221	0.007

TABLE IV
AVERAGE AND STANDARD DEVIATION FOR THROUGHPUT, FORWARDS AND EFFICIENCY AT THE END OF THE SIMULATION. PROPHET, SIMBET AND EPIDEMIC INCORPORATE THE ASSORTATIVITY-BASED QUEUE PROPOSED BY FAIRROUTE

The most important contribution of FairRoute, however, is that unlike other routing algorithms inspired in social mechanisms, our algorithm distributes the traffic load more evenly. While in SimBet the top 10 users in the MIT reality mining project dataset manage 54% of the traffic, they account for only 27.5% of the traffic when FairRoute is used. Our algorithm achieves a load distribution that is even *more balanced* than the *natural* load distribution that arises from the social network-like structure of the MIT dataset, in which the top 10 users manage the 37% of the traffic.

At this point we would like to propose a last thought experiment. Let us assume that the reader is one of the top users in the MIT dataset, i.e., the reader plays a central role in the community and therefore a lot of traffic goes through his/her cell-phone. Let us also assume that the messages size is 100Kb (e.g. an MMS, or an email with an attachment). Then, on average, every single message sent between any two individuals in the network would generate a total traffic of 452Kb, 602Kb and 6230Kb for FairRoute, SimBet and Epidemic routing respectively. The cost for the reader, a central node in the social network, is of 20Kb with FairRoute, 80Kb with SimBet, and as much as 500Kb with Epidemic routing **for every single message sent in the network**.

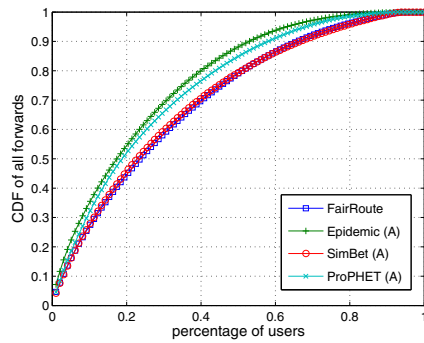
Needless to say that Epidemic routing is prohibitively expensive. The question remaining is whether SimBet, or even FairRoute, are prohibitive as well. Although FairRoute reduces the cost of SimBet by a factor of four, it still carries a significant cost for the central users of the network that they might not be willing to assume. This reflection does not intend to downplay the Delay Tolerant Networks, an area of immense potential, but to highlight the fact that there is a need to switch the metrics of choice from raw efficiency to fairness and load balancing in order to obtain more realistic routing algorithms.

ACKNOWLEDGEMENTS

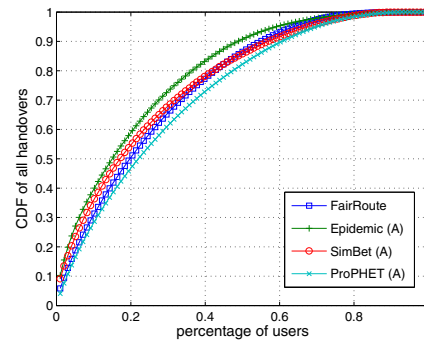
We thank Elizabeth Daly and Heiko Yoneki for kindly answering our questions regarding SimBet and BubbleRAP. Alberto Lopez Toledo is sponsored by the Junior Empresa program of the Institució Catalana de Recerca i Estudis Avançats (ICREA), Barcelona Spain.

REFERENCES

- [1] S. Keshav. Why cell phones will dominate the future internet. *SIGCOMM Comput. Commun. Rev.*, 35(2):83–86, 2005.
- [2] K. Fall. A delay-tolerant network architecture for challenged internets. *In Proc. SIGCOMM*, 2003.



(a) Distribution of the total amount of forwards



(b) Distribution of the total amount of handovers

Fig. 5. Fairness evaluation. The routing algorithms are modified version of SimBet, PROPHET and Epidemic routing with the FairRoute’s assortativity-based queue control.

[3] M. Grossglauser and D. Tse. Mobility increases the capacity of ad hoc wireless networks. *IEEE/ACM Transactions on Networks*, 10(4):477–486, 2002.

[4] A. Vahdat and D. Becker. Epidemic routing for partially connected ad hoc networks. *Technical Report CS-200006, Duke University*, 2000.

[5] Q. Li and D. Rus. Sending messages to mobile users in disconnected ad hoc networks. In *Proc. of MobiCom*, pages 44–55, 2000.

[6] S. Jain, K. Fall, and R. Patra. Routing in a delay tolerant network. *SIGCOMM Comput. Commun. Rev.*, 34(4):145–158, 2004.

[7] J. Burgess, B. Gallagher, B. Jensen, and B.N. Levine. Routing for vehicle-based disruption-tolerant networking. In *Proc. of Infocom*, 4:1688–1698, 2006.

[8] M. Grossglauser and M. Vetterly. Locating nodes with ease: last encounter routing in ad hoc networks through mobility diffusion. In *Proc. of Infocom*, 3(4):1954–1964, 2003.

[9] T. Karagiannis, J.-Y.L. Boudec, and M. Vojnovic. Power-law and exponential decay of intercontact time between mobile devices. *Proc. of ACM MobiCom*, 2007.

[10] A. Chaintreau, P. Hui, J. Crowford, C. Diot, J. Scott, and R. Gass. Impact of human mobility on opportunistic forwarding algorithms. *IEEE Trans. Mob. Comp.*, 6(6):606–620, 2007.

[11] P. Costa, C. Mascolo, M. Musolesi, and G.-P. Picco. Socially-aware routing for publish-subscribe in delay-tolerant mobile ad hoc networks. *IEEE Journal on Selected Areas in Communications*, 26(5), 2008.

[12] Huggle Project. <http://www.huggleproject.org/>.

[13] N. Eagle and A. Pentland. Reality mining: sensing complex social systems. *Personal and Ubiquitous Computing*, 10(4):255–268, 2006.

[14] E. Yoneki, P. Hui, and J. Crowcroft. Distinct types of hubs in human dynamic networks. *EuroSys Workshop on Social Network Systems*, 2008.

[15] A. Chaintreau, P. Fraigniaud, and E. Lebar. Opportunistic spatial gossip over mobile social networks. 6(6):606–620, 2007.

[16] E. Daly and M. Haahr. Social network analysis for routing in disconnected delay-tolerant manets. In *Proc. of ACM MobiHoc*, 2007.

[17] D. Liben-Nowell and J. Kleinberg. The link prediction problem for social networks. In *Proc. of CIKM*, 2003.

[18] L.C. Freeman. A set of measuring centrality based on betweenness. *Sociometry*, 40:35–41, 1977.

[19] P. Hui, J. Crowcroft, and E. Yoneki. Bubble rap: Social based forwarding in delay tolerant networks. In *Proc. of MobiHoc*, 2008.

[20] M.E.J. Newman and M. Girvan. Finding and evaluating community structure. *Phys. Rev. E*, 69:026113, 2004.

[21] N. Sastry. Folksonomy-based reasoning in opportunistic networks. In *Proc. of CoNEXT*, page 49, 2007.

[22] S. Guo and S. Keshav. Fair and efficient scheduling in data ferrying networks. In *Proc. of ACM/CoNEXT*, 2007.

[23] A. Balasubramanian, B.N. Levine, and A. Venkataramani. Dtn routing as a resource allocation problem. In *Proc. of SIGCOMM*, 2007.

[24] Vijay Erramilli, Augustin Chaintreau, Mark Crovella, and Christophe Diot. Delegation forwarding. In *Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing (ACM MobiHoc)*, pages 251–259, May 2008.

[25] A. Mtibaa, A. Chaintreau, J. LeBrun, E. Oliver, A.-K. Pietilainen, and C. Dio. Are you moved by your social network application? In *Proc. of SIGCOMM Workshop on Social Networks*, 2008.

[26] R. Albert and A.-L. Barabási. Statistical mechanics of complex networks. *Review of Modern Physics*, 74:47–97, 2002.

[27] M.E.J. Newman. The structure and function of complex networks. *SIAM Review*, 45(2):167–256, 2003.

[28] P. Hølem and B.J. Kim. Vertex overload breakdown in evolving networks. *Phys. Rev. E*, 65:066109, 2002.

[29] R. Albert, H. Jeong, and A.-L. Barabási. The internet’s achilles’ heel: Error and attack tolerance in complex networks. *Nature*, 406:378–381, 2000.

[30] E. Fehr and S. Gächter. Fairness and retaliation: The economics of reciprocity. *Journal of Economic Perspectives*, 14(3):159–182, 2000.

[31] L. Tassiulas. Adaptive back-pressure congestion control based on local information. *IEEE Transaction on Automatic Control*, 40(2):236, 1995.

[32] B. Hull, K. Jamieson, and H. Balakrishnan. Mitigating congestion in wireless sensor networks. In *Proc. of 2nd International Conference on Embedded Networks Sensor Systems*, pages 134–147, 1994.

[33] S. Floyd. Tcp and explicit congestion notification. *ACM Computer Communication Review*, 24(5):10–23, 1994.

[34] D.J. Watts and P. Sheridan-Dodds. Influentials, networks, and public opinion formation. *Journal of Consumer Research*, 34(4):441–458, 2007.

[35] M. Granovetter. The strength of weak ties. *American Journal of Sociology*, 78, 1973.

[36] P. Blau. *Exchange and Power in Social Life*. Wiley, New York, 1964.

[37] R.V. Gould. Collective action and network structure. *American Sociological Review*, 58:182–196, 1993.

[38] N. Mark. Beyond individual differences: Social differentiation from first principles. *American Sociological Review*, 63:309–330, 1998.

[39] N. Lin. *Social Capital: A theory of social structure and action*. Cambridge University Press, 2001.

[40] M.E.J. Newman and J. Park. Why social networks are different from other types of networks. *Phys. Rev. E*, 68:036122, 2003.

[41] V. Erramilli, A. Chaintreau, M. Crovella, and C. Diot. Diversity of forwarding paths in pocket switched networks. In *Proc. of IMC*, 2007.

[42] D.J. Watts. Network dynamics and the small world phenomenon. *American Journal of Sociology*, 105(2):493–527, 1999.

[43] Sally Floyd and Van Jacobson. Random early detection gateways for congestion avoidance. *IEEE/ACM Transactions on Networking*, 1:397–413, 1993.

[44] IEEE. IEEE, ed., 802.3 standard. IEEE, 2002, ch. 2, annex 31b: MAC control PAUSE operation.

[45] M.C. González, C.A. Hidalgo, and A.-L. Barabási. Understanding individual human mobility patterns. *Nature*, 453:779–782, 2008.

[46] A. Lindgren, A. Doria, and O. Schelen. *Probabilistic Routing in intermittently connected networks*, volume 3126, pages 239–254. 2004.

[47] P.V. Marsden. Egocentric networks and sociometric measures of network centrality. *Social Networks*, 24:407–422, 2002.

[48] P. Hui and E. Yoneki. Distributed community detection in delay tolerant networks. In *Proc. of MobiHoc*, 2007.