Fairness-related challenges in mobile opportunistic networking

Abderrahmen Mtibaa *, Khaled A. Harras

School of Computer Science, Carnegie Mellon University, Doha, Qatar

A R T I C L E   I N F O

Article history:
Received 28 November 2011
Received in revised form 24 May 2012
Accepted 29 August 2012
Available online 21 September 2012

Keywords:
Fairness
Mobile opportunistic networks
Social networks
Rank-based opportunistic forwarding
Fairness efficiency trade-off

A B S T R A C T

The fundamental challenge in opportunistic networking, regardless of the application, is when and how to forward a message. Rank-based forwarding techniques currently represent one of the most promising methods for addressing this message forwarding challenge. While these techniques have demonstrated great efficiency in performance, they do not address the rising concern of fairness amongst various nodes in the network. Higher ranked nodes typically carry the largest burden in delivering messages, which creates a high potential of dissatisfaction amongst them. In this paper, we adopt a real-trace driven approach to study and analyze the trade-offs between efficiency, cost, and fairness of rank-based forwarding techniques in mobile opportunistic networks.

Our work comprises three major contributions. First, we quantitatively analyze the trade-off between fair and efficient environments. Second, we demonstrate how fairness coupled with efficiency can be achieved based on real mobility traces. Third, we propose FOG, a real-time distributed framework to ensure efficiency–fairness trade-off using local information. Our framework, FOG, enables state-of-the-art rank-based opportunistic forwarding algorithms to ensure a better fairness–efficiency trade-off while maintaining a low overhead. Within FOG, we implement two real-time distributed fairness algorithms; Proximity Fairness Algorithm (PFA), and Message Context Fairness Algorithm (MCFA). Our data-driven experiments and analysis show that mobile opportunistic communication between users may fail with the absence of fairness in participating high-ranked nodes, and an absolute fair treatment of all users yields inefficient communication performance. Finally our analysis shows that FOG-based algorithms ensure relative equality in the distribution of resource usage among neighbor nodes while keeping the success rate and cost performance near optimal.

© 2012 Elsevier B.V. All rights reserved.

1. Introduction

Mobile opportunistic networks interconnect nodes with heterogeneous contact rates, unpredictable mobility, and limited resources. These mobile nodes communicate relying on both the transport of messages as well as multi-hop forwarding. Current forwarding techniques [3,28,19,21,9] are generally designed to efficiently select the most likely relay nodes to deliver a message to its destination. Within those techniques, rank-based forwarding [5,7,8] represents one of the most promising methods for addressing this message forwarding challenge. This method differs in the type of information used (e.g., information acquired during contacts [7,8], or social interaction between users [5,26,25]), as well as how it is used to rank nodes in the network. A node with a lower rank will forward messages to nodes with higher ranks. This solution, however, creates a high potential of dissatisfaction among high ranked nodes that carry a heavier burden compared to others.

Ensuring fairness in mobile opportunistic networks is a crucial goal if rank-based forwarding algorithms are to be adopted in the future. In such networks, nodes can behave selfishly and try to only maximize their own utility...
without considering the global system-wide performance. Previous studies have considered absolute fair allocation of user resources [1,18,20]. They have shown that absolute fairness results in higher end-to-end delivery delays. They did not, however, investigate trade-offs between the three metrics in mobile opportunistic networks: fairness, end-to-end delay, and cost.

Throughout this paper, we adopt a real-trace driven approach to quantitatively analyze those trade-offs, and propose a real-time distributed framework for fairness in opportunistic networking. Our initial conference version compromises three major contributions [23]. First, we quantitatively analyze the impact of an absolute fair and efficient environment on the overall network performance. We show that in an unfair environment, selecting preferential or popular relay nodes in forwarding decisions is efficient and yields enhanced forwarding performance. We also show that absolute balancing across nodes causes significant end-to-end delay and severe performance degradation. Second, we study the trade-off between fairness, cost, and efficiency in opportunistic forwarding. Relying on our experimental datasets, we consider an offline approach to construct forwarding paths while ensuring both fairness and efficiency, and keeping the cost near optimal. Finally, we adopt a distributed real-time approach in order to efficiently discover these paths while utilizing only local information. We call our distributed real-time framework FOG (Fairness-based Opportunistic networking framework).

This journal paper adds the following contributions beyond the conference version [23]:

- We address fairness in opportunistic forwarding more generally. While [23] focuses mainly on a trade-off between fairness and success delivery ratio, this paper extends that work to quantitatively study different trade-offs between fairness, cost and success rate. Cost metric is addressed after identifying the best performing algorithms with respect to the other two metrics. We look then at the cost within those nominal values. We investigate fairness issues of social-based forwarding algorithms such as PeopleRank [25] and Simbet [5] and contact-based forwarding such as Frequency (FR) [8] and Last Contact (LC) [7].
- We propose two real-time distributed algorithms to maximize the overall user satisfaction: Proximity Fairness Algorithm (PFA) and Message Context Fairness Algorithm (MCFA). While the main advantage of PFA is its simplicity which allows any rank-based forwarding algorithm to ensure minimum fairness without using additional network information, MCFA uses local information to make better forwarding decisions. MCFA enables cost performance aware message delivery in mobile opportunistic networks taking into account the number of message replicas in the network and user satisfaction of each message. We model the satisfaction a node derives from sending and receiving messages as a utility function. MCFA uses such utility in order to avoid overwhelming the popular nodes with many messages when other nodes can deliver them to theirs destination in a bounded delay.

- Our real-trace analysis includes additional results that rely on a modified San Francisco [27] dataset that we synthesize. We show that FOG uses the two fairness algorithms PFA and MCFA to ensure relative equality in the distribution of resource usage among neighbor nodes while maintaining a high delivery success rate, and cost performance close to optimal. While the simplicity is considered as one of PFA algorithm strengths, we show that MCFA uses fewer message replicas compared to PFA and achieves a better success delivery ratio.

The remainder of this paper is organized as follows. Section 2 provides a brief overview of rank-based forwarding techniques and the experimental datasets used throughout the paper in our analysis. Section 3 discusses the fairness vs. efficiency vs. cost trade-offs. In Section 4 we adopt an offline approach to construct forwarding paths while ensuring both fairness and efficiency. In Section 5, we propose a real-time distributed framework for fairness in opportunistic networking, and implement two distributed algorithms that enable existing rank-based forwarding algorithms to be both fair and efficient. We then present the related work in the field of fair and efficient mobile opportunistic forwarding in Section 6. Finally, Section 7 concludes the paper.

2. Background and datasets

In this section we provide a brief overview on rank-based forwarding algorithms, and present the experimental datasets used in our analysis to improve the fairness of these forwarding algorithms. We also briefly describe our evaluation methodology used throughout the whole paper.

2.1. Rank-based forwarding algorithms

Rank-based forwarding techniques represent one of the most promising methods for message forwarding in opportunistic networks. They differ in the type of information used, as well as how it is used, in order to rank nodes in the network. We distinguish between two type of rank-based forwarding techniques; contact-based ranking techniques, where information learned during contact, are used to rank nodes, and social-based ranking techniques, where social interactions between users are used to rank the nodes.

2.1.1. Social-based ranking algorithms

There are three well-known social-based ranking techniques that differ in the type of social metric used:

- Degree-based forwarding: Consists of forwarding messages to socially well connected nodes. Paths are then constructed according to a non-decreasing social node degree rule (more details in [22]).
- Centrality-based forwarding: The main idea behind this technique is that central nodes in social graphs are more likely to socialize with other people and therefore more likely to forward messages (more details in [5,22]).
• PeopleRank: This is a distributed algorithm which ranks nodes in the social graph similar to what PageRank [2] does for web pages – i.e., it measures the relative “importance” of a node in the social graph (more details in [25]).

2.1.2. Contact-based ranking algorithms

In the following, we present two of the most well-known contact-based ranking algorithms in the literature. As the name indicates, they use locally available contact information to rank nodes and decide whether to forward a message when two nodes meet. These two algorithms are:

• Last Contact (LC): Node $i$ forwards messages to node $j$ if $j$ has contacted any other node more recently than node $i$ [7].
• Frequency (FR): Node $i$ forwards the message to node $j$ if $j$ has more total contacts than node $i$ [8].

2.2. Experimental datasets

Our analysis is based on two datasets collected in conference environments [22,4]. In addition to human mobility information, our datasets contain social relationships between the experimentalists. A summary of the corresponding parameters is provided in Table 1.

2.2.1. CoNext07

This dataset [22] contains mobility information and information about the social relationship between the participants during the ACM CoNext 2007 conference. During the experiment, the social networking application indicated when a contact, or a contact of a contact, was within Bluetooth range/neighborhood. This connection neighborhood was then displayed on the user’s device which in turn could add new connections or delete existing connections based on the physical interaction consequent to the application notification.

2.2.2. Infocom06

This dataset [4] was collected with 78 participants during the IEEE Infocom 2006 conference. Experimentalists were asked to carry an experimental device (called iMote) with them at all times. These devices were logging all contacts between participating devices. Questionnaires were given to participants to fill their nationalities, languages, countries, cities, academic affiliations and topic of interests. Based on the forms filled by the experimentalists (e.g., first name, last name, interests, nationalities, affiliation, etc.), we consider, in this paper, a social graph based on users Facebook friendship graph (obtained offline and connecting all the experimentalists). Note that we did not use the implicit social information (i.e., based on interests and affiliation) which is publicly available in CRAWDAD.1

2.3. Evaluation methodology

In mobile opportunistic networks, we are generally interested in delivering data among a set of $N$ mobile wireless nodes. Communication between two nodes is established when they are within radio range of each other. Data is forwarded from source to destination using these opportunistic contacts. We model the evolution of contacts in the network by a time varying graph $G(t) = (V, E(t))$ with $N = |V|$. We assume that the network starts at time $t_0$ and ends at time $T$ ($T$ can be infinite). We call this temporal network the contact graph. Each $G(t)$ describes the contacts between nodes existing at time $t$. Such a time-varying graph model can be obtained from a mobility/contact trace or from a mobility model along with knowledge of radio properties (e.g., radio range).

We evaluate the performance of forwarding algorithms relying on the two previously described data sets; CoNext07 and Infocom06. In our evaluation, used for data analysis throughout the whole paper, we compute the sequence of optimal paths found between any source and destination in the data set. From the sequence of delay-optimal paths we deduce the delay obtained by the optimal path at all time. We uniformly combine all the observations of a trace among all sources, destinations, and for every starting time (the time in seconds when the message $M$ was generated by the source node $S$). In this paper, we assume that contacts between any two nodes are long enough to transfer a message successfully. We present this aggregated sample of observations via its empirical CDF. The detailed computation process could be found in [4]. Compared with previous generalized Dijkstra’s algorithm, this algorithm directly computes representation of paths for all starting times.

3. Fairness vs. efficiency vs. cost

Our discussion of trade-offs between efficiency, fairness, and cost highly depends on the definition of those three metrics. In this section, we first define these terms. We then quantitatively motivate the need to investigate these trade-offs using real experimental traces.

3.1. Definitions

In this paper, we let “efficiency” denote the successful delivery ratio of a given forwarding algorithm within $t$ seconds. Higher efficiency means shorter end-to-end delay

---

1 http://crawdad.cs.dartmouth.edu/meta.php?name=cambridge/haggle
and more successful message delivery in the network; high success rate within a short delay.

We aim to have fairness act as a means to provide users with incentives to collaborate in mobile opportunistic communication—i.e., encourage them to contribute to forwarding messages and remaining longer in the network. This perception to fairness is localized to neighbors; ensuring fairness within node vicinity even if distant nodes incur less cost. We therefore define “fairness” as the relative equality in the distribution of resource usage among neighboring nodes in the network—i.e., a forwarding algorithm is “fair” if the capacity assignment of a given node \( n \) in the network is equivalent to those of \( n \)'s neighbors.

Cost, in addition to efficiency, is a very important evaluation metric in opportunistic networks. Most mobile devices have limited storage and energy supplies. Store-carry-forward algorithms heavily consume these resources, by sending, receiving, storing messages, and by performing computation. We define the “cost” of a forwarding algorithm as the fraction of contacts involved in the forwarding process as compared to flooding.

3.2. Trade-offs

We show in Fig. 1 a summary of previous studies on the fields of fairness, cost and efficiency in mobile opportunistic networks. Researchers give more attention to cost–fairness, and cost–efficiency trade-offs, while the trade-off between efficiency and fairness remains largely unexplored. Within the proposed solutions, rank-based forwarding represents one of the most promising methods for addressing the cost–efficiency trade-off. This solution, however, creates a high potential of dissatisfaction among high ranked nodes that carry a heavier burden compared to others. Therefore, rank-based forwarding does not address the rising concern of fairness amongst various nodes in the network. An absolute fair treatment of users, however, causes significant end-to-end delay and message delivery performance degradation [1,18,20]. Consequently, there is no technique to date that ensures both fairness and efficiency. It is therefore important to consider whether there exists a tradeoff relation between fairness and efficiency. In this paper, we answer the following question in Section 4: Can we fall in the desired trade-off region of Fig. 1?

3.3. Towards absolute efficiency

As discussed earlier, efficiency deals with high potential of dissatisfaction among popular nodes. Let us assume that dissatisfied nodes decide to boycott the forwarding process. We show the impact that these popular nodes have on network performance when they refrain from message forwarding; popular node may be relevant but not critically needed for a network.

Methodology: We use a technique, which we call “node removal”, to exclude nodes from the forwarding process. We study the impact of excluding popular nodes on the overall forwarding performance by applying this technique to the mobility traces we have. We also studied the impact of removing the unpopular nodes. From the sequence of delay-optimal paths in our datasets, we deduce the delay obtained by the optimal path at all time. We uniformly combine all the observations of a trace among all sources, destinations, and for every starting time (the time in seconds when the message \( m \) was generated by the source node \( S \)). We present this aggregated sample of observations via its empirical CDF. We plot the success rate of forwarding algorithms normalized by the success rate of flooding within a given message delivery delay. This technique is a relevant metric used to analyze forwarding algorithms and is discussed with more details in [4].

Fig. 2 plots the empirical CDF of the normalized delivery success rate of the PeopleRank forwarding algorithm (an example of an efficient rank-based forwarding algorithm according to [25]). We have observed similar results for LC and FR algorithms. Fig. 2a shows the impact of excluding the most popular nodes on the successful delivery rate; 10% popular removal consists of removing the 10% most popular nodes from the original dataset (here, popularity rank is given by the PeopleRank values [25]). We show that mobile communication between users may fail with the absence of fairness, where excluding the most popular...
nodes from the forwarding process causes significant performance regress; if only 10% of the most popular nodes boycott the forwarding process, PeopleRank’s success rate performance degrades by roughly 20% within a 10 min timescale.

However, PeopleRank performance shows insignificant regress when we exclude 10% of the most unpopular nodes; only 1.5% less within a 10 min timescale. This result is consistent with the fact that unpopular nodes do not contribute much to the forwarding process as the popular nodes do. Therefore, it is indeed shown that popular nodes in the network are more suitable than others to deliver a given message to its destination.

In Fig. 2b, we plot the distribution of the successful delivery rate of PeopleRank as a function of different popular node removal percentages. As expected, excluding popular nodes from the forwarding process deteriorates the success rate, especially for small timescales. The probability to reach a destination within 10 min drops from 96% to 83% when 5% of the popular nodes are excluded. Moreover, the success rate regress is more severe when the most popular nodes are removed; while we show a 13% regress (from 96% to 83% within 10 min timescale) when only 5% of the most popular nodes are excluded, this regress is only 3% when we exclude 5% of the remaining 90% popular nodes (i.e., we compare 10% and 15% removal nodes curves).

Fig. 2c shows that excluding unpopular nodes from the forwarding process does not seem to greatly impact the success rate of PeopleRank. At times, this removal may even increase the performance when we remove only 5% of the most unpopular nodes. In contrast, removing the most unpopular nodes may, indeed, subtract high end-to-end delays from the overall distribution (i.e., high delays are mainly caused by high waiting times to reach infrequently seen destinations).

As a summary, we observe that popular nodes in the network are more suitable than others to deliver a given message to its destination; selecting preferential relays in forwarding decisions gives better forwarding performance. It is indeed important to further satisfy popular nodes. Providing fairness is therefore crucial since the unfair treatment of users is considered as an disincentive to participation in the communication process.

3.4. Towards absolute fairness

While fairness is our goal, absolute fairness amongst all nodes is not. In this section, we discuss the impact of absolute fairness on the overall network performance. Let us assume an absolute fair allocation of resources across nodes in the network. We perform an offline study of the path availability in our datasets, while each node forwards the same number of messages compared to all other nodes.
in the network. We ensure fairness by balancing cost across nodes. We call this offline fair path establishment technique, the FAIR algorithm. FAIR provides, by definition, a uniform fair allocation of user resources among nodes in the network.

Fig. 3a compares two contact-based ranking algorithms (i.e., LC and FR) to the FAIR algorithm using the Infocom06 dataset. We show how an absolute fair allocation of resources leads to significant performance regression. The probability of success within 10 min decreases from 72% to 43% for Last Contact algorithm and from 60% to 43% for Frequency algorithm. The reason behind this regression is that an absolute fair balancing of cost across nodes causes significant end-to-end delay and success rate performance degradation.

PeopleRank [25] uses a simplistic technique in order to increase fairness across nodes. We study the impact of such techniques on the PeopleRank performance. PeopleRank allows us to tune the amount of the social information used through a damping factor $d$. Mtibaa et al. [25] shows that social forwarding schemes are effective only for very accurate social information. The damping factor is then used to compensate the mismatch of the social interaction between users and their mobility patterns. Consequently, the more fairness we want, the closer to 0 we will chose the damping factor.

In Fig. 3b, we plot the number of forwarded messages per PeopleRank node (nodes are ranked from the unpopular to the most popular node in the network) with regards to the damping factor $d$. This result is normalized by the number of forwarded messages if we run an epidemic forwarding algorithm. In addition, we indicate the normalized success rate (within 10 min) regarding each damping factor $d$ in brackets in Fig. 3b’s labels. We observe that the more unfair PeopleRank is (higher values of $d$), the better performance it can achieve; while PeopleRank algorithm is given the best distribution of resources among the nodes (less than 25% difference between the highest and the lowest used nodes in CoNext07 dataset when $d = 0.4$), it achieves only 75% of success rate performance within 10 min timescale. However, PeopleRank running with $d = 0.9$ gives 23% more success rate performance, and an important resources usage disproportion among the nodes (more than 55% difference). Similar results have been obtained in Fig. 3c for the Infocom06 dataset. To summarize, an absolute fair balancing of cost across nodes causes significant end-to-end delay and success rate performance degradation.

Fig. 3. Impact of fairness on the network performance.

(a) Comparing rank-based algorithms (Last Contact, Frequency) to FAIR success rate and normalized success rate (in brackets) with performance (using Infocom06 data set).

(b) Normalized number of forwarded messages regards of nodes’ rank (CoNext07 dataset).

(c) Normalized number of forwarded messages and normalized success rate (in brackets) with regards of nodes’ rank (Infocom06 dataset).
4. Can we be both efficient and fair?

In this section we quantitatively identify the trade-off relationship between fairness and efficiency. We first provide some intuition regarding the desired fairness we seek while introducing our satisfaction index metric. We then discuss an offline approach to verify, based on our experimental datasets, the availability of paths that would satisfy both efficiency and fairness.

4.1. Desired fairness and satisfaction index

In a mobile opportunistic network, we can roughly divide nodes into three different categories with respect to their popularity ranking: popular, semi-popular and unpopular nodes (as shown in Fig. 4). While ranking nodes in the network may be considered to efficiently forward messages in DTN, it creates a high potential of dissatisfaction amongst nodes since they forward more messages compared to others (as shown by the solid line in Fig. 4). On the other hand, an absolute fair allocation of resources yields poor forwarding performance (dashed line in Fig. 4). Inspired by the previous results shown in Fig. 2, we believe that the desired fairness is different from an absolute fair distribution of resources (optimal fairness). It is indeed important to increase the satisfaction of popular nodes by reducing their cost. Unpopular nodes, however, are unlikely impactful if they were involved in the forwarding process. Therefore, our goal is to further satisfy popular nodes by moving from a situation where popular nodes carry the largest burden in delivering messages to a “fair distribution” of this burden among popular and semi-popular nodes as shown by the red dotted line in Fig. 4.

To reach this desired situation, we need a metric by which we can assess the level of fairness amongst the nodes. We define a satisfaction index \( SI \) as the difference between the message load distribution given by the forwarding process (current fairness) and uniform distribution among nodes (desired fairness). Our goal is to construct paths between any source and destination that verify the condition in the following equation:

\[
\exists C > 0, \quad \forall i \in 1 \ldots n,
SI(N_i) = \text{load}(N_i) - \sum_{j=1}^{n} \text{load}(N_j) \geq -C,
\]  

where \( N_i \) represents a node \( i \) in the network, \( n \) is a total number of nodes, \( \text{load}(N_i) \) is the current load at node \( N_i \), and \( C \) is a positive constant integer. The goal is then to verify if there exists a relatively small positive integer \( C \), such that we can construct paths between any source-destination pair and satisfy Eq. (1).

4.2. Availability of fair and efficient paths

We consider an offline approach to construct forwarding paths that ensures both fairness and efficiency based on global network information. In order to distribute the burden among semi-popular and popular nodes, we change the common forwarding rule used by most rank-based forwarding algorithms: node \( A \) decides whether to forward a message \( m \) when it meets node \( B \) according to this inequality: \( \text{rank}(B) > \text{rank}(A) \). This rule will exclude semi-popular nodes if \( m \) was generated by a high ranked node. We then apply the following rule: \( \text{rank}(B) > \text{rank}(A) - \varepsilon \), where \( \varepsilon \) is a positive number, or a linear function. In this section, our goal is not to give a “magic” number or function of \( \varepsilon \) since this largely depends on the mobility characteristic present in the dataset. However, we discuss the fairness–efficiency tradeoff feasibility in a given dataset.

Fig. 5 plots the normalized number of forwarded messages per node with respect to different values of \( \varepsilon \). We show that when \( \varepsilon = 0.2 \), PeopleRank ensures a better distribution of the burden among popular and semi-popular nodes compared to the original forwarding rule (when \( \varepsilon = 0 \)). On the other hand, the distribution of the number of forwarded messages by unpopular nodes seems rather insensitive to the variation of \( \varepsilon \); it is indeed unlikely to meet these unpopular nodes, and the message will be delegated directly to popular and semi-popular nodes.

We now consider \( \varepsilon \) to be a function where increasing the nodes’ rank \( r \) linearly increases \( \varepsilon \) value. We denote \( \varepsilon(x, r) \) as a function of the rank \( r \), given by: \( \varepsilon(x, r) = x + r + \beta \). The above forwarding rule can then be given by:

\[
\text{rank}(B) > \text{rank}(A) - \varepsilon(x, r) \quad \text{where} \quad \varepsilon(x, r) = x + r + \beta.
\]

Fig. 4. Node categories with respect to their popularity rank.
\[ \text{rank}(B) > (1 - \epsilon) \times \text{rank}(A) + \beta. \]  

In Fig. 6, we test different values of \( \alpha \) and \( \beta \), and plot the satisfaction index \( SI \) with respect to PeopleRank ranking. We show that the new rule given by Eq. (2) ensures a better distribution of the burden among the nodes and achieves a comparable success rate performance (only 1% or 2% less than optimal success rate performance). The new forwarding rule provides a fairly good satisfaction index performance as described in Eq. (1) using a relatively small \( C \) constant (\( C = 8 \) for \( \epsilon(0.3 - 0.1) \), and \( C = 13 \) for \( \epsilon(0.2 - 0.1) \)).

To summarize, our offline data driven approach shows that fair and efficient paths may exist in the network. However, it does not indicate whether these paths can be found efficiently by a distributed algorithm.

5. A real-time distributed framework for fairness-based forwarding

We now propose a real-time distributed framework that maximizes the overall user satisfaction using existing state-of-the-art rank-based forwarding algorithms; it helps rank-based forwarding algorithms utilize potential nodes to forward messages while satisfying the fairness property given by Eq. (1). We call our systematic framework, Fairness-based Opportunistic Networking (FOG). FOG is a framework that enables state-of-the-art rank-based opportunistic forwarding algorithms to ensure a better fairness–efficiency trade-off while maintaining a low overhead. FOG implements different opportunistic fairness algorithms and selects the suitable one with respect to given network properties. We introduce two real-time distributed algorithms; Proximity Fairness Algorithm (PFA), and Message Context Fairness Algorithm (MCFA).

5.1. Proximity Fairness Algorithm (PFA)

In mobile opportunistic networks, resource limitations may prevent us from delivering all messages. A new challenge arises: how to customize message forwarding, in order to bring maximum satisfaction to the end users.

We introduce a Proximity Fairness Algorithm (PFA) shown in Algorithm 1 implemented within FOG that operates as follows: whenever two nodes are within close proximity of each other, they separately run an update process. They first update their relative maximum burden \( \text{max} \) and relative minimum burden \( \text{min} \) per unit time, where a burden denotes the number of messages a node can carry. Afterwards, they decide whether or not to forward a message \( m \) if the following conditions are verified: (i) the encounter node \( N_j \) is the destination node of the message \( m \) or (ii) \( N_j \) is higher ranked compared to \( N_i \) and its burden \( (\text{burden}(N_j)) \) does not exceed the relative average. Finally, the receiver of the message should update its actual burden value.

**Algorithm 1.** PFA \((N_i)\).

1: \( \text{max} \leftarrow 0 \)
2: \( \text{burden} \leftarrow 0 \)
3: \( \text{time} \leftarrow \text{system time} \)
4: \( \text{min} \leftarrow \text{time} \)
5: while True do
6: \( \text{max} \leftarrow \text{max}(\text{max}, \text{burden}) \)
7: \( \text{min} \leftarrow \text{min}(\text{min}, \text{burden}) \)
8: while \( N_i \) in contact with \( N_j \) do
9: \( \text{update}(\text{max}, \text{min}) \)
10: if \( \text{Rank}(N_j) \geq \text{Rank}(N_i) \) AND \( \text{burden}(N_j) < (\text{max} + \text{min})/2 \) OR \( N_j = \text{destination}(m) \) then
11: \( \text{Forward}(m, N_j) \)
12: end if
13: if receiving \((m, N_i)\) then
14: \( \text{burden} = (\text{burden} + 1)/\Delta T \)
15: end if
16: end while
17: end while
Simplicity is considered as PFA’s strength. We note that, while the offline technique uses different parameters to measure the satisfaction index SI among all nodes, PFA as a fully distributed algorithm that relies on local information at the node to estimate the distribution of the burden among neighboring nodes. We believe that nodes are satisfied by comparing themselves to their neighbors and acquaintances and require only relative equality amongst their neighbors in the network.

5.2. Message Context Fairness Algorithm (MCFA)

By limiting the maximum number of transferred messages per node, PFA balances the traffic among the nodes, but does not prevent a popular node from being unavailable (i.e., overloaded) when its help is needed; most popular nodes will always be the most utilized until they become overloaded and stop collaborating in the forwarding process. It is, therefore, important to better balance the traffic among the nodes over time.

We propose a new algorithm that maximizes the overall satisfaction among all users over time. The idea is to utilize the most popular nodes for forwarding if and only if other nodes estimate that they definitely need them. For this purpose, we associate such need with the number of message copies in the network, and propose a new distributed Message Context Fairness Algorithm (MCFA) to maximize overall user satisfaction. MCFA enables cost performance aware message delivery while taking into account the number of message replicas in the networks by computing the utility function to model the user satisfaction derived from sending and receiving each message.

5.2.1. Estimating the number of replicas

While the excessive replication of a given message increases the message delivery ratio [32], it causes a heavy network load. Therefore, the number of message replicas in the network can be used as an estimation of the likelihood of a given message to reach its final destination within a time period \( \Delta t \).

Previous work have estimated the average delay and the number of replicas in the network as a function of the number of nodes and node mobility patterns [34,31]. While these analyses provide important results, their assumptions on knowing the exact total number of nodes and their mobility patterns remain unrealistic for mobile opportunistic networks.

In our work, we use only information provided to a node such as: \( n_b \), the number of hops traveled by a given message \( m \), \( n_c \), the number of copies of \( m \) the node itself has replicated over time, and \( t_l \), and a time-to-live field of each packet that represent the remaining time before erasing the packet. We note that \( t_l \) are computed locally without any need for global clock synchronization.

Next, we assume that \( n_r \) the total number of message replicas in the system at a given time \( t \) exponentially decrease when the number of hops increase. Since, rank-based forwarding algorithms tend to always forward the messages to higher ranked nodes, after forwarding a message across a large number of hops, the number of forwarding opportunities significantly decreases. We estimate this number using an exponential decay \( e^{-\Delta t n_r(m)} \) with a constant rate \( \lambda \) equal to 1.

\[
   n_r(t, m) = n_r^{(t_0, m)} e^{-\lambda n_r(m)} \times \Delta t,
\]

where \( \Delta t \) denotes the time within which \( n_r \) replicas of the same message \( m \) were sent by the given node (i.e., the node estimating \( n_r(t, m) \)). We validate our \( n_r \) estimation using a CoNext07 dataset; the difference between \( n_r \) and the real values does not exceed 8% among the all nodes throughout the duration of the experiment.

5.2.2. MCFA overview

In order to improve fairness in opportunistic networks, we revisit the satisfaction a node derives from sending and receiving messages. We model such satisfaction as a utility function. Such utility quantifies “the need” to forward to the most popular nodes instead of semi-popular ones. The idea is to avoid overwhelming the most popular node with many messages when other nodes can deliver them to their destination in a bounded delay. We define \( U(m, N_i, N_j) \) the utility a node \( N_i \) derives from sending a message \( m \) to another node \( N_j \) as shown in the following equation:

\[
   U(m, N_i, N_j) = \frac{f(b_j)}{RANK(N_j) \times n_r(t, m)},
\]

where \( b_j \) denotes the actual buffer size at node \( j \), \( f \) estimates the remaining buffer size as the difference between \( b_j \) and the maximum buffer size \( max \) (as defined in Algorithm 1 line 6). A pseudo-algorithm of MCFA is given by Algorithm 2; while two nodes become within proximity, forwarding decisions take into consideration the utility \( U \) to decide whether to forward a message or not. We note that if a node meets the destination, the message is always forwarded even though \( SI \ll \text{THRESHOLD} \). Based on an offline study using the described datasets, we set the threshold value in this paper to 0.36. Note that this value can be deduced dynamically over time relying on local network properties.

Algorithm 2. MCFA \((N_i)\).

```plaintext
1: \textbf{while} \( N_i \) in contact with \( N_j \) \textbf{do}
2: \hspace{1em} exchange\((b_j, RANK(N_j))\)
3: \hspace{1em} \textbf{while} \( m \in \text{SendBuffer}(N_i) \) \textbf{do}
4: \hspace{2em} \textbf{if} \( |U(m, N_i, N_j)| \geq \text{THRESHOLD AND} \)
5: \hspace{2em} \hspace{1em} \( RANK(N_j) \geq RANK(N_i) \) OR \( N_j = \text{destination}(m) \) \textbf{then}
6: \hspace{2em} \hspace{2em} Forward\((m, N_j)\)
7: \hspace{2em} \hspace{1em} \textbf{end if}
8: \hspace{2em} \textbf{end while}
9: \hspace{1em} \textbf{if} \( \text{receiving}(m, N_i) \) \textbf{then}
10: \hspace{2em} \hspace{1em} \( b_j = (b_j + 1)/\Delta T \)
11: \hspace{1em} \textbf{end if}
12: \textbf{end while}
```
5.3. Evaluation

While PFA remains simple and does not use additional storage, it may achieve poor performance in particular scenarios. Fig. 7 compares the number of forwarded messages over time using PFA and MCFA by the most popular node using the CoNext07 experiment (see Table 1). We show that PFA tends to send all message towards the most popular nodes causing buffer saturation early. Such saturation will block the most popular nodes from sending and receiving messages even when they are critically needed to deliver a particular message later in time. On the other hand, MCFA uses the most popular nodes only when they are critically needed or not heavily used. MCFA, therefore, maintains a constant message delivery rate in order to reach critical nodes when needed.

So far, we have been assuming that node rank remains constant during the experiment, and our tests use the rank of the node at the end of the experiment. We, therefore, study the evolution of ranking techniques over time. We plot in Fig. 8 PeopleRank values of four representative nodes (unpopular, popular and semi-popular nodes) over time using the CoNext07 experiment. We show that ranks may change only at the beginning of the experiment, and stabilizes after 6–7 h. We note that we choose to plot the nodes that considerably change their ranks over time; other nodes compute their final rank after only 20 min to 1 h. To summarize, after the initial bootstrapping phase, which last for a short period of time in comparison with the network lifetime (i.e., many days or months), node rank remains relatively constant, and socially popular nodes remain popular throughout the duration of the experiment.

Next, we evaluate the performance of three state-of-the-art social forwarding algorithms (PeopleRank, LC, and FR) relying on the two previously described datasets (Info-com06, and CoNext07). We also use a data-driven artificial trace to evaluate the scalability of PFA and MCFA in large scale social networks.

5.3.1. PFA and MCFA in small social environments

We adopt a real-trace driven approach to analyze and evaluate the trade-off between the efficiency and fairness of PFA and MCFA in order to provide a more realistic evaluation platform.

Next, we compare our distributed algorithms, PFA and MCFA, performance to the offline approach based on the real mobility traces shown in Table 1. In Fig. 9, we compare the satisfaction index $SI$ and the message distribution among nodes using PFA and the offline approach. In addition, we show the success rate of these forwarding algorithms shown in the legends of each figure. We show that PFA outperforms all the offline rules we have tested and achieves one of the best $SI$/success rate trade-offs. PFA also ensures a more fair distribution of the burden among the nodes; the satisfaction index $(SI)$ of popular and semi-popular nodes remain close to zero and verify Eq. (1) with a relatively small constant $C = 3$. PFA is explicitly prohibiting forwarding messages when the burden is not fairly distributed among neighbors; popular nodes are more likely to meet all other nodes as well as each other. The burden is therefore fairly distributed among them. We note that, PFA is also efficient given that it keeps the success rate close to the optimal (2% less compared to the original PeopleRank success rate performance).

We compare in Fig. 10 the performance of our algorithms PFA and MCFA. We show that while PFA ensures the best load balance among nodes, MCFA uses the most popular nodes more often in order to reduce the end-to-end delay and achieve a better success rate; MCFA reduces the load of most popular nodes by 10–20% while keeping the success delivery ratio near to optimal (success rate degradation of only 1%).

5.3.2. Scalability

Owing to the lack of large-scale experimental data sets, we artificially create San Francisco trace, a data set that uses the San Francisco taxi cab trace [27] coupled with three
human mobility traces in order to represent three sub-communities of the San Francisco area such as the airport, downtown, and the sunset areas. To the best of our knowledge, this is the largest data set that captures human mobility contacts as well as human social properties in large scale networks.

The San Francisco taxi trace contains mobility traces of taxi cabs in San Francisco. It contains GPS coordinates of approximately 500 taxis collected over 30 days. Each trace contains the reported time and location for each taxi. We incorporate traces for the duration of 3 days, interpolate the movement of the cabs, and then generate the contacts between these taxis. We assume a contact has occurred when a taxi comes within proximity of 100 m from another taxi. The contacts trace thus contains the starting time stamp of when the contact has occurred and the ending time stamp of when the contact has finished. It also contains the IDs of the two cabs that happened to be in contact with each other during that time.

We artificially incorporate three real human mobility traces in the three different areas of San Francisco city: Infocom06, CoNext07, and Dartmouth campus (see Table 2 for more details). Taxis are moving between these areas. Contacts between taxis and nodes within an area are added based on the same contact time and inter-contact time distribution [4] of the corresponding area.

We evaluate the performance of PFA integrated with each of the three rank-based forwarding algorithms in large scale networks relying on the San Francisco modified dataset. Figs. 11–13 plot for each rank-based algorithm (a) the normalized number of forwarded messages and (b) the normalized success rate performance using the original algorithm and the PFA-based extension. We show that PFA and MCFA ensure better distribution of resources among nodes while keeping the success rate performance close to optimal. However, while PFA and MCFA give almost near load distribution using PeopleRank, LC and FR rankings, MCFA-based extensions achieve better success delivery ratios within a given delay d.

For example, we show that PFA-FR (i.e., PFA-based extension of FR algorithm) ensures fairly distributed allocation of the forwarding burden among nodes with a minimal decrease in success rate by only 4% within a 10 min timescale compared to the original FR success rate. MCFA-FR uses slightly more often the most popular nodes (2–3% more than PFA-FR), however it achieves considerably better success delivery ratio with almost 5% improvement compare to PFA-FR within a 10 min delay.

Finally, we note that MCFA-PeopleRank and PFA-PeopleRank yield a significantly high improvement compared to the FAIR algorithm (described in Section 3) in large scale networks: +20% to 25% in the success rate within a 10 min timescale. Using PeopleRank, highly ranked nodes carry the highest burden since they are more likely to forward...
a message to its destination. PFA-PeopleRank is therefore ensuring an acceptable trade-off between efficiency and fairness in large scale networks.

5.3.3. Cost

Cost, in addition to end-to-end delay and success rate metrics, is a very important evaluation metric in opportunistic networks. We define the cost of a forwarding algorithm as the fraction of contacts involved in the forwarding process. PFA and MCFA are designed to fairly distribute the burden among semi-popular and popular node. However it was shown in previous work [25] that efficiency/cost trade-off may be achieved when most popular node carry the largest burden in delivering messages. In this section, we study the impact of the fair distribution of the burden given by PFA and MCFA on the overall cost.

Fig. 11. Comparison of PFA, MCFA implemented on top of LC algorithm (using the modified SF dataset).

Fig. 12. Comparison of PFA, MCFA implemented on top of FR algorithm (using the modified SF dataset).

Fig. 13. Comparison of PFA, MCFA implemented on top of PeopleRank algorithm (using the modified SF dataset).
Fig. 14 compares the cost of PFA-PeopleRank, MCFA-PeopleRank to FAIR (absolute fairness) and the original version of PeopleRank using three different data sets (Info-com06, CoNext07, and modified SF data sets). The original version of PeopleRank performs well and achieves the best normalized cost in all scenarios we have tested. PeopleRank, as a social-based ranking algorithm, was designed to provide a decent cost/efficiency trade-off. We note that rank-based forwarding algorithms can be fair and keep the cost and the performance close to the optimal. PFA-PeopleRank outperforms the FAIR algorithm while using only 2–6% more replicas of the message m compared to the cost of the original PeopleRank algorithm. We also note that MCFA achieves one of the best trade-offs between efficiency, fairness, and cost; it ensures fairness among most of the nodes in the network while keeping the cost and success delivery ratio close to optimal. MCFA uses 1–4% less message replication compared to PFA and achieves better success delivery ratio. However, it has been shown in the previous section that MCFA may use the most popular nodes only “if needed” to achieve considerably better success ratio.

Finally, it is crucial to be aware that, while the total number of replicas in the network increases, the cost increase per node is not as significant as shown in the previous results. Moreover, as the number of nodes in the network grows, reflecting more realistic deployment environments, the increase in cost of MCFA algorithm becomes significantly smaller. In fact, it has been shown that existing state-of-the-art rank based forwarding algorithms present many drawbacks in large scale networks where multiple sub-communities may exist [24]. MCFA uses its estimation of the number of message replicas to fairly distribute the burden among more semi-popular nodes which increases the probability to reach distant destinations.

6. Related work

Most interactions between people rely on the establishment of the sense of fair treatment. Computer networking communication, and more particularly peer-to-peer file sharing applications and services take into consideration the fair treatment of users. Fairness is therefore particularly important and challenging since it is considered a major incentive for peer-to-peer service usage in today’s Internet. Theses challenges are more critical in infrastructure-less wireless networks given the lack of centralized and trusted mechanisms that control the fair treatment of users. With the recent shift in research interest from centralized services to distributed services in mobile communication, fairness is becoming an interesting field of investigation in many research topics such as resource allocation, congestion control, and network routing.

In peer-to-peer mobile networks such as mobile ad hoc networks (MANETs) [6,17,11,16], wireless sensor networks [30,15,12], or delay tolerant networks (DTNs) [14,19], multi-hop wireless communication between users may fail with the absence of the sense of fairness between participating nodes. In DTNs, a device has to decide whether or not to forward data to an intermediate node that it encounters. Such forwarding decisions are typically guided by the desire to reduce the number of replicas of data items in the network to conserve bandwidth as well as by the desire to reduce end-to-end delay.

Current forwarding techniques in DTN are generally designed to efficiently, and excessively over-use popular nodes to guide and improve forwarding decisions [7,8,28,1,19,21,9,29]. Rank-based forwarding techniques currently represent one of the most promising methods for addressing the message forwarding challenge [5,25,3]. Nodes in these techniques are ranked based on their social profiles or contact history to identify those that have a higher probability of successfully forwarding the message to the destination. While these techniques have demonstrated great efficiency in performance [5,26,25,3], they do not address the rising concern of fairness amongst various nodes in the network. Higher ranked nodes typically carry the largest burden in delivering messages, which creates a high potential of dissatisfaction amongst them. Providing fairness is then an important networking goal since the unfair treatment of users is considered as a disincentive to participation in the communication process. It has been shown that ranking-based forwarding algorithms provide good performance relying on identifying and over-using popular nodes in the network [5,26,25,3]. Such well ranked nodes are more likely than others to deliver a message to its destination in a shorter delay. As a direct consequence, an absolute fair treatment of users causes a significant end-to-end delay and message delivery performance degradation. It is then primordial to consider whether there exists a tradeoff relationship between fairness and efficiency.

Fair sharing of resources has been largely studied in the context of classical networks. Previous work has been inspired by the well known max–min fairness [33] or Jain’s fairness index [13] in order to improve a fair allocation/scheduling of resources in the Internet. In the context of wireless networks, researchers use the link quality variation of access points to maximize aggregate throughput [18,20]. In the context of DTNs many assumptions are made regarding resource constraints (storage and bandwidth), and strictly adhere to max–min fairness for
to PFA and achieves a higher success delivery ratio. Considered as one of PFA algorithm strengths, we have cost performance close to optimal. While the simplicity is nodes while maintaining a high delivery success rate, and fairness algorithms PFA and MCFA to ensure relative equal-treatment of users also yields inefficient communication. In this paper, we have introduced two real-time distributed algorithms: Proximity Fairness Algorithm (PFA) and Message Context Fairness Algorithm (MCFA).

We have adopted a real-trace driven approach to study and analyze the trade-offs between the efficiency, cost, and fairness of rank-based forwarding algorithms in mobile opportunistic networks. Our preliminary results show that mobile opportunistic communication between users may fail with the absence of fairness. Furthermore, an absolute fair treatment of users also yields inefficient communication performance. Our analysis shows that FOG uses both fairness algorithms PFA and MCFA to ensure relative equality in the distribution of resource usage among neighbor nodes while maintaining a high delivery success rate, and cost performance close to optimal. While the simplicity is considered as one of PFA algorithm strengths, we have shown that MCFA uses fewer message replicas compared to PFA and achieves a higher success delivery ratio.

7. Conclusion

Our work addresses the rising concern of fairness amongst various nodes in mobile-based opportunistic networks. Rank-based forwarding algorithms are typically designed to reduce the number of data replicas in the network to conserve bandwidth and reduce end-to-end delay. In these algorithms, higher ranked nodes carry a much heavier burden in delivering messages, which can create high levels of dissatisfaction amongst them. In this paper, we have proposed FOG a real-time distributed framework that maximizes the overall user satisfaction using existing state-of-the-art rank-based forwarding algorithms; it helps rank-based forwarding algorithms utilize potential nodes to forward messages while satisfying fairness properties. FOG implements different opportunistic fairness algorithms and selects the suitable one regarding given network properties. In this paper, we have introduced two real-time distributed algorithms: Proximity Fairness Algorithm (PFA) and Message Context Fairness Algorithm (MCFA).

References

Abderrahmen Mtibaa, Ph.D., is a postdoctoral research associate in the School of Computer Science at Carnegie Mellon University in Qatar. Graduated on June 2010 from the University of Paris VI and Technicolor Paris Research Lab. His current areas of interest include mobile opportunistic networks/DTN, wireless and ad hoc networks, mobility models, protocol design, routing/forwarding, network communities and social networking.

Khaled A. Harras is currently an Assistant Teaching Professor in the Department of Computer Science at Carnegie Mellon University. He is the founder and director of the Networking Systems Lab and currently resides at CMU’s campus in Qatar. Khaled’s main research interests include delay and disruption tolerant networks, specifically protocol and architectural challenges in extreme networking environments, multi-media wireless sensor networks, social pervasive systems, and multi-interface networking and communication. He is a member of IEEE and the ACM.