Fair Content Dissemination in Participatory DTNs

Afra J. Mashhadi[a], Sonia Ben Mokhtar[b], Licia Capra[a]

[a]*Dept. of Computer Science, University College London, Gower Street, London WC1E 6BT, UK,*
{A.JahanbakhshMashhadi | L.Capra}@cs.ucl.ac.uk
[b]*LIRIS, INSA de Lyon, 20 Avenue Albert Einstein, 69621 Villeurbanne, FRANCE,*
sonia.ben-mokhtar@liris.cnrs.fr

Abstract

Thanks to advances in the computing capabilities and added functionalities of modern mobile devices, creating and consuming digital media on the move has never been so easy and popular. Most of the DTN routing protocols proposed in the literature to enable content sharing have been exploiting users' mobility patterns, in order to maximise the delivery probability, while minimising the overall network overhead (e.g., number of message replicas in the system, messages' path length). Common to all these protocols has been the assumption that devices are willing to participate in the content distribution network; however, because of battery constraints, participation cannot be taken for granted, especially if the very same subset of devices are continuously selected as content carriers, simply because of their mobility properties. Indeed, we demonstrate that state-of-the-art DTN routing protocols distribute load in a highly unfair manner, with detrimental effects on delivery once the assumption of unconditional participation is lifted. To overcome this limitation, we propose a load-balancing mechanism whereby nodes maintain local estimates of network workload, and use them to direct traffic towards the least loaded portion of the network. We implement the mechanism on top of a source-based DTN routing protocol, and demonstrate, by means of simulation using a variety of real mobility traces, that high delivery is now achieved without compromising fairness.

Keywords: load-balancing, social-based computing, content dissemination, delay-tolerant networks

1. Introduction

Mobile devices have undergone a major evolution. The new generation of mobile phones (e.g., iPhone, Android-powered devices, Blackberry, etc.) has seen their computing and memory capabilities grow according to Moore's Law. A variety of functionalities have become a commodity, including high-resolution cameras, MP3 players, GPS receivers, as well as multiple network interfaces of increasing bandwidth (e.g., 3G, Wi-Fi, Bluetooth 2), thus enabling users both to create and to consume content on the go.

There exist many scenarios where such content is appealing to a *local* community of users only: consider, for example, promo videos of events happening in town (e.g., festivals, street theaters, cabarets), music recordings of new bands performing in the neighbourhood; electronic fliers about activities organised by university clubs and societies (e.g., sport races, debates, parties), and so on. In metropolitan cities like London, an estimated average of 5,000 social events take place every day; however, only (less than) half of these are being listed on popular websites (thus accessible via 3G networks from users' mobile phones). The remainders are being advertised by word-of-mouth, posters affixed in given areas, hand-distributed fliers, and the like. In these cases, where the people producing content, and those willing to receive it, tend to be living in the same geographical area, content distribution can more effectively happen by means of Delay Tolerant Networking (DTN) protocols. These protocols exploit the freely available *local* networking capabilities of mobile devices (e.g., Bluetooth 2) to opportunistically distribute content during periods of colocation. The launch of Wi-Fi Direct[1] represents a major step forward in this direction, enabling users to transfer content quickly and easily with other users in reach, using the Wi-Fi network interface on their mobile phone, even when a Wi-Fi access point is not available.

A variety of DTN protocols has been proposed in recent years (e.g., [1, 2, 3, 4, 5, 6, 7]); common to these approaches is the idea of reasoning about users' mobility when selecting content carriers, in order to achieve high delivery, while keeping the overall network overhead (i.e., number of message replicas) low. As battery constraints on mobile phones are not expected to be significantly lifted in the coming years [8] (i.e., Moore's Law will apply to the miniaturisation of battery size, rather than increasing its lifetime), energy efficiency has become an important topic for this type of participatory networks. New protocols [9, 10] have been proposed with the aim of saving up energy, by further limiting the number of message replicas in the network. However, reducing the *overall* network overhead does not imply this is done *fairly* across all nodes involved in the content distribution network. Indeed, these approaches reduce the overall network overhead by exploiting users' mobility patterns and/or by reasoning on network topology; as our research demonstrates (Section 2.1), this leads to a small subset of nodes to be repeatedly

[1] http://www.wi-fi.org/news_articles.php?f=media_news&news_id=909

selected as content carriers over and over again, leading to highly unfair workload distribution. As these nodes will inevitably see their battery drain very quickly, they are more likely to cease participating in the content delivery network, with detrimental effects on delivery overall.

In this paper, we present a new load-balancing mechanism for DTN that, once integrated with source-based routing protocols, achieves *fair workload distribution over time* without compromising delivery. The mechanism consists of the following two key components:

1. *Load Prevention*: each node locally estimates the load of other nodes in the network, by monitoring the headers (where next-hop carriers are listed) of the messages that pass through it. When such node creates a new message, it source-selects a path that tries to reach its destination while exploiting the least-loaded portion of the network;
2. *Load Alleviation*: local estimates of network load will never be 100% accurate. Even if they were, when messages are injected in bursts, and/or when the nodes' inter-contact time is long, prevention alone does not suffice. We thus add a load alleviation component, whereby each device locally monitors its own participation in the network, ceasing to collaborate for some periods of time, in order to recover from periods of heavy load.

The remainder of the paper is structured as follow: we begin with a review of the state-of-the-art in DTN research, and follow up with the results of a comparative evaluation of some of the latest generation routing protocols, demonstrating how they *all* cause heavily unfair workload distributions (Section 2.1). We then illustrate how the problem of load balancing has been tackled in other areas, tracing back works from which we draw inspiration, whilst at the same time highlighting why they could not be straightforwardly applied in our domain (Section 2.2). In Section 3 we present our approach: we begin with a detailed description of the model and its two key components, that is, load prevention and load alleviation (Section 3.1), and continue discussing a specific implementation, realised on top of the Habit source-based routing protocol (Section 3.2). In Section 4, we report the results of our simulation study, conducted using a variety of real mobility traces, demonstrating how fairness can indeed be achieved without compromising delivery. We then conclude the paper, discussing our ongoing and future work (Section 5).

2. Background

2.1. Motivation

In the past decade, a variety of routing protocols have been proposed to support content dissemination in DTNs. The very first generation of routing protocols focused almost entirely on achieving high delivery; they did so by replicating (flooding) messages in the network, while relying on incidental deliveries caused by opportunistic encounters (e.g., [11, 12]). However, the traffic they generated caused significant network overhead; to strike a balance between delivery and overhead, probabilistic routing schemes were then developed so to control flooding (e.g., [3, 1, 13, 14, 15, 16]). These protocols were mainly evaluated by means of simulation, using synthetic movement traces that followed the Random Waypoint Model [17]. However, as real and large-scale traces of human mobility started to be collected, scientists demonstrated that human movement is not random at all (e.g., [18, 19, 20, 21]); rather, it can be predicted to a large extent (e.g., [22, 23]). A new generation of DTN protocols has then been proposed, that reasons upon human mobility patterns, in a quest to better trade delivery with overhead (e.g., [1, 24, 25, 7]). Whilst largely successful in attaining this goal, all DTN routing protocols heavily rely on the voluntary participation of nodes in the network to forward content. We argue that such an assumption is unrealistic in the context of human DTNs, as such networks are formed by mobile devices with rather limited battery; if a device is asked to forward many messages in a brief period of time, it will deplete its battery at unexpected speed, causing its user to most likely cease participation in the content delivery network altogether.

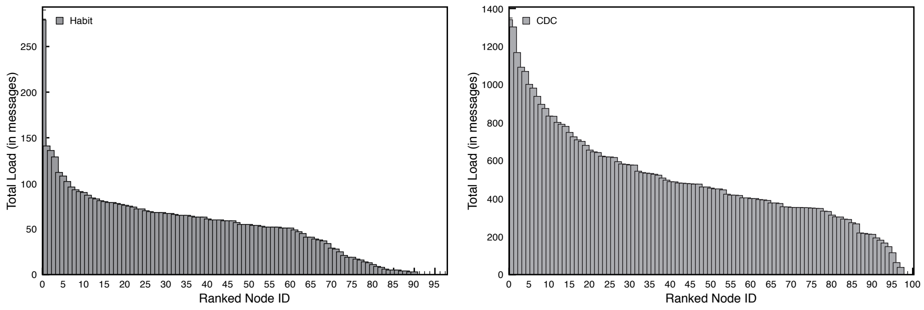
To investigate the magnitude of the problem, we have conducted the following analysis: we have taken real mobility traces of two typical DTN settings: 100 MIT staff and students, moving around the university campus, for a period of nine months [26], and 100 taxicabs, moving within the San Francisco bay area, for a period of one month [27]. In both cases, nodes were posting messages to a varying number of recipients and at a varying publication rate (both parameters, i.e., number of message recipients and rate of publication, were set based on real data, as further discussed in Section 4). Two latest-generation DTN protocols were used to distribute these messages: *Habit* [7], a routing protocol were nodes exploit both social and mobility network information to *source-compute* the whole delivery paths; and *Change in Degree of Connectivity* [1] (CDC), a mobility-based protocol where nodes select as *next* carrier the node who exhibits higher popularity (i.e., who encounters the largest number of distinct nodes in a given period of time). In this experimental setup, the maximum battery lifetime of a device has been modelled as the number of messages the device can *forward* in a given time period; we call this limit the *drainage threshold*. If this threshold is reached, the device ceases participation altogether, that is, it stops forwarding messages for others. Note, however, that we still allow devices to send the messages they produce, as well as to receive the messages they are interested in (selfish behaviour). In so doing, we abstract devices' participation in the content delivery network as the number of messages they forward within a time period; this provides a simple yet realistic indication of energy consumption (communication has been measured to be the highest energy draining factor in mobile devices [28]). To give a flavour of how much actual data can be transferred for different drainage thresholds, we have performed the following calculation: we have considered the HTC Fuze smartphone, whose overall battery capacity is 1340mAh, and assumed that, based on a light usage of the phone, people are willing to recharge their phone at most twice in a period of 5 days (thus leading to an overall capacity of 3 * 1340mAh = 4020mAh); we have then assumed that, of this capacity, users are willing to devote at most 50% (i.e., 2010mAh) to opportunistic content forwarding. Based on the energy consumption study presented in [29] for data transfers using Wi-Fi on the same HTC Fuze smartphone, we have then calculated the maximum number and size of messages that a node can send over a period of 5 days. The table below presents these values:

Number of Messages	Message Size (Type)
300 / 5 days	800KB (Text)
150 / 5 days	1.6M (Picture)
75 / 5 days	3.2M (Music)
35 / 5 days	6.4M (Video)

Note that the reported values are quite optimistic, and only representative of scenarios where users make a rather light use of their mobile phone (e.g., for texting and calling, so that they only need to recharge every 2 days), and are thus willing to devote up to 50% of their battery to content dissemination. However, users who make a more advanced use of their device (e.g., for browsing, social networking, emails, etc.), are unlikely to devote that much battery to DTN content sharing, as they already have to recharge their phone once per day just to support their normal use[2]. The number of messages that these users are willing to forward is likely to be less than what reported in Table 2, thus making the workload distribution problem even more severe.

In our analysis, we gave each node the same drainage threshold, and assumed all messages count the same in terms of consumption (e.g., they are media files of comparable size/type). Using this setup, we have measured the workload distribution among nodes in the network. Figure 1a and 1b illustrate the results for Habit and CDC respectively, when considering the taxicabs traces (very similar results were obtained when considering the MIT traces, and are thus omitted). As shown, the load-per-node varies considerably in both cases, with a few nodes performing significant amounts of work, while others contributing very little. While it can

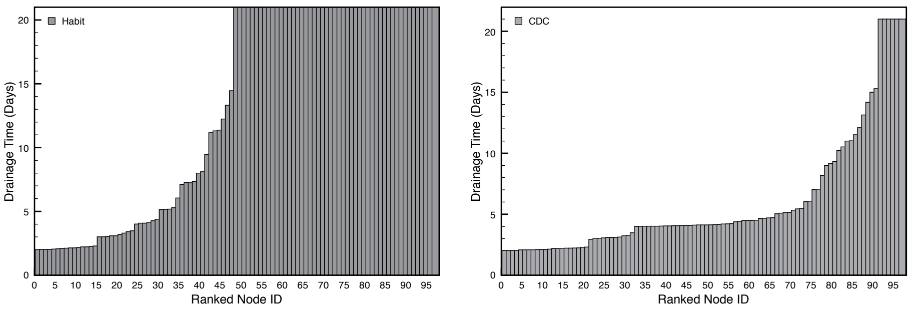
[2] http://reviews.cnet.com/smartphones/htc-fuze-at-t/4505-6452_7-33378478.html



(a) Habit with threshold $\frac{50msgs}{5days}$

(b) CDC with threshold $\frac{100msgs}{5days}$

Figure 1: Load distribution



(a) Habit with threshold $\frac{50msgs}{5days}$

(b) CDC with threshold $\frac{100msgs}{5days}$

Figure 2: Life expectancy

be expected that those nodes sitting at the very edges of the network will always contribute little to the overall content delivery, a fairer load distribution should be attained among all others, in particular, relieving the most central nodes from part of their load. Note that, while in Habit a threshold of 50 messages in a 5-day period was used, for CDC we used a much higher one (i.e., 100 messages per 5-day); this is because CDC makes an aggressive use of colocated devices to deliver content (i.e., via multiple message replicas), causing the whole network to reach the (lower) threshold very rapidly, thus ceasing participation immediately.

As the above experiment confirms, state-of-the-art DTN protocols that reason on node mobility patterns to disseminate content distribute load very unfairly. This would not be a significant problem per se, if the load were distributed over a long period of time, so that the drainage threshold was rarely reached. Unfortunately, this is not the case. We have conducted a second experiment where, under the same simulation setup, we have measured the number of days nodes continue distributing content, before the drainage threshold is reached. The results are shown in Figure 2a and 2b for Habit and CDC respectively (once again for the taxicab traces). As shown, there is a significant number of nodes that cease participation in the very first day of the simulation, as they are immediately selected as content carriers for many messages; by the end of the simulation period, 50% of nodes using Habit have reached the drainage threshold and left, and the percentage becomes as high as 90% for more aggressive protocols like CDC. This has immediate and detrimental impact on content delivery. It is thus clear that, for DTN protocols to become a viable means to share content, they must be designed from the outset to spread load more fairly, while not compromising delivery.

2.2. Related Work

The problem of unfair load distribution in networked systems has been the subject of extensive research, from Internet-based settings (e.g., [30, 31, 32]), to more challenged environments, such as mobile ad hoc networks (e.g., [33, 34, 35, 36, 37]). In MANETs, the high density of network connections, coupled with their relative stability, made it possible for routing protocols to use a feedback mechanism in order to reason upon, and equally spread, the network workload. For example, the approach presented in [35] relies on probing packets, which are sent periodically to each different disjoint path toward a given destination; based on their round-trip time (RTT), the path delay is estimated and the congested nodes identified and avoided in the future. Similarly, in MALB [36], each source sends a probe message to its destinations; nodes receiving the probe reply, also stating their battery level, thus enabling the source to estimate both congestion and available power resources along each path. However, while successful in a MANET setting, these approaches cannot be directly applied in DTNs; in fact, they rely on both the probes and the feedback packets to be delivered quickly, so that the information they carry about the network is fresh and up-to-date. In DTNs, however, this is usually not possible, as the network is much sparser, and nodes have a much longer inter-contact time, making probe and feedback mechanisms not suitable.

Most of the ongoing effort in the DTN research community is being directed towards the design and realisation of protocols that lift the assumption of unquestionable user participation, by means of fair resource allocation schemes. To date, the resource that has been investigated the most is storage. In [38], for example, the effect of users selfishly (or maliciously) refusing to contribute their buffer to the common pool of network resources is studied, to quantify the effect such behaviour has, in terms of delivery, on the content distribution network. To tackle this problem, a technique based on prioritising messages by relying on authenticating users is introduced. [39] similarly focuses on buffer management and addresses the problem of the trade-off between the cost associated to receiving a piece of content (in terms of resource consumption) and the value of the content to the end-user. [40, 41, 42] further investigate the issue of congestion in DTNs, tackling the problem of storage exhaustion. In [43] and [44], approaches that tackle the resource allocation problem in rural villages (as opposed to urban settings) are presented: [43] focuses on storage allocation, while [44] focuses on bandwidth allocation, both with a similar aim of minimising message transit delay, while achieving fairness. In RAPID [45], a utility-based approach, is introduced which can optimize delay-related metrics (e.g., average delay, missed deadline), by treating routing as a resource allocation problem; the uniqueness of the protocol lies in taking constraints of both bandwidth and storage into account, in order to determine how packets should be replicated in the system. ORWAR [46] is another approach that also reasons in terms of utility, but focusing on network-wide optimization, in order to minimise the probability of partially transmitted messages.

All the approaches above focus on storage and/or bandwidth allocation. To our knowledge, the only attempt to achieve fairness in DTNs, while reasoning on available battery, is FairRoute [47], a routing protocol that uses concepts from social sciences (e.g., interaction strength, social status) to select message carriers. By means of simulation using the MIT Reality Mining movement traces, the authors demonstrate that their protocol does achieve more fairness than protocols such as epidemic and SimBet [25], a routing protocol that chooses intermediaries based on betweenness centrality and probability of future interactions. However, one important dimension that FairRoute neglects is *time*: while it distributes load fairly over the *simulation time as a whole*, devices' battery may still drain very rapidly when short periods of intense connectivity follow long periods of disconnection. This has direct impact on delivery, as well as long-term node participation, as we shall demonstrate in Section 4.

In the next section, we present a load-balancing mechanism that achieves fair content dissemination continuously over time, *regardless* of the inter-contact time distribution between nodes in the network.

3. Load-Balancing in Participatory DTNs

Our approach to content dissemination in DTNs aims at distributing load fairly, while still achieving a high delivery ratio. Section 3.1 presents the load-balancing model we propose to achieve this goal, while Section 3.2 illustrates a specific realisation of the model within a DTN source-based routing protocol.

3.1. Model

Our load-balancing model consists of two parts: *Load Prevention*, whereby each node locally estimates other nodes' workload, in order to source-select paths that rely on the less-loaded part of the network; and *Load Alleviation*, whereby each node monitors the traffic it is asked to forward, reacting to bursts by means of a controlled forwarding mechanism that preemptively reduces participation, without reaching drainage. While these two components are similar, in spirit, to Internet-based solutions (e.g., [31, 32]), their mechanics are rather different: in fact, the lack of end-to-end connectivity, which characterises DTN settings, implies these feedback-based mechanisms cannot be called upon.

3.1.1. Load Prevention

In order to distribute load on nodes fairly, two challenges need to be faced: first, the workload of nodes must be locally estimated, without relying on end-to-end shared knowledge; second, based on this knowledge, message routes must be chosen so to effectively spread the load fairly, whilst not affecting delivery.

Challenge 1 - Load Detection. Creating accurate estimates of nodes' load is a challenging task in a DTN setting: in fact, load varies dynamically over time, and if one has to use such information to prevent overloading, only very up-to-date information is of value. Disseminating load information in a DTN setting thus makes little sense, as such information does not tolerate delay.

In source-based DTN protocols, however, there is a great deal of information already encoded in the headers of the messages that can be locally exploited to estimate the network workload. Each node x locally maintains an *Estimation Log*, which logically consists of a table with one row for each known node in the network, and one column for each *slot* in a given *observation period* (e.g., with a slot of one day over an observation period of five days, there would be five columns). Each cell in the log records how many times a known node y has been used as carrier in that slot (e.g., on that day). The counter is updated in two circumstances: (1) node x creates a new message and computes source-based paths for its delivery, containing node y at some point as intermediary; (2) a message is routed through x, containing y in its header as another intermediary somewhere along the path. Note that the Estimation Log is not time-accurate: in fact, although the log is increased when a message is either created on x or when it reaches x, the intermediary y could indeed be affected by this load at another point in time. Nonetheless, given that it is not possible to know exactly when the message will be routed through y, this technique provides valuable information to be used during load distribution. Moreover, it is not the load recorded in a single slot, but an *aggregated* load value, computed over the whole observation period, that is used, as described next.

Challenge 2 - Load Distribution. Source-based routing protocols enable nodes to retain control over the routes followed by the messages they produce. Such property is very attractive as we can leverage it to build a load-aware reasoning scheme, whereby sources inspect the estimated load of intermediaries, and consequently select paths that distribute load more evenly. More precisely, we associate to each path an estimate of its *current overall load*. This value, computed by consulting the Estimation Log, is the sum of the load values recorded for each intermediary node in the path, and for the past observation window ΔT. A path is then *probabilistically selected* based on its *expected utility*:

$$Util_{path} = 1 - Load_{path} \tag{1}$$

where $Load_{path}$ is a value in the range $[0, 1]$, representing the current load of the path, normalized over the most loaded path found. Once a path has been selected, the source node logs its intermediaries in the Estimation Log.

3.1.2. Load Alleviation

We claim that, no matter how good the load prevention technique is, nodes can still become overloaded if the inter-contact time between nodes is long, and the system is under a high rate of message publications. In such circumstances, nodes may end up storing many messages for various destinations which are less frequently encountered (i.e., high inter-contact time); if such nodes are then encountered in a relatively short time period, then an enormous amount of traffic is generated in a very short timeframe, causing local battery to rapidly exhaust (i.e., the node's actual load will hit the drainage threshold).

The load alleviation component has been developed to monitor nodes' actual load, enabling them to step back from participation in relaying for others when a critical load threshold is reached, and giving them enough time to recover before resuming participation. Each node locally keeps track of the actual number of messages it has forwarded for each slot (e.g., each day) within a given observation period (e.g., $\Delta T = 5$ days); if the total number of messages forwarded within an observation period reaches a certain *Critical Boundary*, the node temporarily stops relaying messages, until it is safely out of the critical zone. The Critical Boundary is defined as a percentage of the Drainage Threshold; for example, if the Drainage Threshold is 100 messages within 5 days, and the Critical Boundary is 90%, then the load alleviation component restrains a node from further message forwarding once *Critical Boundary* $\times$ *Drainage Threshold* $= 90$ messages have been sent in the past $\Delta T = 5$ days. Due to the sliding of the ΔT observation window, the refuse-to-relay is only temporary, and once enough time has passed (i.e., once their load during the last ΔT has fallen under the Critical Boundary), they resume participating in the content delivery protocol.

3.2. Realisation

In order to evaluate the fairness that our load-balancing model brings, we have implemented it on top of a source-based routing protocol called Habit [7]. Habit source-computes paths with the goal of making content flow through nodes who are themselves interested in receiving such content, thus minimising the messages relayed to uninterested intermediaries. To achieve this goal, each device logs and processes colocation information with other devices, in order to identify potential regularity in human movement, and to be able to predict future encounters. Regularity information is dynamically propagated over n-hops upon encounters, thus enabling each node to locally build their own view of the *physical network*. Users' interest profiles, containing information about what other users they want to receive messages from, are also propagated n-hops away during encounters, so that each node builds a local view of the whole *social network* too. These two layers are then combined to compute routes that content should follow, in order to have a high probability of reaching all interested nodes, while crossing a minimum number of uninterested carriers. Paths are computed at the source, and information about the carriers is stored in the header of each message. Intermediaries strictly adhere to the next-hop information recorded in the message, unless the intended final recipient is encountered first, in which case the message is simply handed over to its final destination.

Habit has been evaluated in terms of precision (i.e., the proportion of relevant messages each node receives out of all messages passing through it) and recall (i.e., the proportion of relevant messages received out of all relevant content produced). A comparative analysis with various other DTN protocols [48] has shown that Habit achieves high delivery (high recall), while keeping overhead low (high precision). However, as demonstrated in Section 2.1, Habit fails to distribute load fairly. We have thus implemented our load-balancing model on top of Habit. To reconcile Habit's goal of minimising the number of uninterested relayers with fairness, the load prevention component is simply modified so that a path is probabilistically selected based on its expected utility, which is now a combination of load and cost:

$$Util_{path} = 1 - \left(\frac{(1-\alpha) \cdot Cost_{path} + \alpha \cdot Load_{path}}{2} \right),$$

where *cost* is quantified in Habit as the number of uninterested intermediaries in a path, normalized over the most costly path found, so to vary in $[0, 1]$; $\alpha \in [0, 1]$ is a weight that can be tuned to give more priority to load or cost. We call the new protocol CoHabit. In the next section, we demonstrate that, while Habit achieves high delivery ratio by means of path choices that are often unfair (i.e., that result in the same nodes always being selected as intermediaries between a given source and destination), CoHabit maintains high delivery ratio while also balancing load evenly.

4. Evaluation

Evaluation has been conducted by means of simulation: we first describe our simulation settings (Section 4.1), before commenting on the obtained results (Section 4.2).

4.1. Simulation Settings

Datasets. In order to perform a realistic evaluation of our work, we required a dataset containing three pieces of information: human mobility traces (to simulate encounters), users' social network (to determine who is interested in receiving content from whom), and users' publication rate (to simulate how often new content is generated). Unfortunately, to date there is no available dataset offering all this information at once; we thus selected three distinct datasets, each providing one piece of information, before overlaying them together as we will describe later. The selected datasets are the following:

- *Mobility Traces:* we experimented with two mobility datasets of different topological properties: the Reality Mining traces, containing mobility information of staff and students at the MIT campus [26], and a vehicular dataset of taxicabs in San Francisco [27]. The former contains colocation information from 96 subjects at the MIT campus over the course of the 2004-2005 academic year, to whom Bluetooth-enabled Nokia 6600 phones were given; colocation information was collected via frequent (5 minute) Bluetooth device discoveries. In our experiments, we extracted five months of colocation data, from September to February; we used the first five weeks of these traces as training period for Habit to discover nodes' regularity of movement; the remaining period was then the actual test period for CoHabit, with nodes creating and sharing content. The MIT dataset has been widely used to evaluate DTN protocols [49]; while being representative of some DTN settings (e.g., a university campus), its sparsity is not representative of DTN urban settings, where nodes are much more frequently connected (as in a MANET), but with short contact time (as in a DTN) [50]. To cover this scenario, we have thus expanded our evaluation to include an urban vehicular dataset [27].

The traces recorded the GPS coordinates of 500 taxicabs, logged every 10 seconds, over a period of 21 days, in the San Francisco Bay Area. In order to infer colocation information from GPS coordinates, we have assumed that two taxicabs are colocated if their physical distance is less than 50 meters (i.e., within WiFi range); furthermore, as taxicab clocks are not synchronized (i.e., two taxicabs may be physically colocated, but may log their respective location with a few seconds difference), we have assumed a 60 second interval during which, if the distance between two taxicabs is less then 50 meters, those taxicabs are assumed to be colocated. In our experiments, we have considered a subset of 100 randomly chosen taxicabs. As the period covered by these traces is quite short, we have replayed the traces twice back-to-back, with the first 21 days used as training period by Habit, and the subsequent 21 days used to actually test CoHabit (i.e., with nodes creating and distributing content). It is worth emphasizing that, in this work, we are not evaluating Habit and its ability to learn regularity (the interested reader may refer to [7]), so that this re-playing of traces does not distort our evaluation of CoHabit.

- *Social Network:* we experimented with two social networks to determine who is interested in receiving content from whom. The first has been implicitly extracted from the MIT Reality Mining dataset; apart from colocation information, this dataset recorded voice calls and text messages exchanged between the participants in the study. Similarly to [51, 52], we have used this information to extract a social network whereby a link from user A to user B is created if A sent a text message or made a phone call to B. We also experimented with a sample of the self-reported social network of Last.fm[3], a music social networking website: first, we have gathered 10,000 Last.fm users with a breadth-first search using the Audioscrobbler Web Services[4]; we have then sampled a (connected) subgraph of 100 users; an analysis of their in-degree distribution highlights the long-tailed degree distribution typical of human social networks.

- *Publication Dataset:* finally, each user in our simulation is asked to create and inject content in the network. In real settings, this does not happen uniformly across days/hours and across users (i.e., some users will create more content than others, and some days of the week will see higher content being produced overall than others). To mimic realistic behaviour as much as

[3] http://www.last.fm

[4] http://www.audioscrobbler.net

possible, we have extracted content publication rates from users of the content bookmarking website Digg[5]: first, we have selected Digg publication data for period of five months starting August 2007; based on this subset we then filtered users such that every user had published at least 50 messages during this period. We then ordered Digg users based on the number of publications and extracted 100 users, making sure the original publication rate distribution is preserved.

To conduct our simulations, we needed to combine these three datasets together. Let us consider mobility and social network first: when using the Reality Mining dataset, users in the mobility traces corresponded to actual users in the social network, so a direct mapping was created. When using the taxicab and the Last.fm datasets, random overlays have been created instead, as no work exists to date describing a realistic way of performing such correlation (the results presented in the next section are averages computed over three random overlays). Let us now consider social network and publication datasets: in this case, we have overlayed the selected Digg users onto the social network (MIT/Last.fm) users, so that the most popular nodes in the social network were also the most active nodes in the publication dataset; this fits the intuition that users with many followers have more content to share than users with few/no followers. Note that we have also experimented with three random mappings between social network and publication dataset; as the results were consistent with those presented in the following, we do not dwell further in it. In the following, we use 'taxicab' to concisely refer the dataset comprising taxicabs movement, Last.fm social network, and Digg publication rate; we use 'Reality Mining' to refer to the dataset comprising MIT movement, MIT social network, and Digg publication.

As described in Section 2.1, we model energy consumption in terms of the number of messages forwarded by intermediaries: if a node reaches its drainage threshold (i.e., maximum number of messages that can be forwarded in a given time window), the node stops forwarding. However, we assume that such nodes continue to create and publish their own messages, as well as receive those they are interested in, despite the fact that they have given up relaying for others. In terms of simulations, this guarantees that the number of published messages remains the same regardless of how many nodes drain their battery, thus enabling better analysis and comparison of results. In reality, this behaviour could be explained as selfishness, where nodes who have seen their battery drain when carrying messages for others now stop cooperating, whilst still wishing their own content to be

[5] http://www.digg.com

distributed.

Metrics and Benchmarks. The goal of CoHabit is to distribute workload fairly amongst nodes, while not compromising delivery ratio. In order to quantify the trade off between these two aspects, we have thus computed *fairness* and *delivery ratio* as our metrics. We define fairness as the coefficient of variation of the total load forwarded by nodes, and formulate it as:

$$fairness = 1 - \frac{\sigma}{\mu},$$

where μ is the average number of messages forwarded by nodes over the whole simulation period, σ presents the standard deviation from the mean, and fairness is a value within the range of $(-\infty, 1]$. As the standard deviation σ converges toward zero (i.e., nodes forward the same amount of traffic), the fairness value becomes closer to 1 (i.e., a uniform workload distribution); at the opposite extreme, when σ is very high and far from the mean μ (i.e., the load is highly skewed, with only a tiny portion of nodes in charge of the whole content distribution), fairness will drift towards $-\infty$. It is worth noting that achieving $fairness = 1$ is impossible in any real human mobility setting; this is because those nodes sitting at the edge of the network will never have the opportunity to contribute to the content forwarding protocol as much as central nodes, even if they wished to. In other words, the fairness value will inevitably depend, and to a potentially large extent, on the topology of the network. In order to put an upper bound on the *achievable fairness* for a given topology, we studied the fairness obtained by a simple epidemic protocol: as this protocol does not favor any particular node as carrier (unlike DTN protocols such as [1, 2, 3, 4]), its fairness value is only impacted by the topology of the mobility traces. In the following, we will report both the row fairness values obtained with CoHabit, and fairness ratios with respect to what epidemic can achieve.

Previous works have investigated the trade-off between message delivery and communication overhead that various DTN protocols entail [7]. In this work, we are not interested in comparing message delivery across different protocols. Rather, we want to investigate the *change in delivery* between a realistic setting (where nodes would defeat cooperation if overloaded) and the idealistic setting (where nodes can be thought of having infinite battery and thus always participate). We then measure *delivery ratio* as the ratio of messages successfully delivered by CoHabit under different drainage thresholds, with respect to Habit in an ideal (albeit unrealistic) consumption-less environment, where nodes have infinite battery power, formulated as:

$$delivery\ ratio = \frac{\sum m_i}{\sum M_i}$$

where $\sum m_i$ is the sum of messages delivered to the interested users by CoHabit, and $\sum M_i$ is the sum of the messages delivered to interested users by original Habit in a consumption-less environment.

Benchmark. We compare CoHabit's performance in terms of both fairness and delivery ratio with FairRoute [47], the only other routing protocol which attempts to distribute load uniformly across the network. In its original formulation, FairRoute, which is based on SimBet [25], does not take time into consideration: it is thus possible, following the participation model described above, that nodes using FairRoute will forward many messages in short periods of time, causing their battery to drain all of a sudden, and consequently withdrawing from future cooperation. In order to have an impartial comparison between CoHabit and FairRoute, we have adapted FairRoute so to include a load monitoring component that forces nodes to temporarily step back from participation once a critical boundary is reached (i.e., similar to the load alleviation component in CoHabit). This adaptation ensures that the very same number of participatory nodes is maintained throughout the simulation, for both CoHabit and FairRoute. Note that FairRoute requires the setting of various parameters which are dependent on the underlying mobility traces; as the protocol was originally evaluated over the Reality Mining traces only, we report its performance over these mobility traces only, thus using the same (optimal) parameter setting reported in the original paper. For completeness, we also compare CoHabit's performance in terms of delivery ratio against CDC and Habit.

Parameters. The results presented next were obtained using the parameters summarised in Table 1; further parameters, required by the underlying Habit protocol, have not been reported, but the reader should assume them to be the very same used in [7]. We have conducted a much broader set of experiments, to quantify, for example, the effect of the critical boundary parameter on the fairness and delivery ratio of CoHabit. Due to space constraints, we only report here a subset of the results obtained; further details can be found in a companion technical report [48].

4.2. Results

In this section, we first perform a sensitivity analysis of CoHabit, in order to analyse the effect of protocol-specific parameters on both delivery ratio and fairness (Section 4.2.1). This first set of experiments was conducted using the taxicab traces only, as they are the ones which cause the highest load, and thus enable us to stress-test the protocol. Having set CoHabit's parameters, we then proceed to evaluate the protocol against the benchmarks and across the traces described before (Section 4.2.2).

	Reality Mining	taxicab
Mapped Social Network	Inferred From Phone Calls	Last.FM
Publication Dataset	Digg	Digg
Number of nodes	100	100
Training Period	35 days	21 days
Simulation Period	150 days	42 days
Time-to-live of the messages	10 days	3 days
Critical Boundary	70%	70%

Table 1: Simulation parameters

4.2.1. Sensitivity Analysis

The α Parameter. As a first experiment, we have measured the impact of CoHabit's load detection component on both delivery ratio and fairness, by varying the α value, when fixing the drainage threshold to 50 messages over 5 days. The results are reported in Table 2. A value of $\alpha = 0$ means that routes are constructed at the source purely based on cost; the fairness achieved thus only depends on the *load alleviation* component. As α increases, the routes are selected by giving more precedence to the load reasoning, thus relying more on the *load prevention* component of CoHabit. When $\alpha = 1$, routes are source-computed based on their estimated load only. As expected, when $\alpha = 0$ fairness is significantly reduced; moreover, delivery ratio is compromised, as messages continue to queue up (and eventually expire) on heavily congested nodes, on which the load alleviation component is continuously called upon. As more weight is given to load reasoning (i.e., increasing α), the load is more evenly distributed, leading to considerably higher fairness. Note that with $\alpha = 1$ delivery ratio is slightly worse than with $\alpha = 0.75$. This is because paths are more conservatively constructed at source, choosing longer routes which result in some messages to expire; this is particularly the case for the very first few messages published, when nodes have not yet learned enough about the actual load distribution in the network, thus making unnecessary cautious choices.

	Delivery Ratio	Fairness (ratio to upper bound)
$\alpha = 1$ (Load only)	81.47%	0.51 (0.83)
$\alpha = 0.75$	81.63%	0.46 (0.75)
$\alpha = 0$ (Cost only)	77.90%	0.33 (0.54)

Table 2: CoHabit fairness for different α (drainage threshold = $\frac{50msgs}{5days}$) on taxicab traces

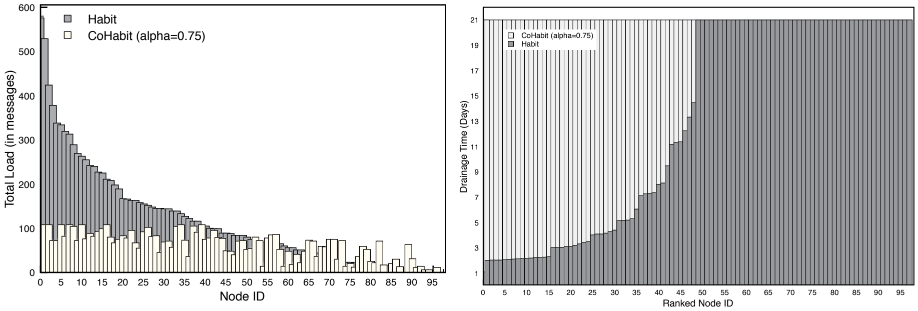
Drainage Threhsold. As a second experiment, we have evaluated the effect of the drainage threshold on fairness. Table 3 presents the results for $\alpha = 1$ (i.e., the setting where maximum weight is given to load, and thus the highest fairness is achieved). A high threshold is representative of applications that share small messages (e.g., Twitter posts); in this case, more messages can be exchanged before battery drains, and nodes can be more relaxed about the workload distribution. When applications share bulkier messages instead (e.g., images and videos), each message forwarded has a high impact on the node's power consumption (small threshold); in such scenario, nodes will not tolerate uneven workload, and fairness becomes a primary objective. As shown, CoHabit offers a fairness as high as 95% of that offered by epidemic in these circumstances.

Drainage Threshold ($\alpha = 1$)	Fairness (ratio to upper bound)
$\frac{25msgs}{5days}$ (e.g., Video files)	0.58 (0.95)
$\frac{50msgs}{5days}$ (e.g., Music files)	0.51 (0.83)
$\frac{100msgs}{5days}$ (e.g., Text files)	0.41 (0.67)

Table 3: CoHabit fairness for different drainage thresholds on taxicab traces

We now set $\alpha = 0.75$ and drainage threshold= $50msgs/5days$ on the taxibab traces, and draw our attention back to the problems highlighted in Section 2.1: when using Habit- and CDC-like DTN protocols, nodes experienced very different load distribution (Figure 1), with some nodes quickly exhausting their battery reserves (Figure 2). We can now illustrate how both problems have been neatly ameliorated using CoHabit. In particular, Figure 3a presents the load distribution experienced when using CoHabit and when using Habit[6]; note that, in the case of Habit, we have given nodes infinite battery (idealistic case), otherwise they would all reach the drainage threshold causing the full content distribution network to collapse. While with Habit the distribution is heavy-tailed (with a few nodes doing all the work, and many nodes contributing almost nothing), in CoHabit the load is rather evenly spread, thus promoting cooperation via reciprocative behaviour [53]; furthermore, by spreading load evenly across many different nodes, CoHabit is less susceptible to node failures and uncooperativeness. The price to pay is delivery, as CoHabit can now deliver only 81% of the messages that Habit brought to destination, although Habit did so in an idealistic setting where battery was infinite. We have then removed the assumption of unlimited battery for Habit, and repeated the experiment with a drainage threshold for both Habit and CoHabit of 50 messages

[6] We omit results for CDC in Figure 3a purely for clarity; however, what we illustrate here for Habit also holds for CDC, as previously shown in Figure 1.



(a) Load Distribution

(b) Drainage Time

Figure 3: CoHabit vs. Habit performance on the taxicab traces

over 5 days (as done in Figure 2): as Figure 3b illustrates, half of the nodes running Habit stop participation at some point in the simulation (and one third of them in the very first couple of days), causing Habit's delivery to collapse, as we shall details in the next Section; on the contrary, all nodes running CoHabit continue contributing to the delivery for the whole duration of the simulation, demonstrating the effectiveness of the Load Prevention and Alleviation components in preventing unfair workload distribution.

4.2.2. Benchmark Evaluation

Having completed a sensitivity analysis of the CoHabit protocol, we now proceed with its evaluation in terms of delivery ratio and fairness, against all benchmarks (Habit, CDC, FairRoute) and traces (Reality Mining and taxicab) described before.

Delivery Ratio. We first measure the delivery ratio achieved by CoHabit, Habit, CDC, and FairRoute when different drainage thresholds were considered. Table 4 and 5 present the results for the taxicab and RealityMining settings respectively. Note that a different TTL was used in the two datasets (i.e., 3 days for taxicab, 10 days for Reality Mining); also, we have been experimenting with different drainage thresholds (i.e., from 25 to 100 messages over 5 days for taxicab, and from 50 to 150 messages for Reality Mining). This is because the two settings exhibit rather different mobility properties: in particular, as Figure 4 illustrates, the Reality Mining mobility traces are very sparse, with very high inter-contact time (in the order of 500 hours) compared to the taxicab traces (in the order of 50 hours). As a result, intermediaries in the Reality Mining setting were locally storing many messages, waiting to be delivered at a later stage upon encounters; when such encounters occurred, nodes' queues contained more messages than they

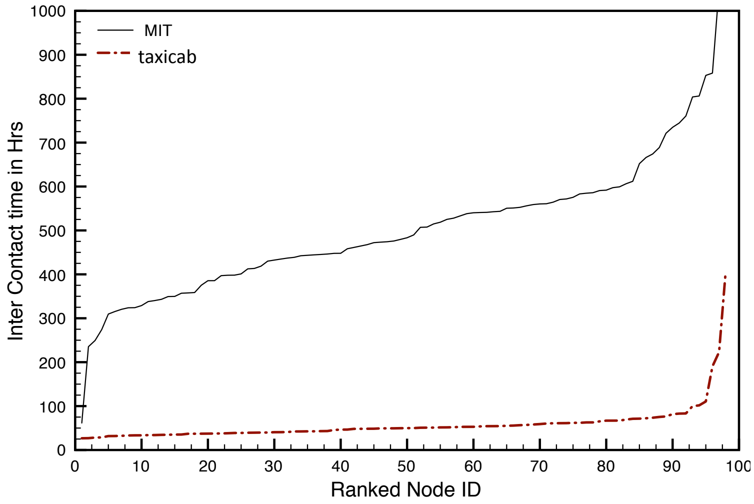


Figure 4: Average inter-contact time per node

could forward within their drainage threshold, causing Habit and CDC to drain resources immediately. For this reason, we experimented with both higher drainage threshold and TTL when using the Reality Mining setup.

Drainage Threshold	$\frac{25msgs}{5days}$	$\frac{50msgs}{5days}$	$\frac{100msgs}{5days}$
CDC	25.23 %	26.99 %	30.87%
Habit	49.39 %	63.25%	81.61%
CoHabit	73.99 %	81.63 %	90.79 %

Table 4: Delivery ratio achieved for taxicab traces

Drainage Threshold	$\frac{50msgs}{5days}$	$\frac{100msgs}{5days}$	$\frac{150msgs}{5days}$
CDC	41.83 %	43.77 %	46.74%
Habit	22.66%	39.18%	46.14 %
FairRoute (with load alleviation)	9.13%	18.80%	27.9%
CoHabit	57.88%	66.57%	72.73%

Table 5: Delivery ratio achieved for the Reality Mining traces

As both tables illustrate, the smaller the threshold, the smaller the delivery ratio achieved with any of the studied protocols; this is because small thresholds imply quicker battery drainage, with nodes consequently ceasing participation in CDC and Habit, while entering the conservative load alleviation phase in CoHabit.

However, there are remarkable differences to be observed: FairRoute (with load alleviation) is the protocol that suffers the highest loss in message delivery, with at most 27% of messages delivered with respect to the idealistic infinite-battery scenario. This low performance is caused by the aggressive forwarding strategy of FairRoute, which causes nodes to quickly reach the critical boundary and step back from participation until sufficient time has passed (e.g., they have recharged their battery). In addition, FairRoute had been designed based on the idea of offloading messages from central nodes to least loaded intermediaries, causing the network delivery to entirely depend on the forwards made by the central nodes, and thus collapsing once they temporary stop forwarding. While less severe than FairRoute, CDC and Habit suffer a 46% reduction in delivery too. CoHabit exhibits the *highest delivery ratio* across all benchmark protocols instead: in the very same situation where CDC and Habit achieve only 46% delivery (100 message in 5 days threshold), CoHabit achieves 72%. The CoHabit load-aware mechanism is thus capable of redirecting traffic toward the less-used parts of the network, offering nodes the ability to recover when becoming overloaded. There is still a portion of messages that even CoHabit is not able to bring to destination within their time to live (TTL): this is because some messages are set to follow longer paths during the load prevention phase; moreover, when the load alleviation phase begins, messages are held in queues for longer periods of time

To illustrate this result further, Figure 5 depicts the cumulative number of messages delivered in the system over time, for a threshold of 50 messages over 5 days, using the Reality Mining traces. As can be seen, Habit's and CDC's delivery slows down significantly very early into the simulation; this is because the most central nodes in the network have quickly reached their drainage threshold and ceased participation. After 60 days, the curve goes completely flat, and no further deliveries are accomplished. In comparison, CoHabit manages to keep up delivering messages throughout, effectively finding fairer paths to destinations.

Fairness. We now focus our attention on fairness, and compute the achieved fairness for both CoHabit and FairRoute, under various drainage thresholds. We omit reporting these values for CDC and Habit, as they were not designed to maintain a fair distribution of load. Table 6 reports these results for the Reality Mining setting. Note that the reported values are computed as a ratio to the highest achievable fairness (i.e., epidemic's fairness).

Drainage Threshold	$\frac{50msgs}{5days}$	$\frac{100msgs}{5days}$	$\frac{150msgs}{5days}$
FairRoute (with load alleviation)	85%	83%	81%
CoHabit	70%	65%	59%

Table 6: Fairness achieved for the Reality Mining traces

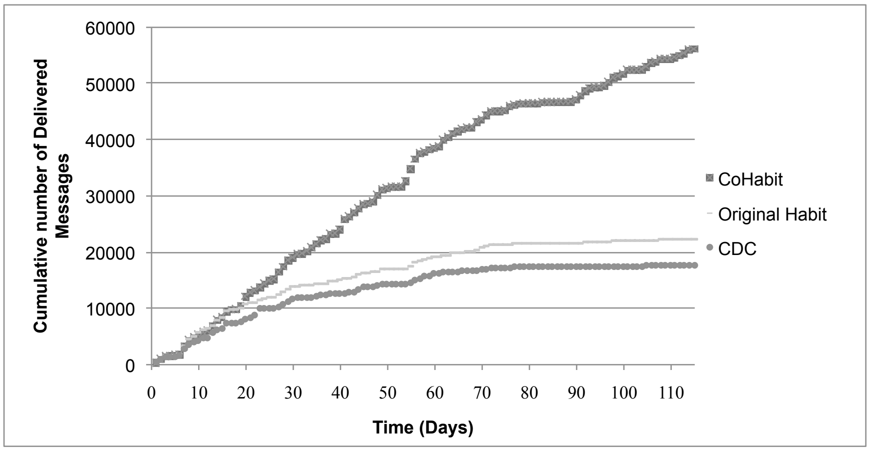


Figure 5: Cumulative number of delivered messages in the Reality Mining traces

Although FairRoute (augmented with load alleviation) achieves a higher fairness than CoHabit, this more uniform distribution of workload comes at the expense of delivery (Table 5). More precisely, when FairRoute achieves a fairness of 83% (drainage threshold of 100 messages in 5 days), its delivery suffers extensively as it only reaches 20% of what it would deliver if there were infinite battery; CoHabit, on the other hand, still delivers 67% of the messages with a fairness of 65%. Note that FairRoute's trade-off between fairness and delivery ratio is even more severe for smaller drainage thresholds, with up to 90% of loss in message delivery so to achieve a high fairness. On the contrary, by combining the results reported in Table 5 (delivery ratio) with those in Table 6 (fairness), we can conclude that CoHabit is capable of distributing load fairly amongst participants without compromising on delivery.

5. Conclusion and Future Work

This paper has presented a novel approach to achieve fair content distribution in participatory DTNs. Each node locally estimates the load of other nodes in the network, and uses this information to source-select the paths that messages should follow, thus spreading them evenly in the network. Moreover, each node locally monitors how much traffic it has been forwarding, so that, should a critical limit be reached, it can back off, and rejoin later when less loaded. We have implemented this approach on top of the Habit source-based DTN protocol, and demonstrated, by means of simulations on real traces, that the approach distributes load fairly, without compromising delivery ratio.

Our plan for the future spans three main directions: first, we will continue investigating CoHabit performance, both in terms of delivery ratio and fairness, in more heterogeneous settings, for example, where nodes have different drainage thresholds, and/or where different proportions of nodes maliciously drop messages. In line with the observations reported in [54], we expect CoHabit to exhibit good resilience to node un-cooperativeness, as a consequence of its strategy to rely on different paths when source computing routes, thus reducing the risks of crossing a malicious relayer. Evaluation against different DTN protocols, while varying cooperation thresholds, will be conducted. Second, we will integrate the load balancing model within other DTN protocols, to quantify the fairness it brings. Indeed, the load alleviation component is of general applicability and can be used as is by any content sharing protocol. Furthermore, the load prevention component can be adapted to non source-based scenarios too, whereby rather than being used to compute whole message paths, it could be used to select what nodes to elect as replica holders (e.g., [1]). Note that, whilst effective on the simulation settings considered in this paper, the load detection technique CoHabit uses is quite simple; understanding whether better detectors can be built, and to what extent, remains an interesting open question. Finally, we intend to integrate our load-aware reasoning with priority-based forwarding, whereby the priority of a message can be determined by, for example, its time-to-live and the strength of the social tie between source and destination. In so doing, nodes will have more control over how to allocate their resources.

References

[1] P. Costa, C. Mascolo, M. Musolesi, Socially-aware Routing for Publish-Subscribe in Delay-tolerant Mobile Ad Hoc Networks, IEEE Journal on Selected Areas in Communications 26 (5) (2008) 748–760.

[2] E. Yoneki, P. Hui, S. Chan, J. Crowcroft, A Socio-Aware Overlay for Publish/Subscribe Communication in Delay Tolerant Networks, in: Proc. of the 10th Symposium on Modelling, Analysis, and Simulation of Wireless and Mobile Systems, ACM, New York, NY, USA, 2007, pp. 225–234.

[3] A. Lindgren, A. Doria, O. Schelén, Probabilistic Routing in Intermittently Connected Networks, Mobile Computing and Communication Review 7 (3) (2004) 19–20.

[4] J. Leguay, A. Lindgren, J. Scott, T. Friedman, J. Crowcroft, Opportunistic Content Distribution in An Urban Setting, in: Proc. of the SIGCOMM Work-

shop on Challenged Networks, ACM, New York, NY, USA, 2006, pp. 205–212.

[5] P. Hui, J. Crowcroft, E. Yoneki, Bubble Rap: Social-based Forwarding in Delay Tolerant Networks, in: Proc. of the 9th International Symposium on Mobile Ad Hoc Networking and Computing, ACM, New York, NY, USA, 2008, pp. 241–250.

[6] Q. Yuan, I. Cardei, J. Wu, Predict and Relay: An Efficient Routing in Disruption-tolerant Networks, in: Proc. of the 10th International Symposium on Mobile Ad Hoc Networking and Computing, ACM, New York, NY, USA, 2009, pp. 95–104.

[7] A. J. Mashhadi, S. B. Mokhtar, L. Capra, Habit: Leveraging Human Mobility and Social Network for Efficient Content Dissemination in Delay Tolerant Networks, in: Proc. of the 10th Annual International Symposium on the World of Wireless, Mobile and Multimedia Networks, IEEE Computer Society, Washington, DC, USA, 2009, pp. 1–6.

[8] J. A. Paradiso, T. Starner, Energy Scavenging for Mobile and Wireless Electronics, IEEE Pervasive Computing 4 (1) (2005) 18–27.

[9] W. Wang, V. Srinivasan, M. Motani, Adaptive Contact Probing Mechanisms for Delay Tolerant Applications, in: Proc. of the 13th Annual International Conference on Mobile Computing and Networking, ACM, New York, NY, USA, 2007, pp. 230–241.

[10] H. Jun, M. Ammar, M. Corner, E. Zegura, Hierarchical Power Management in Disruption Tolerant Networks with Traffic-Aware Optimization, in: Proc. of the SIGCOMM Workshop on Challenged Networks, ACM, New York, NY, USA, 2006, pp. 245–252.

[11] J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, J. Jetcheva, A Performance Comparison of Multi-hop Wireless Ad hoc Network Routing Protocols, in: Proc. of the 4th Annual International Conference on Mobile Computing and Networking, ACM, New York, NY, USA, 1998, pp. 85–97.

[12] A. Vahdat, D. Becker, Epidemic Routing for Partially Connected Ad hoc Networks, Tech. rep., Duke University (2000).

[13] C. Boldrini, M. Conti, J. Jacopini, A. Passarella, HiBOp: a History Based Routing Protocol for Opportunistic Networks, in: Proc. of the International Symposium on a World of Wireless, Mobile and Multimedia Networks, IEEE Computer Society, Washington, DC, USA, 2007, pp. 1–12.

[14] T. Spyropoulos, K. Psounis, C. Raghavendra, Single-Copy Routing in Intermittently Connected Mobile Networks, in: Proc. of the 1st Annual Conference on Sensor and Ad Hoc Communications and Networks, IEEE Computer Society, Washington, DC, USA, 2004, pp. 235–244.

[15] T. Spyropoulos, K. Psounis, C. Raghavendra, Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks, in: Proc. of the SIGCOMM Workshop on Delay Tolerant Networking, ACM, New York, NY, USA, 2005, pp. 252–259.

[16] R. Ramanathan, R. Hansen, P. Basu, R. Rosales-Hain, R. Krishnan, Prioritized Epidemic Routing for Opportunistic Networks, in: Proc. of the 1st International MobiSys Workshop on Mobile Opportunistic Networking, ACM, New York, NY, USA, 2007, pp. 62–66.

[17] C. Bettstetter, G. Resta, P. Santi, The Node Distribution of the Random Waypoint Mobility Model for Wireless Ad hoc Networks, IEEE Transactions on Mobile Computing 2 (3) (2003) 257–269.

[18] M. Balazinska, P. Castro, Characterizing Mobility and Network Usage in a Corporate Wireless Local-Area Network, in: Proc. of the 1st International Conference on Mobile Systems, Applications and Services, ACM, New York, NY, USA, 2003, pp. 303–316.

[19] T. Henderson, D. Kotz, I. Abyzov, The Changing Usage of a Mature Campus-wide Wireless Network, in: Proc. of the 10th Annual International Conference on Mobile Computing and Networking, ACM, New York, NY, USA, 2004, pp. 187–201.

[20] P. Hui, A. Chaintreau, J. Scott, R. Gass, J. Crowcroft, C. Diot, Pocket Switched Networks and Human Mobility in Conference Environments, in: Proc. of the SIGCOMM Workshop on Delay Tolerant Networking, ACM, New York, NY, USA, 2005, pp. 244–251.

[21] T. Spyropoulos, K. Psounis, C. Raghavendra, Performance Analysis of Mobility-Assisted Routing, in: Proc. of the 7th International Symposium on Mobile Ad Hoc Networking and Computing, ACM, New York, NY, USA, 2006, pp. 49–60.

[22] I. Rhee, M. Shin, S. Hong, K. Lee, S. Chong, On the Levy-Walk Nature of Human Mobility, in: Proc. of the 27th International Conference on Computer Communications, IEEE Computer Society, Washington, DC, USA, 2008, pp. 924–932.

[23] C. Song, Z. Qu, N. Blumm, A. Barabasi, Limits of Predictability in Human Mobility, Science 327 (5968) (2010) 1018.

[24] A. Mtibaa, A. Chaintreau, J. LeBrun, E. Oliver, A. Pietilainen, C. Diot, Are You Moved by Your Social Network Application?, in: Proc. of the 1st Workshop on Online Social Networks, ACM, New York, NY, USA, 2008, pp. 67–72.

[25] E. Daly, M. Haahr, Social Network Analysis for Routing in Disconnected Delay-tolerant MANETs, in: Proc. of the 8th International Symposium on Mobile Ad hoc Networking and Computing, ACM, New York, NY, USA, 2007, pp. 32–40.

[26] N. Eagle, A. S. Pentland, Reality Mining: Sensing Complex Social Systems, Personal Ubiquitous Computing. 10 (4) (2006) 255–268.

[27] M. Piorkowski, N. Sarafijanovoc-Djukic, M. Grossglauser, A Parsimonious Model of Mobile Partitioned Networks with Clustering, in: Proc. of the 1st International Conference on Communication Systems and Networks, IEEE Computer Society, Washington, DC, USA, 2009, pp. 1–10.

[28] E. Miluzzo, N. D. Lane, K. Fodor, R. Peterson, H. Lu, M. Musolesi, S. B. Eisenman, X. Zheng, A. T. Campbell, Sensing Meets Mobile Social Networks: the Design, Implementation and Evaluation of the CenceMe Application, in: Proc. of the 6th Conference on Embedded Network Sensor Systems, ACM, New York, NY, USA, 2008, pp. 337–350.

[29] N. Balasubramanian, A. Balasubramanian, A. Venkataramani, Energy Consumption in Mobile Phones: A Measurement Study and Implications for Network Applications, in: Proc. of the 9th International Conference on Internet Measurement Conference, ACM, New York, NY, USA, 2009, pp. 280–293.

[30] J. Kleinberg, Y. Rabani, va Tardos, Fairness in Routing and Load Balancing, Journal of Computer and System Sciences 63 (1) (2001) 2–20.

[31] Z. Cao, Z. Wang, E. Zegura, Performance of Hashing-based Schemes for Internet Load Balancing, in: Proc. of the 19th Annual International Conference on Computer Communication, IEEE Computer Society, Washington, DC, USA, 2002, pp. 332–341.

[32] A. Rao, K. Lakshminarayanan, S. Surana, R. Karp, I. Stoica, Load Balancing in Structured P2P Systems, in: Peer-to-Peer Systems II, Vol. 2735 of Lecture Notes in Computer Science, Springer, 2003, pp. 68–79.

[33] X. Li, L. Cuthbert, A Reliable Node-disjoint Multipath Routing with Low Overhead in Wireless Ad Hoc Networks, in: Proc. of the 7th International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems, ACM, New York, NY, USA, 2004, pp. 230–233.

[34] M. K. Marina, S. R. Das, Ad Hoc On-Demand Multipath Distance Vector Routing, SIGMOBILE Mobile Computing and Communications Review 6 (3) (2002) 92–93.

[35] L. Zhang, Z. Zhao, Y. Shu, L. Wang, O. Yang, Load Balancing of Multipath Source Routing in Ad Hoc Networks, in: Proc. of the International Conference on Communications, IEEE Computer Society, Washington, DC, USA, 2002, pp. 3197–3201.

[36] S. Yin, X. Lin, MALB: MANET Adaptive Load Balancing, in: Proc. of the International Vehicular Technology Conference, IEEE Computer Society, Washington, DC, USA, 2005, pp. 2843–2847.

[37] H. Hassanein, A. Zhou, Load-aware Destination-Controlled Routing for MANETs, Computer Communications 26 (14) (2003) 1551–1559.

[38] J. Solis, N. Asokan, K. Kostiainen, P. Ginzboorg, J. Ott, Controlling Resource Hogs in Mobile Delay-tolerant Networks, Elsevier Computer Communications Journal 33 (1) (2010) 2–10.

[39] C. Boldrini, M. Conti, A. Passarella, Context and Resource Awareness in Opportunistic Network Data Dissemination, in: Proc. of the 9th Annual International Symposium on the World of Wireless, Mobile and Multimedia Networks, IEEE Computer Society, Washington, DC, USA, 2008, pp. 1 –6.

[40] M. Radenkovic, A. Grundy, Congestion Aware Data Dissemination in Social Opportunistic Networks, SIGMOBILE Mobile Computing and Communication Review 14 (3) (2010) 31–33.

[41] M. Seligman, K. Fall, P. Mundur, Storage Routing for DTN Congestion Control, Wireless Communication and Mobile Computing 7 (10) (2007) 1183–1196.

[42] A. Grundy, M. Radenkovic, Decongesting Opportunistic Social-based Forwarding, in: Proc. of the 7th International Conference on Wireless On-demand Network Systems and Services, IEEE Computer Society, Washington, DC, USA, 2010, pp. 82–85.

[43] M. Ye, X. Tang, D. L. Lee, Fair Delay Tolerant Mobile Data Ferrying, in: Proc. of the 10th International Conference on Mobile Data Management: Systems, Services and Middleware, IEEE Computer Society, Washington, DC, USA, 2009, pp. 182–191.

[44] S. Guo, S. Keshav, Fair and Efficient Scheduling in Data Ferrying Networks, in: Proc. of the 3rd International Conference on Emerging Networking Experiments and Technologies, ACM, New York, NY, USA, 2007, pp. 1–12.

[45] A. Balasubramanian, B. Levine, A. Venkataramani, DTN Routing as a Resource Allocation Problem, SIGCOMM Computer Communication Review 37 (4) (2007) 373–384.

[46] G. Sandulescu, S. Nadjm-Tehrani, Opportunistic DTN Routing with Window-Aware Adaptive Replication, in: Proc. of the 4th Asian Conference on Internet Engineering, ACM, New York, NY, USA, 2008, pp. 103–112.

[47] J. Pujol, A. Toledo, P. Rodriguez, Fair routing in delay tolerant networks, in: Proc. of the 28th International Conference on Computer Communications, IEEE Computer Society, Washington, DC, USA, 2009, pp. 837–845.

[48] A. J. Mashhadi, Source Based Content Dissemination in Participatory DTN, Tech. Rep. RN/09/06, University College London (2009).

[49] Z. Zhang, Routing in Intermittently Connected Mobile Ad Hoc Networks and Delay Tolerant Networks: Overview and Challenges, IEEE Communications Surveys & Tutorials 8 (1) (2007) 24–37.

[50] J. Whitbeck, V. Conan, HYMAD: Hybrid DTN-MANET Routing for Dense and Highly Dynamic Wireless Networks, Computer Communications 33 (13) (2010) 1483–1492.

[51] A. Lindgren, C. Diot, J. Scott, Impact of Communication Infrastructure on Forwarding in Pocket Switched Networks, in: Proc. of the SIGCOMM Workshop on Challenged Networks, ACM, New York, NY, USA, 2006, pp. 261–268.

[52] D. Quercia, L. Capra, FriendSensing: Recommending Friends using Mobile Phones, in: Proc. of the 3rd International Conference on Recommender Systems, ACM, New York, NY, USA, 2009, pp. 273–276.

[53] M. A. Nowak, K. Sigmund, Evolution of Indirect Reciprocity by Image Scoring, Nature 393 (6685) (1998) 573–577.

[54] G. Resta, P. Santi, The Effects of Node Cooperation Level on Routing Performance in Delay Tolerant Networks, in: Proc. of the 6th Annual Conference on Sensor, Mesh and Ad Hoc Communications and Networks, IEEE Computer Society, Washington, DC, USA, 2009, pp. 413–421.