Scoop: Decentralized and Opportunistic Multicasting of Information Streams

Dinan Gunawardena
Microsoft Research
Cambridge, UK
dinang@microsoft.com

Thomas Karagiannis
Microsoft Research
Cambridge, UK
thomkar@microsoft.com

Alexandre Proutiere*
KTH
Stockholm, Sweden
alepro@kth.se

Elizeu Santos-Neto†
University of British Columbia
Canada
elizeus@ece.ubc.ca

Milan Vojnović
Microsoft Research
Cambridge, UK
milanv@microsoft.com

ABSTRACT

We consider the problem of delivering information streams to interested mobile users, leveraging both access to the infrastructure and device-to-device data transfers. The goal is to design practical relaying algorithms that aim at optimizing a global system objective that accounts for two important aspects: first, the user interest in content with respect to its type and delivery time; and, second, resource constraints such as storage and transmission costs.

We first examine a set of real-world datasets reporting contacts between users moving in relatively restricted geographic areas (e.g. a city). These datasets provide evidence that significant performance gains can be achieved by extending the information dissemination from one to two hops, and that using longer paths only brings marginal benefits. We also show that correlation of delays through different paths is typically significant, thus asking for system design that would allow for general user mobility.

We then propose a class of relaying strategies (referred to as SCOOP) that aim at optimizing a global system objective, are fully decentralized, require only locally observable states by individual devices, and allow for general user mobility. These properties characterize a practical scheme whose efficiency is evaluated using real-world mobility traces.

Categories and Subject Descriptors: C.2.1 [Network Architecture and Design]: Wireless Communication
General Terms: Algorithms, Design, Performance
Keywords: DTNs, decentralized opportunistic relaying, two-hop relaying

*Work performed while a researcher with Microsoft Research Cambridge.

†Work performed in part while an intern with Microsoft Research Cambridge.

MobiCom'11, September 19–23, 2011, Las Vegas, Nevada, USA.

1. INTRODUCTION

The concept of opportunistic communications has emerged as an alternative and augmentation of traditional networks for devices that experience intermittent connectivity. In such networks, besides the regular access to wireless or wired networks, mobile devices may exploit opportunistic device-to-device data transfers to increase network performance and achieve dissemination of information.

While initially targeted for disaster recovery, vehicular or challenged networks that are delay-tolerant (DTNs), opportunistic communications have recently attracted additional interest as a means to reduce the communication cost both for Internet Service Providers (ISPs) and individual users, especially for the case of 3G networks. Applications that constantly push information streams and content to mobile devices (e.g., news broadcasting, Facebook and Twitter feeds, podcasting [16, 13, 11]) are commonplace and their data volumes are projected to significantly increase [21], posing a challenge to the existing infrastructure. Operators, thus, consider opportunistic data transfers to alleviate congestion in their backhaul networks, e.g., in 3G networks [1] and, similarly, in the British Telecom FON network [4]. Additionally, from the user perspective, device-to-device transfers may considerably reduce the cost of mobile data plans and prevent extra charges imposed when exceeding monthly data limits or when roaming. Combining the direct access data dissemination with device-to-device data transfers may well enable cost and time effective dissemination of media-rich content, which otherwise may be too expensive for dissemination through direct access.

Typically, proposals for routing or, more generally, information dissemination in DTNs, either attempt to keep a single copy of a message to deliver in the network (i.e., *forwarding protocols*, e.g., [12]), or replicate messages at transfer opportunities to find a path to the destination (i.e., *epidemic routing*). As message replication implies resource costs, existing solutions attempt to limit the message replication in the network by deploying various heuristics, such as limiting the number of existing replicas (e.g., [26, 5]), inferring the likelihood of delivering the message (e.g., [19, 3]), or leveraging the social structure of the network [8].

Most of the proposed protocols that limit message replication try to infer device mobility and track the expected delays towards various nodes. Information about node mobility and message delay helps the protocols to make informed decisions on which messages to relay. In order to estimate delays, proposed protocols often make simplifying assumptions about user mobility such as that delays through different paths are independent. However, as our analysis of real-world mobility traces shows, such independence assumptions may not hold in practice. Given the status quo, an outstanding problem is to devise a practical message relaying algorithm that aims at maximizing an a priori defined global system objective for general user mobility.

This work proposes a class of decentralized and opportunistic relaying strategies (referred to as SCOOP) that aim at optimizing an a priori defined global system objective. The admitted global system objective captures two important aspects: i) the value of information streams to users by accounting in a natural way for both users' preference across various information streams with respect to the content as well as the timeliness of delivery; and, ii) the storage and transmission costs due to the message relaying strategy. In summary, SCOOP features the following desired properties: (1) it aims at optimizing a well-defined global system objective, (2) it supports *multi-point to multi-point* communication, i.e., a multicast delivery of information streams, unlike to previous proposals of point-to-point (unicast) routing schemes, (3) it is *decentralized* and requires only local observations to make message relaying decisions, and (4) it allows for general user mobility, and, in particular, it *does not require any independence assumptions* with regard to message forwarding paths and is thus practical.

Additionally, SCOOP is simple in that the decision to relay a message from a given information stream, when the relay meets the corresponding source, depends on a single *per-channel control variable* that is identical for *all* messages of a stream (the probability to relay messages of that stream). This per-channel control variable contrasts with some other proposals (e.g., RAPID [3]), where this decision is taken on a per-message basis, depending on whether or not the message has been already relayed by other nodes and on delay estimates. The approach used by SCOOP allows for scalable implementation and yet yields efficient message delivery without requiring delay estimations. Furthermore, our proposed relaying algorithm is based on a well-defined sub-gradient procedure for finding extremum points of the global objective function, which we show can be done using only locally observable information by individual nodes. The use of only local information is unlike to most previous proposals that require global knowledge by individual nodes such as assuming that for a given message, at relaying decisions instances, a node knows the set of relays that received the message, and for this reason must rely on approximations.

The key assumption that underlies the design of the relaying strategies proposed in this paper is the restriction to forward messages along paths of length one or two hops (i.e., messages are transferred to a user either through a direct contact with a source or through another user acting as a *relay*), which is referred to as *two-hop relaying*. The restriction to two-hop path relaying is crucial for tractability reasons as it allows us to formulate an optimization problem for the underlying information delivery task, which seems to be rather difficult if not impossible without limiting the path lenghts. While this restriction may degrade the efficiency of the information dissemination compared with a relaying strategy that would allow for longer-length paths, using real-world mobility traces, we provide indications that in many cases in practice, forwarding along paths of length at most two may already provide nearly optimal performance. Our data analysis further indicates that relaying paths in mobile networks are typically positively correlated, and thus the path delay independence assumption may not be valid in practice; interestingly, positive correlations persist across a wide range of communication delays.

In summary, our contributions include the following:

• The analysis of several real-world traces indicates that two-hops may be enough for opportunistic relaying of information and that relaying paths are positively correlated in practical scenarios (Section 2). This characterization result provides a justification to restrict the design to two-hop relaying. The observed positive correlations suggest that deriving information dissemination schemes using an underlying user mobility model under which delays through distinct paths are statistically independent is unrealistic. To fill this gap, SCOOP admits a general user mobility described by a stationary ergodic process. Thus, it allows for user mobility to be statistically non-identical across users and, in general, statistically dependent across individual users and time.

• We formulate a natural global system objective and devise a decentralized relaying strategy that aims at optimizing this global system objective (Section 3). This is unlike to proposals that deploy various heuristics to relay messages that are not necessarily optimal with respect to an underlying global system objective. These strategies lack to achieve optimal performance because they are either not designed for the given objective or rely on some simplifying assumptions on user mobility that may not hold in practice. Our decentralized relaying strategy is derived as a sub-gradient scheme for an a priori defined global system objective combining the techniques from the Smoothed Perturbation Analysis (SPA) (e.g. [7]) and the theory of stochastic approximation (e.g. [15]).

• We describe a baseline implementation of SCOOP and demonstrate the performance and practicality of the proposed framework through simulations using real-world mobility traces (Section 4). Specifically, the results show that, overall, SCOOP achieves good delivery rates that are in many cases near to the delivery rates of an omniscious RAPID-like scheme.

To the best of our knowledge, this work is the first to present a class of relaying strategies for multicast content delivery in mobile ad-hoc networks that optimize a global system objective and allow for general user mobility. In particular, compared to previous work, our framework alleviates making any specific assumptions on user mobility that may not hold in practice, such as statistical independence of inter-contact times between mobile devices, homogeneity of distributions of inter-contact times between distinct pairs of mobile devices, and assuming specific parametric families for the distributions of inter-contact times.

2. MULTI-HOP RELAYING

This section examines the benefits of multi-hop relaying strategies through a set of real-world datasets. These datasets, summarized in Table 1, report contacts between users moving within a relatively restricted geographic area (a conference site, a campus, or a city). Limiting the coverage of DTNs allows information to be disseminated within reasonable delays. We note that the results presented here are unlikely to hold when considering wider DTNs (i.e., several cities), but such networks would naturally exhibit much higher delays.

Overall, the analyzed traces have widely different properties in terms of their duration, the environment, and the type of contacts studied, such as contacts of human mobility, e.g., bluetooth contacts of human carried devices (*Infocom* trace) [25], device presence in WiFi hotspots (*UCSD* trace) [20], contacts from moving vehicles, e.g., opportunistic data transfers across the *DieselNet buses* [2], and GPS inferred contacts (*SF Taxis* trace) [23].

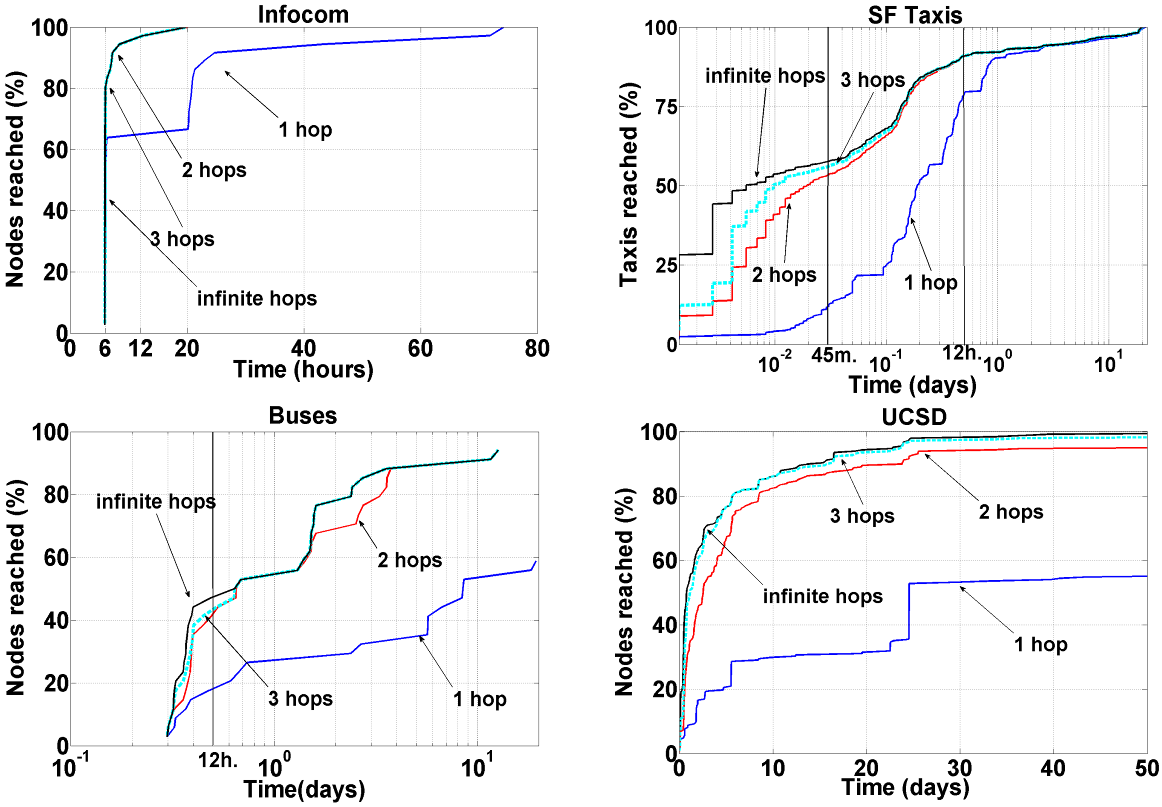

Figure 1: Dissemination delay vs. the maximum number of hops allowed.

Table 1: Traces studied.

Name	Technology	Duration	Devices	Contacts	Year
UCSD	WiFi	77 days	275	116,383	2002
Infocom	Bluetooth	3 days	37	42,569	2005
DieselNet	WiFi	20 days	34	3,268	2007
SF Taxis	GPS	24 days	535	183M	2008

In particular, this section addresses the following questions:

Q1: Do a few number of hops suffice for content relaying or are long paths required to achieve acceptable performance?

Q2: What are the properties of the discovered multi-hop paths, and in particular, are the paths independent?

In the remainder of this section we provide support for the following two claims: (1) *two-hop relaying brings most of the benefits when considering multi-hop relaying* and (2) *dependence across two-hops paths is significant, and thus independence assumptions do not appear to be valid.*

2.1 Benefits of Two-Hop Relaying

The characterization that follows examines the benefits of exploiting multiple hops in opportunistic information dissemination. To this end, we track a message dissemination through contacts for the four traces in Table 1. In particular, this investigation focuses on the message dissemination time defined as the time it takes for a message originated from a node to reach all nodes connected to the source through a path of length limited to some fixed number of hops. To measure this dissemination time, we randomly choose a source node, and observe how information originated from this node spreads through the network allowing for k-hop paths only, with k varying from 1 (i.e., direct contacts only) to "infinite" (i.e., the total number of devices in the trace).

Fig. 1 shows the results of this analysis for all the traces studied. In all cases, it is evident that, for all practical purposes, using just two hops yields nearly the same performance as using "any-hop" paths to disseminate information. The information relayed using direct contacts (i.e., one-hop relaying) only reaches a fraction of the population for the *DieselNet* and *UCSD* traces, equal to roughly 60%; in general, exploiting direct contacts only results in significant delays compared to multi-hop forwarding. Going beyond two hops brings marginal delay benefits, an observation which holds irrespective of the type and properties of the trace, and irrespective of the source node chosen. Specifically, we find that the improvement of two-hop paths compared to one-hop paths is typically at least one order of magnitude with regard to the dissemination time. These benefits significantly diminish going beyond two-hop paths (see Table 2).

From the system design perspective, *it is important that two-hop relaying schemes can achieve delays close to the "optimal"*, as restricting to two-hop relaying schemes significantly simplifies the design of relaying strategies.

2.2 Paths are Positively Correlated

Typically, opportunistic relaying algorithms operate by replicating messages at device contacts based on pre-defined heuristic rules,

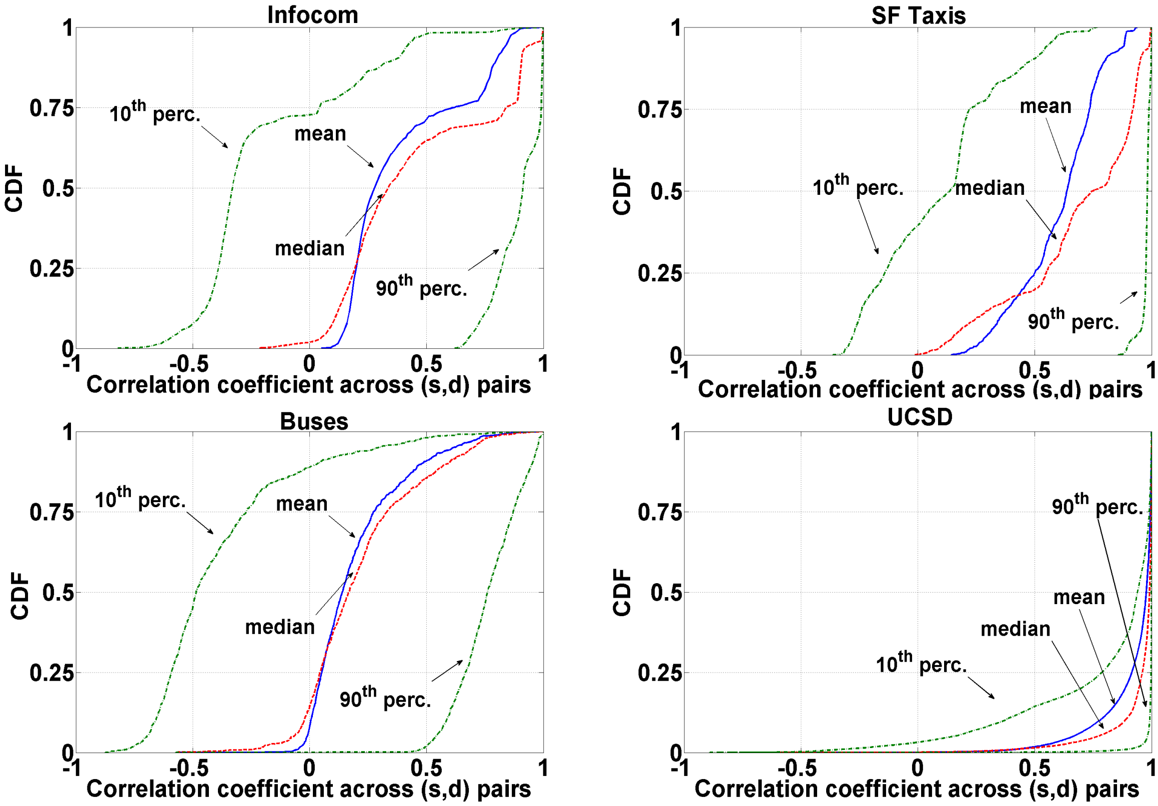

Figure 2: Correlation coefficients for two-hop paths per (s,d) pair.

Table 2: Median delivery delay vs. number of hops.

	1	2	3	∞
UCSD	25 days	2.5 days	1 day	1 day
Infocom	6 hr	6 hr	6 hr	6 hr
DieselNet	8 days	40 min	40 min	40 min
SF Taxis	4 hr	15 min	7 min	3 min

and/or attempt to optimize a utility function using some simplifying assumptions about user mobility or global information state [3, 14, 26, 5]. In most cases, content relaying ignores any possible relationships among the various relay nodes and similarly analytical tractability favors the assumption of statistically independent relaying paths. However, in practice, one would expect that correlations among relaying nodes do exist, and that such correlations might result in sub-optimal forwarding and duplication of the content of interest. For example, devices carried by friends or co-workers might exhibit similar daily patterns with regard to their contacts with other devices. Thus, content duplication in such cases, where delay patterns between two devices are highly correlated, would bear little or no benefit in practice.

Having established that two-hop paths are sufficient, we now concentrate on analyzing the independence hypothesis by studying possible correlations across two-hop relaying paths. To this end, for a source device s, and a destination device d, we examine the time it takes (i.e., delay) for a message originated at device s to reach device d through a relay device r, for all possible (s,r,d) paths. We estimate the path delay by sampling at regular intervals throughout the trace, thus creating a delay time series per (s,r,d) path. For example, one could sample once per day at 10 am, where the delay would specify the time passed since d last received content from s through r, assuming that s always has new content to offer. Two paths for the same (s,d) appear independent, if the correlation of the respective time series is close to zero.

Fig. 2 aggregates the (Pearson's) correlation coefficients across (s,d) pairs (i.e., correlating the delay time series across all possible relays) for all traces, by providing the mean, the median, the 10- and the 90-percentile of the Cumulative Distribution Function (CDF). For the specific figure, we sample the delay every two hours for the *Infocom* trace and once per day at noon for the other traces. Note, that for each (s,d) pair, we need to compute $n(n-1)/2$ correlation coefficients, where n is the number of nodes in the given data trace. For the *SF-Taxis* trace, in order to restrict the number of (s,r,d) paths to consider for reasons of computational complexity, we performed the analysis for a sample of 75 taxis as a source or a destination (i.e., 5550 (s,d) pairs) and all possible 535 relays.

Most (s,d) pairs exhibit significant correlations. Fig. 2 highlights that in the vast majority of cases, most paths per (s,d) pair exhibit correlations (i.e., most points in the CDF are far away from 0). Table 3 displays the median correlation coefficients. Overall, positive correlations are prominent, while uncorrelated pairs seem limited. This implies that carefully selecting relays is crucial to optimize content distribution, as disseminating content through positively correlated paths might lead to sub-optimal performance.

The observed correlation is present irrespective of the dissemination delay. Fig. 3 examines the median value of the correlation coefficient across relay paths conditioned on their respective delay values. Specifically, for two relay paths (s,r_1,d) and (s,r_2,d), the

Table 3: **Correlation of two-hop paths.**

	Median
UCSD	0.98
Infocom	0.3
DieselNet	0.2
SF Taxis	0.75

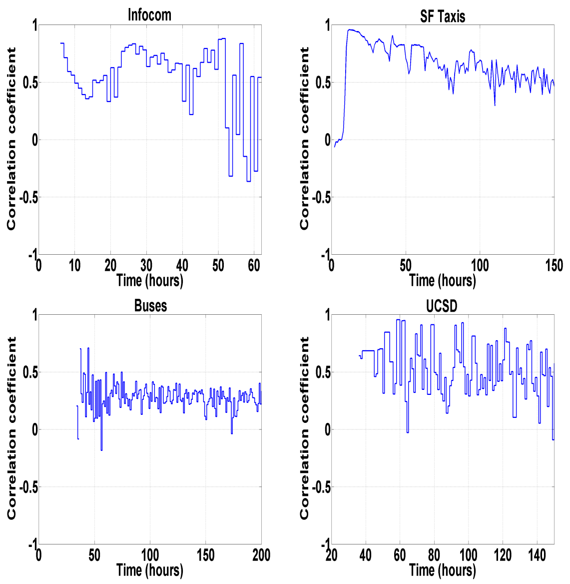

Figure 3: Correlation coefficient vs. the delay of relay paths. Positive correlations exist over the whole range of delays.

figure examines their correlation coefficient value versus the maximum of the respective mean path delays, and aggregates the correlation values by plotting the median per delay value. Maximum over the delay of the two paths was chosen to ensure that the delay of the two paths examined is bounded by the value on the *x-axis*. As previously observed, positive correlations are prominent irrespective of the mean path delays in all traces. Further, no clear trend is observed, with the median correlation coefficient value remaining roughly invariant to the mean path delay values.

Finally, we remark that not all paths per (s,d) pair exhibit correlations. We have examined the fraction of paths per (s,d) pair for which the correlation coefficient is within some interval $(-\delta, \delta)$ for small values of $\delta > 0$. Depending on the value of δ (e.g., from 0.01 to 0.09), the median fraction of uncorrelated paths varies between 1% to 5%. This amounts to roughly 1,500 to 7,000 paths per (s,d) for the *SF-Taxis* trace, or 10 to 300 paths for the *Infocom* trace. This indicates existence of paths with statistically independent delays that can be leveraged for the content dissemination task.

3. RELAYING ALGORITHMS

This section introduces a natural global objective for relaying strategies and describes a distributed relaying scheme that aims at optimizing this objective. The global objective captures both user preferences over information channels and their timeliness of delivery. The proposed scheme opportunistically and optimally exploits mobility so as to deliver content to users without relying on any specific assumptions about their mobility.

3.1 Objectives

3.1.1 Channels

We consider a system that consists of a set of information channels I, assumed to be finite, and a set of users $\mathcal{U}$. Each user is interested in the content of some of these channels. Channel i publishes messages at instances of a stationary and ergodic process at rate $0 < \lambda_i < \infty$ of messages per second. Users may receive messages directly from a source of the corresponding channel or through another user acting as a relay. For each channel i, we assume that a message published at time t may be of interest to a user if it reaches this user no later than $t + t_i$, where t_i is a *deadline* associated with messages of channel i.

3.1.2 User Mobility

We assume that user mobility is a stationary and ergodic process, thus allowing for general user mobility. For example, individual user movements are allowed to be statistically non identical and correlations are allowed across time and across individual user movements. User mobility is naturally assumed to be independent of message generation processes at the sources. As a consequence, we may define the stationary one-hop and two-hop delays: $D_{i,u}$ is the time it takes for a channel-i message to reach user u without the help of any relay, and $D_{i,r,u}$ is the time it takes for a channel-i message to reach user u through relay r. Note that it might well be that the two-hop delays of channel-i messages to user u through different relays are correlated.

3.1.3 Probabilistic Relaying Strategies

We consider randomized relaying strategies that are specified by $x \in [0,1]^{|I| \times |\mathcal{U}|}$ where $x_{i,r}$ represents the probability that user r relays a message of channel i. User r provides, for relaying purposes, a finite storage of size B_r messages. Denote by s and t two consecutive contact times between a relay r and a source of channel i. At time t, relay r considers all messages published by channel i in the time interval $(\max(s, t-t_i), t]$ in decreasing order of message age, and downloads each such message with probability $x_{i,r}$, where draws for different messages are independent. Note that no transmission constraints are considered here, as relays are assumed to be able, while in contact with a source, to download all messages generated in the interval $(\max(s, t-t_i), t]$. We may however extend our algorithm and analysis to model short mobile-to-mobile or mobile-to-source contacts and include transmission constraints[1].

The relays deploy a First-In-First-Out *buffer policy*: to allow the download of a new message when the buffer is full, the least recently downloaded message is discarded. Note that the buffer policy could also be age based. In the following we denote by $\mathbb{P}_x[\cdot]$ and $\mathbb{E}_x[\cdot]$ the steady-state probability distribution and expectation of random variables under relaying strategy x.

Using probabilistic relaying strategies incurs some network *overhead* as some messages, downloaded by relays, would not actually be delivered to any user. It proves rather difficult to quantifying this overhead using mathematical analysis. Note, however, that our strategies are by design trying to minimize this overhead.

3.1.4 Performance Metrics and Objectives

We consider relaying strategies that aim at optimizing a natural global system objective, which we define in the following. Two factors determine the value of an information channel to a user: (1) user-specific preference for the content of the given channel and (2)

[1] We provide details in the companion technical report [10].

timeliness of delivery. Let $p_{i,u}(x)$ be the steady-state probability that a message of channel i is received within deadline by user u under relaying strategy x. That is

$$p_{i,u}(x) = \mathbb{P}_x[A_{i,u} \le t_i]$$

where $A_{i,u}$ is the age of a message of channel i when received by user u (assumed to be infinite if the message is never received by the user). Notice that in a stationary and ergodic regime, $p_{i,u}(x)$ corresponds to the delivery rate of channel-i messages to user u (counting only messages received within deadline t_i) over an asymptotically large number of published messages.

We define the value of channel i to user u by $V_{u,i}(p_{i,u}(x))$, where $V_{i,u}$ is an increasing function $V_{u,i} : [0,1] \to \mathbb{R}$. This definition naturally captures both intrinsic user interest for the content of given channel and its timeliness of delivery. Special cases include linear functions such that $V_{i,u}(p_{i,u}(x)) = w_{i,u} p_{i,u}(x)$ where $w_{i,u}$ is a positive constant that captures user u's intrinsic preference for channel i. For example, $w_{i,u}$ may take binary values, value 1 if user u subscribes to channel i, and value 0 otherwise.

The global system objective is to optimize aggregate value of information channels across users in the system:

$$\begin{array}{rl} & \textbf{SYSTEM} \\ \text{maximize} & \sum_{i \in I, u \in \mathcal{U}} V_{i,u}(p_{i,u}(x)) \\ \text{over} & x \in [0,1]^{|I| \times |\mathcal{U}|}. \end{array} \tag{1}$$

With linear utility functions, the objective is simply to maximize the number of messages delivered to interested users, which may be quite unfair as users with large two-hop delays may be penalized. To ensure some level of fairness, one could choose logarithmic utility functions yielding a *Proportionally Fair* allocation of relaying resources. In practice, the specific choice of user utility functions and channel deadlines would ultimately be made on the basis of the service usability as perceived by users, which would require a user study and is out of scope of the present paper. In our work, we allow for general utility functions which can be instantiated to particular choices in practice.

The above optimization problem accounts for buffer constraints at individual user devices, which are implicit in the definition of the delivery probability $p_{i,u}(x)$. We will provide an explicit characterization of the delivery probability, $p_{i,u}(x)$, later in this section. Notice that in the above optimization problem, there is no cost for relays to download messages from sources or to transmit these messages to interested users. In a more realistic setting, relays may wish to limit the number of transmissions, for example to save battery power. Assume that the cost for relay r to transmit and receive messages to be relayed at average rate a_r is captured through a cost function $C_r(a_r)$, assumed to be increasing, continuously differentiable, and convex. This is accommodated by replacing the objective function in (1) by:

$$\sum_{i \in I, u \in \mathcal{U}} V_{i,u}(p_{i,u}(x)) - \sum_{r \in \mathcal{U}} C_r(a_r(x)) \tag{2}$$

where $a_r(x)$ represents the average transmission and reception rate of relay r under strategy x. The analysis and the relaying strategies proposed here to solve SYSTEM can be extended to include transmission costs[1].

3.2 Sub-gradient Algorithms

This section focuses on describing relaying strategies that aim at solving SYSTEM, introduced in (1). Our strategies are based on sub-gradient method that amounts to updating the relay probabilities as follows, for every channel i and relay r,

$$\frac{d}{dt} x_{i,r} = \sum_{j \in I, u \in \mathcal{U}} V'_{j,u}(p_{j,u}(x)) \frac{\partial}{\partial x_{i,r}} p_{j,u}(x). \tag{3}$$

Under this dynamics, the objective function in SYSTEM increases over time and converges to a maximum value. Due to space limitations, this paper skips the presentation of structural properties of the optimization problem (1), but note that we were able to establish uniqueness of optima under some simplifying assumptions[1].

The difficulty of this approach lies in evaluating the gradient in (3), i.e. for every channel j and user u, we need to evaluate $\partial p_{j,u}(x)/\partial x_{i,r}$, for every channel i and relay r. To address this challenge, we combine techniques from Smoothed Perturbation Analysis (SPA) (e.g., [7]), and stochastic approximation (e.g., [15]). In what follows, in order to simplify the presentation, we use linear utility functions so that $V_{i,u}(p_{i,u}(x)) = w_{i,u} p_{i,u}(x)$, for some positive constant $w_{i,u}$. Note that the analysis readily extends to more general classes of utility functions (indeed for such functions, one just need to further evaluate $p_{j,u}(x)$ which is arguably easier than to compute its derivates).

3.2.1 *Smoothed Perturbation Analysis*

We show how to evaluate the gradient of the function $p_{j,u}(x)$, for every channel j and relay r, using smoothed perturbation techniques [7]. This yields an explicit characterization of the gradient in terms of expectations of some random variables whose realizations can be locally observed by users and estimated by an online procedure that we describe in §3.3.

The age $A_{j,u}$ of a message of channel j when received by user u, if received at all, exceeds the deadline t_j for user u, if the message could not have been received within deadline by user u through neither a direct contact with a source of the message nor via any relay. We characterize this event in the following.

We first need to introduce some notation for a message of channel j. Without loss of generality, we assume that this message was generated at time equal to 0. Let $\tilde{A}_{j,r,u}$ denote the age of the message of channel j at earliest time instant at which it could have been received by user u through relay r (if it was downloaded by relay r). Let $N_{j,r,u}$ denote the number of messages admitted by relay r in the time interval $(D_{j,r}, D_{j,r,u}]$, where $D_{j,r}$ is the one-hop delay from j to relay r and $D_{j,r,u}$ is the two-hop delay from j to u through relay r. Notice that each message admitted in the latter time interval moves the message of channel j towards the head of the queue. The age $\tilde{A}_{j,r,u}$ is less than t_j if and only if both of the following two conditions hold true: (1) there exists a path to user u through relay r within deadline t_j, i.e. $D_{j,r,u} \le t_j$ and (2) the message is not evicted by the buffer policy at relay r, i.e. $N_{j,r,u} < B_r$. Therefore, we have

$$\{\tilde{A}_{j,r,u} \le t_j\} = \{D_{j,r,u} \le t_j\} \cap \{N_{j,r,u} < B_r\}.$$

Note that $\tilde{A}_{j,r,u}$ is defined for each message of channel j and may have a finite value even if the message was not downloaded by relay r. To account for this, we define

$$A_{j,r,u} = \begin{cases} \tilde{A}_{j,r,u} & \text{if } R_{j,r} = 1 \\ \infty & \text{otherwise} \end{cases}$$

where $R_{j,r}$ is a binary indicator that takes value 1 if the message was admitted by relay r and value 0, otherwise.

Now, observe that $A_{j,u} > t_j$ holds if and only if (1) the message could not have been delivered through a direct contact of user u with a source, i.e. $D_{j,u} > t_j$, and (2) there exists no path to deliver the message through a relay within the deadline, i.e. $A_{j,r,u} > t_j$, for every relay r. In other words, we have

$$\{A_{j,u} > t_j\} = \{D_{j,u} > t_j\} \cap_r \{A_{j,r,u} > t_j\}.$$

In order to present the main result of this section, we need to introduce some new notation. Let $N^i_{j,r,u}$ be the number of channel-i messages downloaded by relay r in the time interval $(D_{j,r}, D_{j,r,u}]$ and let $K^i_{j,r,u}$ be the number of channel-i messages that are *observed* by relay r in $(D_{j,r}, D_{j,r,u}]$ but not downloaded. Denote by $A^{-r}_{j,u}$ the age of a message of channel j when arriving at user u, assuming that relay r is not used to disseminate the message. Notice that $A^{-r}_{j,u} > t_j$ holds if and only if $D_{j,u} > t_j$ and $A_{j,r',u} > t_j$, for every relay $r' \neq r$. Finally, let us define the following indicator, for a message of channel j, relay r, and user u,[2]

$$I_{j,r,u} = \mathbb{I}_{A^{-r}_{j,u} > t_j} \mathbb{I}_{D_{j,r,u} \leq t_j}.$$

We can now state the main result of this section that characterizes the gradient of the function $p_{j,u}(x)$, for every channel j and relay r. This is a key result that will enable us to devise optimal relaying strategies. The proof of the theorem is presented in § 3.4.

THEOREM 3.1. *For every channel $j \in I$ and user $u \in \mathcal{U}$, the gradient of the function $p_{j,u}(x)$ is given by, for every channel i and relay r,*

$$\begin{aligned}\frac{\partial}{\partial x_{i,r}} p_{j,u}(x) &= \mathbb{E}_x[\mathbb{I}_{A^{-r}_{i,u} > t_i} \mathbb{I}_{\tilde{A}_{i,r,u} \leq t_i}] \mathbb{I}_{j=i} \\ &- \mathbb{E}_x \left[I_{j,r,u} R_{j,r} (N^i_{j,r,u} \mathbb{I}_{N_{j,r,u}=B_r} + K^i_{j,r,u} \mathbb{I}_{N_{j,r,u}=B_r-1}) \right].\end{aligned} \quad (4)$$

The component of the gradient, $\partial p_{j,u}(x)/\partial x_{i,r}$, consists of a positive and a negative element that admit the following intuitive interpretations. First, the positive element is zero for every $i \neq j$; for $i = j$, it corresponds to the probability that the message of channel i could have been delivered through relay r and not through any other path. Second, the negative element can be interpreted as a *negative externality* term that captures the effect of increasing the relaying probability $x_{i,r}$ on the probability of delivery of channel-j messages. This term measures the number of channel-i messages downloaded by relay r during the time the channel-j message, which was dropped by r just before meeting user u, was in the buffer of relay r; and the number of channel-i messages that were rejected by relay r during the time the channel-j message was in the buffer and it was at the head of the queue (next to be evicted) when relay r meets user u.

The gradient in (4) can be estimated in an online fashion by relays using only locally observable information, as we describe in the next section.

3.2.2 Stochastic Approximation

We identify an online algorithm for updating the relaying probabilities by relays based on locally observed information and show convergence to the sub-gradient dynamics introduced in (3).

We consider updating of the relaying probabilities by a relay r and introduce the following variables per message m of channel c that are locally observable by relay r. Let us introduce $Y_{i,r}(m)$ defined as follows

$$Y_{i,r}(m) = \sum_{u \in \mathcal{U}} \left(\alpha_{i,r,u}(m) - \beta_{i,r,u}(m)\right)$$

[2]Hereinafter, for a relation A, $\mathbb{I}_A$ is equal to 1 if A is true, and 0, otherwise.

where

$$\alpha_{i,r,u}(m) = \left(w_{i,u} \mathbb{I}_{A^{-r}_{i,u}(m) > t_{i,u}} 1_{\tilde{A}_{i,u}(m) \leq t_{i,u}}\right) \mathbb{I}_{c=i}$$

and

$$\begin{aligned}\beta_{i,r,u}(m) &= w_{c,u} \mathbb{I}_{A^{-r}_{c,u}(m) > t_{c,u}} \mathbb{I}_{D_{c,r,u}(m) \leq t_{c,u}} R_{c,r}(m) \times \\ &\times \left[N^i_{c,r,u}(m) \mathbb{I}_{N_{c,r,u}(m)=B_r} + K^i_{c,r,u}(m) \mathbb{I}_{N_{c,r,u}(m)=B_r-1} \right].\end{aligned}$$

For an interpretation of the expected values of $\alpha_{c,r,u}(m)$ and $\beta_{c,r,u}(m)$ we refer to the discussion following Theorem 3.1.

Remark that relay r can observe $Y_{i,r}(m)$ when it receives feedback from users for message m. Each user u interested in messages m must inform relay r whether other relays were able to successfully deliver message m to user u. This can be achieved by letting each user u interested in message m keep a record whether message m could have been received within deadline through a unique path, if it could have been received at all. Then, when for the first time $T_{r,u}(m)$ after the deadline of message m expires, relay r and user u meet, user u sends the required information to r which allows to compute the part of $Y_{i,r}(m)$ corresponding to user u. We assume that relay r updates the relaying probability $x_{i,r}$ at instances $T_{r,u}(m)$, for each message m and interested user u by an online update rule that we describe in the following.

We denote with $S_r(n)$ the n-th feedback from a user to relay r (notice that $(S_r(n), n \geq 0)$ is a superposition of the instances $T_{r,u}(m)$ for every message m and every user u. Denote by $c(n)$ the channel of the corresponding message, and by $u(n)$ the user from which relay r receives feedback. We update the relaying probabilities $(x_{i,r}, i \in I)$ using the following stochastic approximation algorithm per each new feedback received, for $0 < \varepsilon < 1$,

$$x_{i,r}(n+1) = x_{i,r}(n) + \varepsilon \frac{\sum_{j \in \mathcal{U}} \lambda^r_j}{\lambda^r_{c(n)}} (\alpha_{i,r,u(n)}(n) - \beta_{i,r,u(n)}(n)) \quad (5)$$

where λ^r_j is the publishing rate of messages of channel j as observed by relay r. Notice that λ^r_j is equal to the publishing rate of channel j messages if the relay meets a source of channel j at a positive rate. Remark that the update rule (5) conveniently aggregates feedback from different users in an online fashion.

We show that the update rule (5) approximates the sub-gradient algorithm specified in (3). Let $\bar{x}(t)$ be a continuous-time process, defined for channel i and relay r as follows:

$$\bar{x}_{i,r}(t) = x_{i,r}(n), \text{ for } t \in [\varepsilon S_r(n), \varepsilon S_r(n+1)).$$

We next present a convergence result whose proof is provided in § 3.4.2.

THEOREM 3.2. *For the stochastic approximation algorithm (5), $\bar{x}(t) = (\bar{x}_{i,r}(t), i \in I, r \in \mathcal{U})$ uniformly converges over compact time intervals, for asymptotically small parameter $\varepsilon > 0$, to the solution of the following system of ordinary differential equations, for every channel i and relay r,*

$$\frac{d}{dt} \bar{x}_{i,r}(t) = \frac{1}{\tau_r} \sum_{j \in I, u \in \mathcal{U}} w_{j,u} \frac{\partial}{\partial x_{i,r}} p_{j,u}(\bar{x}(t))$$

where $\tau_r := 1/(\sum_{j \in I} \lambda^r_j)$.

3.3 A Baseline Implementation

For concreteness, we describe an implementation of the stochastic approximation algorithm (5). We describe the state kept by individual users and the actions performed at user encounters.

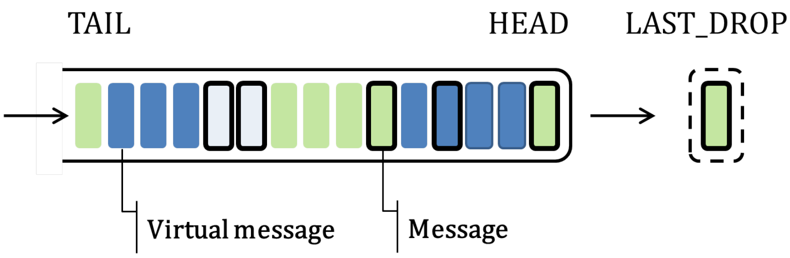

Figure 4: Buffer of a relay containing messages (real and virtual) from three different channels, indicated with different shades.

3.3.1 Relay r

Relay r maintains a buffer of messages observed from sources which includes *real* messages whose payload was downloaded and also *virtual* messages that are messages observed by the relay whose payload was not downloaded. Note that at any time, there are at most B_r real messages in the buffer, where B_r is a configuration parameter, while virtual messages do not consume the buffer of relay r and some control information is maintained for these messages in order to compute $\alpha_{i,r,u}(m)$ and $\beta_{i,r,u}(m)$ for each message m of channel i and each interested user u. Relay r further maintains a reference to the last message dropped from the buffer. Refer to Figure 4 for an illustration of a relay's buffer structure. We now describe the procedures run by relay r when meeting a source and a user, respectively.

Relay r meets a channel-i source. Relay r first updates its estimate, λ_i^r, the publish rate of fresh channel-i messages as observed by relay r. This is done by using a recursive estimator such as exponential weighting smoothing that is commonplace in the design of networking systems. It then downloads each message with probability $x_{i,r}$ in decreasing order of age, and during this procedure, updates the reference to the last dropped message.

Relay r meets user u. Relay r first transmits all messages from its buffer to user u which are of interest to this user (user is subscribed to this channel and the age of a message is smaller than the deadline). The relay maintains two records per message $\texttt{dec\_h}[m][i][u]$ and $\texttt{dec\_ld}[m][i][u]$, where m is identifier of a message, i is identifier of a channel, and u is identifier of a user, which we describe in the following. Notice that these records are created only if a message m is in either state head-of-the-queue or last-dropped at an encounter with a user u and the user expressed interest for message m. At such an event, if m is at the head-of-the-queue, for each channel i, $\texttt{dec\_h}[m][i][u]$ is created and set to the difference of the number channel-i messages in the buffer (real and virtual) and the number of real messages in its buffer (notice that this difference corresponds to the parameter $K^i_{c,r,u}(m)$ where c is the channel of message m). On the other hand, if message m is the last dropped message, for each channel i, $\texttt{dec\_ld}[m][i][u]$ is created and set to the number of real channel-i messages in its buffer (notice that this corresponds to $N^i_{c,r,u}(m)$). The records $\texttt{dec\_h}[m][i][u]$ and $\texttt{dec\_ld}[m][i][u]$ are kept by relay r until feedback from user u for message m is received and at that time are used to adjust the relaying probabilities for relay r, which we describe in more detail shortly.

Finally, relay r receives feedback from user u and updates its relaying probabilities. Specifically, for a message m of channel c, user u sends a ternary feedback $(f_1(m), f_2(m), f_3(m))$ where $f_1(m)$, $f_2(m)$, and $f_3(m)$ are binary values that are used to adjust the relaying probabilities as follows, for a fixed configuration parameter $\varepsilon > 0$,

$$x_{i,r} \leftarrow x_{i,r} + \frac{\varepsilon}{\hat{\lambda}_c^r}\left[f_1(m)\,\mathbb{I}_{c=i} - f_2(m)\Big(f_3(m)\texttt{dec\_h}[m][i][u] + (1-f_3(m))\texttt{dec\_ld}[m][i][u]\Big)\right].$$

Notice that $f_1(m)$ signals whether an increment of the relaying probability $x_{c,r}$ should be made, $f_2(m)$ signals whether a decrement of the relaying probabilities of relay r should be made, and $f_3(m)$ signals whether the decrement is because m was either in the head-of-the-queue or the last-dropped state.

Garbage collection. For each message m observed by relay r, relay r maintains a list of receivers that need to provide feedback for this message. These are receivers that observed message m for the first time from the buffer of relay r or in the last-drop state at relay r. The state maintained for message m is deleted by relay r when feedback is received from all receivers that needed to provide feedback.

3.3.2 Receiver u

For each message m of interest for receiver u, the latter maintains a list of relays, $\texttt{inc\_list}[m]$, which at the time when the feedback collection is completed, contains identities of relays through which m was observed within deadline and the payload of this message could not have been downloaded from neither a source nor another relay, and which thus should receive a positive feedback. Similarly, user u maintains a list of relay identities, $\texttt{dec\_list}[m]$, for which message m was observed in either the head-of-the-queue or the last-dropped state, and which thus should receive a negative feedback.

Receiver u meets relay r. For each observed message m, receiver u maintains a boolean variable $\texttt{seen\_real}[m]$, which will be used to distinguish the case where user u could have downloaded the payload of message m from more than 1 user (either a source or a relay), or otherwise.

We first describe the updates of $\texttt{dec\_list}[m]$. If the variable $\texttt{seen\_real}[m]$ is equal to 0 (i.e. message m has not been downloaded earlier), then, if message m is either head-of-the-queue or last-dropped at relay r, then r is appended to $\texttt{dec\_list}[m]$. Otherwise, if $\texttt{seen\_real}[m]$ is equal to 1, then any entries in the list $\texttt{dec\_list}[m]$ are deleted (because there existed a path to deliver message m to receiver u).

The updates of $\texttt{inc\_list}[m]$ obey the following rules. If message m is observed for the first time by receiver u and is in the buffer of relay r, $\texttt{inc\_list}[m]$ is initialized to r and $\texttt{seen\_real}[m]$ is set to 0, if m is a virtual message, and set to 1, otherwise. On the other hand, if message m was already observed at an earlier instance, the algorithm distinguishes two cases. First, if message m is a real message, then any entries from `inc_list`$[m]$ are removed and r is appended, if `seen_real`$[m]$ is equal to 0; then, `seen_real`$[m]$ is set to 1. Second, m is a virtual message, then r is appended to `inc_list`$[m]$, if `seen_real`$[m]$ is equal to 0.

Finally, feedback is computed as follows. For each message m such that there exists an entry r in either `inc_list`$[m]$ or `dec_list`$[m]$ and the deadline of message m expired, the feedback is set as follows. If r is in `inc_list`$[m]$, then $f_1(m) = 1$, otherwise, $f_1(m) = 0$. If r is in `dec_list`$[m]$ then $f_2(m) = 1$, otherwise $f_2(m) = 0$. If receiver u has downloaded message m, then $f_3(m) = 1$, otherwise $f_3(m) = 0$. Notice that conditional on $f_2(m) = 1$, $f_3(m) = 1$ means that message m was in the head-of-the-queue state when r and u were in contact, and otherwise, in the last-dropped state. Feedback $(f_1(m), f_2(m), f_3(m))$ is communicated to relay r.

Receiver u meets source s. If message m is observed from source s within deadline, then any entries are removed from both `inc_list`$[m]$ and `dec_list`$[m]$, and `seen_real`$[m]$ is set to 1.

3.4 Proofs of Theorems 3.1 and 3.2

3.4.1 Proof of Theorem 3.1

First, note that for every relay $r \in \mathcal{U}$, we have

$$\begin{aligned} 1 - p_{j,u}(x) &= \mathbb{P}_x[A_{j,u} > t_j] \\ &= \mathbb{E}_x[\mathbb{I}_{A^{-r}_{j,u} > t_j}(1 - \mathbb{I}_{\tilde{A}_{j,r,u} \le t_j} R_{j,r})] \\ &= \mathbb{P}_x[A^{-r}_{j,u} > t_j] - \mathbb{E}_x[I_{j,r,u} R_{j,r} \mathbb{I}_{N_{j,r,u} < B_r}]. \end{aligned}$$

Since the ages of messages through paths other than those traversing relay r do not depend on $x_{i,r}$, in order to compute the partial derivative of $p_{j,u}(x)$ with respect to $x_{i,r}$, it suffices to consider only the second term in the right-hand side of the above equation.

Let $M^i_{j,r,u}$ be the number of channel-i messages that are observed by relay r in the time interval $(D_{j,r}, D_{j,r,u}]$. Notice that $M^i_{j,r,u} = N^i_{j,r,u} + K^i_{j,r,u}$, for every channels i and j and relay r. In order to easy the notation, we will use the following shorthand notation $N \equiv N_{j,r,u}$, $N^i \equiv N^i_{j,r,u}$, $N^{-i} \equiv N - N^{-i}_{j,r,u}$, and $M^i \equiv M^i_{j,r,u}$.

Notice that the following holds

$$\mathbb{E}_x[I_{j,r,u} R_{j,r} \mathbb{I}_{N<B_r}] = \mathbb{E}_x[I_{j,r,u} R_{j,r} \mathbb{P}_x[N < B_r | M^i, N^{-i}]]$$

and, then consider

$$\begin{aligned} h(x_{i,r}) &= \mathbb{P}_x[N < B_r | M^i, N^{-i}] \\ &= \mathbb{P}_x[N^i + N^{-i} < B_r | M^i, N^{-i}]. \end{aligned}$$

Since conditional on M^i and N^{-i}, N^i is a binomial random variable with parameters M^i and $x_{i,r}$, we have

$$h(y) = \sum_{j=0}^{M^i} \mathbb{I}_{j<B_r-N^{-i}} \binom{M^i}{j} y^j (1-y)^{M^i - j}.$$

Taking the derivative, we obtain

$$\begin{aligned} h'(y) &= -M^i \binom{M^i - 1}{B_r - 1 - N^{-i}} y^{B_r - 1 - N^{-i}} (1-y)^{(M^i - 1) - (B_r - 1 - N^{-i})} \\ &= -M^i \mathbb{P}[Z = B_r - 1 - N^{-i} | M^i, N^{-i}] \end{aligned}$$

where Z is a binomial random variable $(M^i - 1, x_{i,r})$. Now, it is readily showed that for any two binomial random variables $Z \sim \text{Bin}(m-1, p)$ and $Y \sim \text{Bin}(m, p)$,

$$\mathbb{P}[Z = z] = \left(1 - \frac{z}{m}\right) \mathbb{P}[Y = z] + \frac{z+1}{m} \mathbb{P}[Y = z+1].$$

Therefore, since N^i is a binomial random variable $(M^i, x_{i,r})$, conditional on M^i, we have

$$h'(x_{i,r}) = -\mathbb{E}_x\left[(M^i - N^i) \mathbb{I}_{N=B_r-1} + N^i \mathbb{I}_{N=B_r}\right]. \quad (6)$$

We use the latter identity for the following two cases.

Case 1: $j \neq i$. In this case, we have

$$\begin{aligned} &\frac{\partial}{\partial x_{i,r}} p_{j,r}(x) = \mathbb{E}_x[I_{j,r,u} R_{j,r} h'(x_{i,r})] \\ &= -\mathbb{E}_x[I_{j,r,u} R_{j,r}((M^i - N^i) \mathbb{I}_{N_r=B_r-1} + N^i \mathbb{I}_{N=B_r})] \quad (7) \end{aligned}$$

where the last equality follows from (6).

Case 2: $j = i$. In this case, we have

$$\begin{aligned} &\frac{\partial}{\partial x_{i,r}} p_{i,r}(x) = \mathbb{E}_x[I_{i,r,u}[h(x_{i,r}) + x_{i,r} h'(x_{i,r})]] \\ &= -\mathbb{E}_x[I_{i,r,u}[x_{i,r} h'(x_{i,r}) + h(x_{i,r})]] \\ &= \mathbb{E}_x[I_{i,r,u} \mathbb{I}_{N<B_r}] - \mathbb{E}_x[I_{i,r,u} R_{i,r} h'(x_{i,r})] \quad (8) \\ &= \mathbb{E}_x[I_{i,r,u} \mathbb{I}_{N<B_r}] - \\ &\quad \mathbb{E}_x[I_{i,r,u} R_{i,r}((M^i - N^i) \mathbb{I}_{N=B_r-1} + N^i \mathbb{I}_{N=B_r})] \quad (9) \end{aligned}$$

where (8) holds because $R_{i,r}$ and $I_{i,r,u} h'(x_{i,r})$ are mutually independent random variables and (9) follows from (6).

The asserted result follows from (7) and (9) by turning back to the original notation.

3.4.2 Proof of Theorem 3.2

The result follows from Kushner and Yin [15][Chapter 12, Theorem 3.1] in view of the following facts. First, since we assume that the message publishing by sources and user mobility are stationary ergodic processes, so that we have for every relay r,

$$\lim_{N \to \infty} \frac{1}{N} \sum_{n=1}^{N} (T_r(n+1) - T_r(n)) = \tau_r$$

where recall $T_r(n)$, $n \geq 1$, are instances at which feedback is received by a relay r.

Second, we establish that the following holds for every channel i and relay r,

$$\lim_{N \to \infty} \frac{1}{N} \sum_{n=1}^{N} \frac{\sum^r_{j \in I} \lambda^r_j}{\lambda_{c(n)}} Y_{i,r}(n) = \frac{\partial}{\partial x_{i,r}} \sum_{j \in I, u \in \mathcal{U}} w_{j,u} p_{j,u}(x)$$

where $c(n)$ and $Y_i(n)$ are under $x(n)$ fixed to x, for every $n \geq 1$. Notice that

$$\begin{aligned} \frac{1}{N} \sum_{n=1}^{N} \frac{\sum_j \lambda^r_j}{\lambda^r_{c(n)}} Y_{i,r}(n) &= \frac{1}{N} \sum_{n=1}^{N} \sum_{j \in I} \frac{\sum_j \lambda^r_j}{\lambda^r_{c(n)}} Y_{i,r}(n) \mathbb{I}_{c(n)=j} \\ &= \sum_{j \in I} \frac{1}{N_j} \sum_{l=1}^{N_j} \frac{N_j}{N} \frac{\sum_j \lambda^r_j}{\lambda^r_j} Y_{i,r}(n^j(l)) \end{aligned}$$

where for each fixed channel j, $n^j(l)$ is a subsequence at which $c(n) = j$ and N_j is the length of this subsequence. Noting that for every channel j, $\lim_{N_j \to \infty} N_j / N = \lambda^r_j / \sum_j \lambda^r_j$ and

$$\lim_{N_j \to \infty} \frac{1}{N_j} \sum_{l=1}^{N_j} Y_{i,r}(n^j(l)) = \frac{\partial}{\partial x_{i,r}} \sum_{u \in \mathcal{U}} w_{j,u} p_{j,u}(x).$$

The asserted result follows.

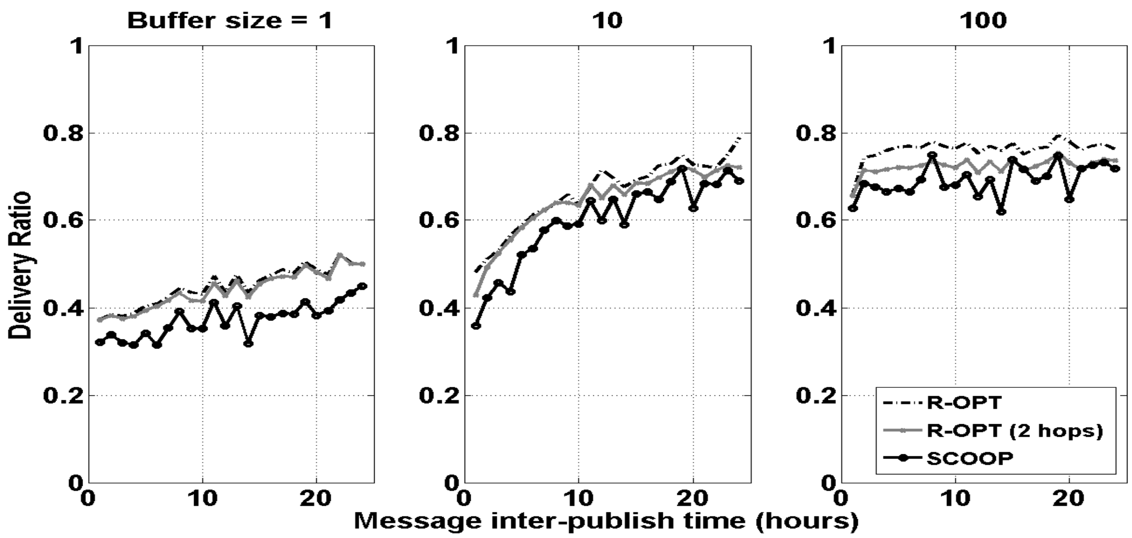

Figure 5: SCOOP's performance compared to R-OPT at various publishing rates and buffer sizes.

4. PERFORMANCE

This section presents the performance evaluation of SCOOP by comparing with current state-of-the-art protocols based on realistic mobility scenarios. The evaluation examines the significance and effect of various parameters such as the buffer size, the publishing rate of messages, and the message expiration deadline. We conducted simulations using our own built simulator that receives as input: i) a mobility trace that specifies contacts between nodes over time; and, ii) a message publishing trace specifying the time instances and source channel of published messages. All results reported here are obtained for $\varepsilon = 0.01$ (we varied ε around this value and observed only negligible impact on performance).

Since previous work does not support multicast delivery of streams, we have adapted RAPID [3] to support the delivery of messages from a source to multiple destinations. Previous evaluations of RAPID show that it outperforms other strategies and hence, it is the baseline used to compare SCOOP. In RAPID, a relay forwards messages greedily aiming at maximizing the marginal utility which is similar in spirit to our scheme. The utility might be, as in our system, the probability to deliver messages to destinations within specified deadlines. In order to estimate the marginal utility gain, a relay has to ideally know which other nodes possess replicas of the message and when they expect to meet the destination. As this requires global knowledge, strategies to estimate the gain resort to approximation and simplifying assumptions about user mobility (e.g., statistically identical individual user movement, independence of delivery paths, and some Poisson approximations). In particular, RAPID assumes that delays through various relays are statistically independent, an assumption contradicted by the experimental results presented in Section 2.

To adapt RAPID for a multicast scenario, we examine the aggregate utility of the probability of delivering a message across all destinations for every message. We further compare SCOOP against an optimized version of RAPID, that will henceforth be referred to as R-OPT. The optimizations include the following: (1) Each relay node has *complete* knowledge of the dissemination state, i.e., at any point in time each node knows exactly which messages are carried by all nodes; (2) Each relay knows the complete matrix of mean pairwise inter-contact times for all nodes. As discussed above, in the original RAPID algorithm, these quantities are approximated since it is practically infeasible for all nodes to have a complete view of the whole network. In essence, R-OPT presents *a best-case scenario* for RAPID, where *each node exactly knows all the state required by the dissemination algorithm.*

Fig. 5 presents how SCOOP performs against R-OPT, for the DieselNet trace that was initially used to evaluate RAPID in [3]. For comparison purposes, we further highlight R-OPT's performance by restricting it to two-hop relay-paths only. Fig. 5 highlights the message delivery ratio as we vary the node buffer sizes and the source publishing rate. Each point represents mean value computed based on ten runs where five sources and five destinations were chosen randomly, and the rest of the nodes are relays. We computed the average delivery ratio and confidence intervals for 95% of confidence, but omit to show the confidence intervals in figures for visualization purposes, whose lenghts were about 10%. Message deadline is set to 1 week. Finally, SCOOP's initial relaying probabilities are set to 1.

We observe that despite relying only on locally observable information, SCOOP provides good performance that is in many cases near to R-OPT for different choices of relay buffer size and message publishing rates. Note that the delivery ratio gap between SCOOP and R-OPT is approximately constant for a wide range of message publish rate. We further note that the delivery ratio gap between R-OPT with and without the two-hop relaying path restriction is small which conforms with the analysis of path delays in Section 2.

Fig. 6 examines similar scenarios, but, in this case, the buffers are fixed to intermediate size (equal to 10 messages), and the message deadlines vary from 1 to 14 days. As expected, increasing message deadlines improves the performance for all algorithms. We observe that SCOOP yields comparatively high delivery ratios, especially, for the range of small message deadlines. In this case, as well, the 95% confidence intervals were approximately of 10% length and are omitted to simplify the presentation.

For completeness, Fig. 7 presents performance of the three dissemination protocols for one of the other mobility traces, namely the *SF-Taxis* trace. The figure is equivalent to the middle one of Fig. 6 in terms of buffer size and message inter-publishing time.

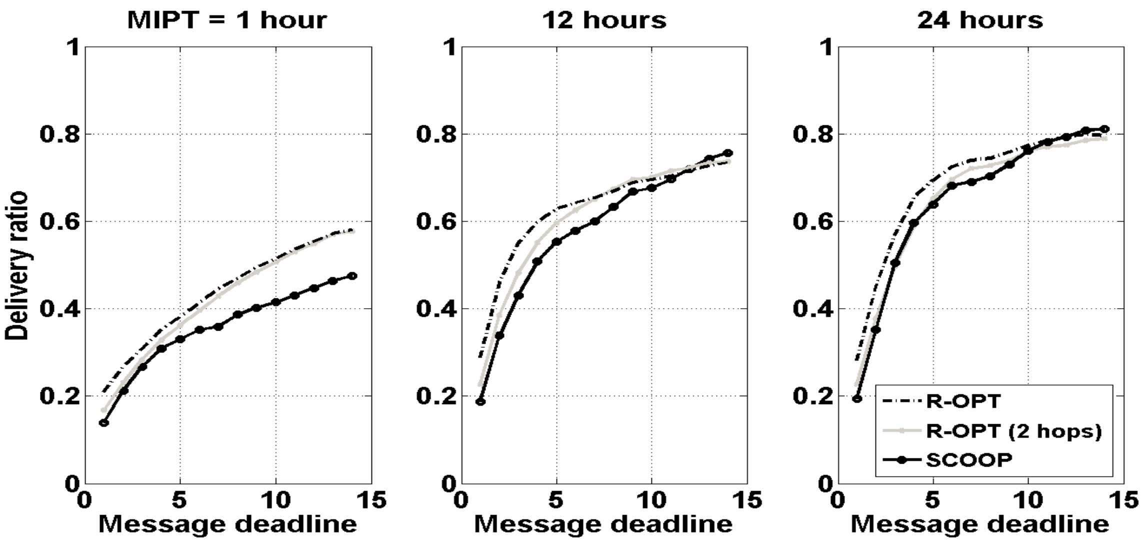

Figure 6: SCOOP**'s performance compared to** R-OPT **with varying message deadlines in days. Buffer size = 10. (MIPT = Message inter-publish time.)**

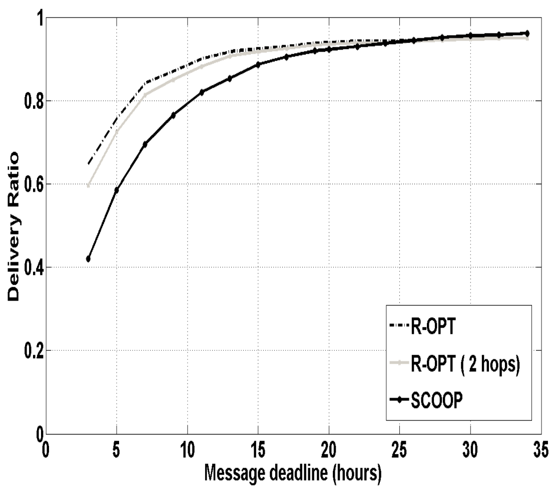

Figure 7: SCOOP**'s performance compared to** R-OPT **with varying message deadlines for the *SF-Taxis* trace. Buffer size = 10 and message inter-publish time = 12 hours.**

However, for this scenario, we select quite aggressive deadlines (a few hours) to stress-test the dissemination protocols. For efficiency reasons, we have randomly selected 50 nodes and used the first two weeks of the *SF-Taxis* trace.[3]. Again, the average delivery ratio improves for all three protocols with the message deadline and R-OPT performance is not significantly affected by restricting the relaying to one- or two-hop paths. We observe that SCOOP provides average delivery ratios that for sufficiently large message deadline is within 10% of the omniscuous R-OPT.

[3]Note that our R-OPT implementations are quite computationally expensive since they maintain all the possible state in the system (e.g., inter-contact times and utilities are estimated across all messages and nodes at every meeting instance).

5. RELATED WORK

Several proposals for routing or disseminating messages in DTNs have been made and we refer to [28, 29] and [3] for an overview of the state-of-the-art. Routing protocols in DTNs are usually classified into two broad categories: (1) *forwarding protocols* that keep a single copy of the message to deliver in the network, see e.g. [12]; (2) *epidemic routing protocols* that replicate messages at transfer opportunities to find a path to the destination. We are interested in the second category of protocols since our system goal is to disseminate the channel contents to multiple interested users.

In most of the algorithms proposed so far, nodes limit the number of times they forward a message using various kinds of information. For example, in [26, 5], the routing uses the number of replicas already generated by nodes to decide whether new replicas should be created; most of the protocols use the history of node encounters to infer likelihood of message delivery if forwarded to a particular node (e.g. [19, 6, 5, 22, 3, 17]); replication algorithms may also try to leverage the social structure of the network for message forwarding decisions [8] as socially-related nodes are more likely to meet. Some routing algorithms account for storage limits at nodes, see e.g., [9, 19, 18, 6, 27]. Only a few papers, e.g. [5, 3], propose algorithms that in addition, try to cope with transmission or bandwidth constraints (the amount of information that can be exchanged per contact is limited).

Our framework differs from all previous proposals. First, it addresses multi-point (channel sources) to multi-point (interested users) communication. The closest related work are the protocols RAPID [3] and the one proposed in [14]. As discussed for RAPID in Section 4, these protocols are based on simplifying assumptions regarding user mobility (e.g. statistically identical individual user movements, independence of delivery paths). It remains unclear whether they perform well under general mobility models. In contrast, our framework identifies decentralized relaying strategies that provably converge to optimal solutions of a global system objective, and allow for general user mobility, and thus alleviate to resorting to any simplifying assumptions that may not be met in practice. Finally, we remark that aiming at a global system objective underlie some other work on the design of protocols for opportunistic communications. For example, [24] but the problem therein is optimizing caching of content and is thus different.

6. CONCLUDING REMARKS

We presented SCOOP, a relaying strategy that supports multicast delivery of information streams to interested users in mobile networks. SCOOP is designed to optimize a well-defined global system objective using an algorithm that is fully decentralized, requires only locally observable information by individual nodes, allows for general user mobility, and converges to optimal points of the underlying global system objective. The relaying algorithm alleviates relaying on restrictive user mobility assumptions, such as statistical independence of delays through different paths, and unlike to most previous proposals does not require nodes to posses any global knowledge.

For future work, it would be interesting to evaluate analytically how fast is the convergence and how it depends on user mobility, publishing rates, and the configuration parameters of the algorithm. It would also be of interest to pursue more extensive analysis by simulations to further evaluate SCOOP's performance across a wide range of practical mobility scenarios, in particular, investigate potential benefits over alternative approaches in cases where path delays exhibit strong correlations.

7. REFERENCES

[1] AT&T doubles 3G cell capacity but still faces backhaul challenges, 2010. `http://connectedplanetonline.com/3g4g/news/att-doubles-3g-010510/`.

[2] A. Balasubramanian, B. N. Levine, and A. Venkataramani. Enabling Interactive Web Applications in Hybrid Networks. In *Proc. ACM Mobicom*, 2008.

[3] A. Balasubramanian, B. N. Levine, and A. Venkataramani. Replication Routing in DTNs: A Resource Allocation Approach. *IEEE/ACM Trans. on Networking*, 18(2):596–609, April 2010.

[4] BT FON. `http://www.btfon.com/`.

[5] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine. MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks. In *Proc. of IEEE INFOCOM*, April 2006.

[6] B. Burns, O. Brock, and B. N. Levine. MV Routing and Capacity Building in Disruption Tolerant Networks. In *Proc. of IEEE INFOCOM*, March 2005.

[7] X. R. Cao. Perturbation analysis of discrete event systems: Concepts, algorithms, and applications. *European Journal of Operational Research*, 91(1):1–13, May 1996.

[8] P. Costa, C. Mascolo, M. Musolesi, and G. P. Picco. Socially-aware Routing for Publish-Subscribe in Delay-tolerant Mobile Ad Hoc Networks. *IEEE Journal on Selected Areas In Communications (JSAC)*, 26(5):748–760, June 2008.

[9] J. Davis, A. Fagg, and B. N. Levine. Wearable Computers and Packet Transport Mechanisms in Highly Partitioned Ad hoc Networks. In *Proc. of IEEE Intl. Symp on Wearable Computers (ISWC)*, 2001.

[10] D. Gunawardena, T. Karagiannis, A. Proutiere, E. Santos-Neto, and M. Vojnovic. Scoop: Decentralized and opportunistic multicasting of information streams. Technical Report MSR-TR-2010-98, Microsoft Research, 2010.

[11] D. Gunawardena, T. Karagiannis, A. Proutiere, and M. Vojnovic. Characterizing podcast services: Publishing, usage, and dissemination. In *Proc. of ACM IMC 2009*, 2009.

[12] S. Jain, K. Fall, and B. Patra. Routing in a delay tolerant network. In *Proc. of ACM SIGCOMM*, 2004.

[13] G. Karlsson, V. Lenders, and M. May. Delay-tolerant broadcasting. *IEEE Transactions on Broadcasting*, 53(2):369 – 381, March 2007.

[14] A. Krifa, C. Barakat, and T. Spyropoulos. Optimal buffer management policies for delay tolerant networks. In *Proc. of SECON*, June 2008.

[15] H. J. Kushner and G. G. Yin. *Stochastic Approximation and Recursive Algorithms and Applications*. Springer, 2nd edition, 2003.

[16] V. Lenders, G. Karlsson, and M. May. Wireless Ad Hoc Podcasting. In *Proc. of SECON*, June 2007.

[17] H. Liang, J.-Y. Le Boudec, and M. Vojnovic. Optimal channel choice for collaborative ad-hoc dissemination. In *Proc. of IEEE Infocom*, March 2010.

[18] C. Lindemann and O. P. Waldhorst. Modeling epidemic information dissemination on mobile devices with finite buffers. In *Proc. of ACM SIGMETRICS*, 2005.

[19] A. Lindgren, A. Doria, and O. Schelén. Probabilistic routing in intermittently connected networks. *ACM SIGMOBILE Mob. Comput. Commun. Rev.*, 7(3):19–20, 2003.

[20] M. McNett and G. M. Voelker. Access and Mobility of Wireless PDA Users. In *Mobile Computing Communications Review*, 2005.

[21] Morgan Stanely Research. The Mobile Internet Report, 2009. `http://www.morganstanley.com/institutional/techresearch/pdfs/mobile_internet_report.pdf`.

[22] S. Nelson, M. Bakht, and R. Kravets. Encounter-Based Routing in DTNs. In *Proc. of IEEE INFOCOM*, April 2009.

[23] M. Piorkowski, N. Sarafijanovic-Djukic, and M. Grossglauser. A Parsimonious Model of Mobile Partitioned Networks with Clustering. In *Proc. of COMNETS*, Jan. 2009.

[24] J. Reich and A. Chaintreau. The age of impatience: optimal replication schemes for opportunistic networks. In *Proc. of ACM CoNEXT*, 2009.

[25] J. Scott, R. Gass, J. Crowcroft, P. Hui, C. Diot, and A. Chaintreau. CRAWDAD trace cambridge/haggle/imote/infocom (v. 2006-01-31), January 2006.

[26] T. Spyropoulos, K. Psounis, and C. S. Raghavendra. Spray and wait: an efficient routing scheme for intermittently connected mobile networks. In *Proc. of WDTN: ACM SIGCOMM workshop on Delay-tolerant networking*, pages 252–259, 2005.

[27] X. Wang, Y. Shu, Z. Jin, Q. Pan, and B. Sung Lee. Adaptive Randomized Epidemic Routing for Disruption Tolerant Networks. In *Proc. of IEEE MASS*, 2009.

[28] Z. Zhang. Routing in intermittently connected mobile ad hoc networks and delay tolerant networks: Overview and challenges. *IEEE Communications Surveys and Tutorials*, 8(1-4):24–37, 2006.

[29] Z. Zhang and Q. Zhang. Delay/disruption tolerant mobile *ad hoc* networks: latest developments. *Wireless Communications and Mobile Computing*, 7(10):1219–1232, 2007.