Optimal Channel Choice for Collaborative Ad-Hoc Dissemination

Liang Hu*, Jean-Yves Le Boudec†, Milan Vojnović‡
* Technical University of Denmark † EPFL ‡ Microsoft Research

***Abstract*—Collaborative ad-hoc dissemination of information has been proposed as an efficient means to disseminate information among devices in a wireless ad-hoc network. Devices help in forwarding the information channels to the entire network, by disseminating the channels they subscribe to, plus others. We consider the case where devices have a limited amount of storage that they are willing to devote to the public good, and thus have to decide which channels they are willing to help disseminate. We are interested in finding channel selection strategies which optimize the dissemination time across the channels. We first consider a simple model under the random mixing assumption; we show that channel dissemination time can be characterized in terms of the number of nodes that forward this channel. Then we show that maximizing a social welfare is equivalent to an assignment problem, whose solution can be obtained by a centralized greedy algorithm. We show empirical evidence, based on Zune data, that there is a substantial difference between the utility of the optimal assignment and heuristics that were used in the past. We also show that the optimal assignment can be approximated in a distributed way by a Metropolis-Hastings sampling algorithm. We also give a variant that accounts for battery level. This leads to a practical channel selection and re-selection algorithm that can be implemented without any central control.**

I. Introduction

Several applications relying on opportunistic data transfers between devices have been proposed recently. In [1], the authors propose a wireless ad-hoc podcasting system, where, in addition to downloading content onto devices while docked to a desktop computer, the content is exchanged between devices while users are on the go. They propose several heuristics for content exchange between devices based on the inferred preference of the user owning a device and that of encountered devices. Another related system is CarTorrent [2], a BitTorrent-style content dissemination system designed to exploit the wireless broadcast nature, where the authors propose various solicitation strategies.

We call *channel* an abstraction for various information feeds that generate content recurrently over time. For example, a podcast feed is a channel as well as a profile page of an online social network application (e.g. Facebook or Twitter). While many such services can well be provisioned at mobile devices by accessing the cloud, it is still of interest to speed up information dissemination by augmenting it with device-to-device information transfer. Efficient multi-channel information dissemination through infrastructure and multi-hop wireless transfer would well support various mobile content sharing applications, e.g. Serendipity [3], in particular, in environments where access to the cloud is intermittent, either because of the lack of connectivity or access cost (e.g during roaming).

We are interested in scenarios where nodes are willing to devote some amount of their resources to help content dissemination. Now the number of information channels can be very large compared to user's interest; for example in the Zune dataset there are 8000+ podcast channels and each user subscribes 6 channels on average [4], [5]. In such a setting, we propose to limit the amount of resource that a node devotes to the dissemination of channels other than the ones it subscribes to. This is motivated by the cost for a user in terms of bandwidth usage during meetings, energy consumption, and perhaps also storage. We thus assume that each user device has to decide which channels to help disseminate, in addition to the subscribed ones. We consider a setting where users are cooperative in optimizing the content dissemination, an assumption that underlies the prior work [1]. The cooperation could be induced through various mechanisms like in any other peer-to-peer service. One implicit incentive is indirect reciprocity where users expect that other users would help disseminate the channels subscribed by this user, so the user may well be willing to reciprocate.

We are interested in finding channel selection strategies which optimize channel dissemination times with respect to a system welfare objective. The key assumption that facilitates our framework is a relation between the channel dissemination time and the fraction of the nodes that forward a given channel. Such a relation can be obtained by modeling or empirical analysis, examples of which we show in this paper. However, it is noteworthy that in this paper, we do not advocate any specific function to describe the relation between the dissemination time and the fraction of the forwarding nodes– a thorough analysis of this is left for future work. We cast the problem in the framework of system welfare optimization where the objective is to optimize an aggregate of the utility functions associated with individual channels. We show that for a broad class of utility functions, optimizing the system welfare is equivalent to an assignment problem whose solution can be obtained by a centralized greedy algorithm [6]. We provide empirical evidence, based on real-world data about subscriptions of Zune [4], [5] users to podcasts, that there is a substantial difference between optimal system welfare assignment and some heuristics that were used in the past.

Then we consider the problem of defining a practical, distributed algorithm run by individual nodes to attain a given system objective. We show that the optimal assignment can be approximated in a distributed way by a Metropolis-Hastings sampling algorithm. The algorithm requires knowledge about the fractions of nodes subscribed or forwarding given channel which can be estimated based on local observations by each individual node. We also identify a class of Metropolis-Hastings algorithms that do not require any estimation. We show simulation results that demonstrate that our proposed distributed algorithms concentrate near the optimum system welfare with the rates of convergence of interest in practice.

Our contributions can be summarized as follows:

- We propose a framework for optimizing the dissemination of *multiple* information channels in wireless ad-hoc networks. The optimization is with respect to dissemination times of individual channels subject to the end-user cache capacity constraints. To the best of our knowledge, this is the first proposal for optimizing dissemination of *multiple* information channels in wireless ad-hoc networks with respect to a global system objective.
- The framework enables a *direct engineering* by allowing derivation of the algorithms that decide which channels are helped by which users so as to optimize a given system objective.
- The framework also allows a *reverse engineering* so that for some given channel selection algorithms deployed by individual nodes, we can determine which underlying global system objective is optimized.
- We show that an optimum system assignment of users to channels for forwarding can be found by a centralized greedy algorithm for a broad class of system objectives identified in this paper.
- Using the data about subscriptions of Zune users to audio podcasts, we demonstrate that there exist scenarios where for given system objective, significant gains can be attained by the optimum system assignment over some heuristics suggested by previous work [1].
- We show that the optimum system objective can be well approximated by a distributed algorithm based on the Metropolis-Hastings sampling run by individual nodes using only local observations.
- We show how to incorporate in our framework and algorithms the objective to optimize the battery expenditure.
- We present extensive simulation results that provide validation and demonstrate practicality of the proposed algorithms.

The paper is structured as follows. Sec. II introduces our system model and notation. Sec. III presents modeling and empirical analysis about the relation between the channel dissemination time and the fraction of the nodes that forward the channel. In this section, we also define the system objective, the utilities associated to channels, and discuss some of their properties. Sec. IV-A presents the system welfare problem and the result that the problem can be solved by a centralized greedy algorithm. Sec. V presents results on the gain of the optimum system welfare based on the Zune data. Sec. VI presents our Metropolis-Hastings algorithms. In Sec. VII we show simulation results. Finally, related work is discussed in Sec. VIII and Sec. IX concludes the paper. We defer some of our proofs to Appendix [7].

II. System Model and Notation

We consider a system of N wireless nodes, or users, participating in the ad-hoc dissemination of J channels. We denote with $\mathcal{U}$ and $\mathcal{J}$ the sets of user and channels, respectively. Every node, say, u has a list $S(u)$ of *subscribed* channels. In the context of this study, we assume that $S(u)$ is fixed for every u. In contrast, every node maintains a variable list of *helped* channels, i.e. channels that this node keeps in its public cache in order to facilitate their dissemination.

When two nodes meet, they update their cache contents. More precisely, if nodes u and u' meet, u gets from u' the content that is newer at u' for the channels that u either subscribes to or helps, and vice-versa. We do not account for the overhead of establishing contacts and negotiating content updates. We assume that when nodes meet the contact duration is large enough for all useful contents to be exchanged, i.e. we assume that the bottlenecks in the system performance are the disconnection times and cache content. In addition, we assume that, once in a while, a node gets direct contact to the Internet and downloads fresh content for the subscribed or helped channels.

At any given point in time, we call x the global system configuration, defined by

$$x_{u,j} = 1 \Leftrightarrow \text{ node } u \text{ subscribes to or helps channel } j.$$

Let $H(u,x)$ be the set of channels helped by node u when the configuration is x and let $F(u,x)$ be the set of forwarded channels, i.e.

$$F(u,x) = H(u,x) \cup S(u), \ u \in \mathcal{U}.$$

We assume that every node u has a maximum cache capacity C_u, to simplify we count it in the number of channels. We assume that[1] $C_u \geq |S(u)|$, i.e. every node can store all the subscribed channels. The configuration is thus constrained by

$$|F(u,x)| \leq C_u \text{ for all } u \in \mathcal{U}.$$

The problem is then to find a configuration x that satisfies these constraints and maximizes some appropriate performance objective, defined in the next section. Further, we want to find a method to approximate the optimal configuration in a fully distributed way which we do in Sec. VI.

We use the following notation:

$$\begin{aligned} s_j &= \text{proportion of nodes that subscribe to channel } j \\ f_j(x) &= \text{proportion of users that forward channel } j \\ &= \frac{1}{N}\sum_{u\in\mathcal{U}} x_{u,j}. \end{aligned}$$

Without loss of generality and unless indicated otherwise, we assume that channels are labeled in nonincreasing order with respect to their subscription popularity, i.e. $s_1 \geq \cdots \geq s_J$. Also $\vec{s} = (s_1,\ldots,s_J)$ and $\vec{f} = (f_1,\ldots,f_J)$.

[1] $|A|$ denotes the cardinality of a finite set A.

III. Dissemination Time and Utility

To get a better handle on the performance objective we first use an epidemic style analysis, using ordinary differential equations, and then do the same through analysis of real-world mobility traces.

A. Model-Based Dissemination Time

Consider an arbitrary channel j and let the time origin be the time at which a piece of information was created by the source of channel j. We assume that the configuration x is fixed and omit it from the notation in this section. Let $\sigma_j(t)$ be the proportion of the subscribers to channel j that at time t have received a piece that was originated by the source in the time interval $[0,t]$. Let $\phi_j(t)$ be the proportion of the forwarders of channel j that have received a piece made available by the source in the time interval $[0,t]$. The dynamics of the system can be described by the following system of ordinary differential equations:

$$\frac{d}{dt}\sigma_j(t) = (\lambda_j + \eta\phi_j(t))(s_j - \sigma_j(t)) \quad (1)$$

$$\frac{d}{dt}\phi_j(t) = (\lambda_j + \eta\phi_j(t))(f_j - \phi_j(t)) \quad (2)$$

where λ_j is the contact rate between a node and an infrastructure able to deliver channel j, and η is the contact rate between nodes. These equations hold under the "random node mixing" assumption and are asymptotically valid when N is large. The system (3)-(2) admits a closed-form solution. Here we only state the solution $\sigma_j(t)$ for $t \geq 0$, as this suffices to compute the dissemination time of a piece.[2] We have

$$\sigma_j(t) = \sigma_j(0) + (s_j - \sigma_j(0)) \times \frac{(\lambda_j + \eta\phi_j(0))(1 - e^{-(\eta f_j + \lambda_j)t})}{\lambda_j + \eta\phi_j(0) + \eta(f_j - \phi_j(0))e^{-(\eta f_j + \lambda_j)t}}. \quad (3)$$

Dissemination Time: Let t_j be the time for a fraction α of the subscribers to channel j to recieve a piece available at the source at time 0 or a more recent piece. We refer to t_j as the dissemination time for channel j that is a metric of our primary interest. Setting $\sigma_j(t_j) = \alpha$, it follows

$$t_j = \frac{1}{\lambda_j + f_j\eta} \ln \frac{(f_j - \phi_j(0))\eta K_j + \lambda_j + \eta\phi_j(0)}{(\lambda_j + \eta\phi_j(0))(1 - K_j)} \quad (4)$$

where $K_j = \left(\alpha - \frac{\sigma_j(0)}{s_j}\right) / \left(1 - \frac{\sigma_j(0)}{s_j}\right)$.

Proposition III.1. *The dissemination time t_j is a monotonic nonincreasing, strictly convex function of f_j.*

Proof is in Appendix [7] A. Of particular interest is the small injection rate regime, where the dissemination is dominated by epidemic content. In this case, we have $\sigma_j(0) \ll \lambda_j/\eta \ll s_j$ and $\phi_j(0) \ll \lambda_j/\eta \ll f_j$ and Eq.(4) becomes

$$t_j \approx \frac{1}{\eta f_j}\left(\ln\frac{\alpha}{1-\alpha} + \ln\frac{\eta f_j}{\lambda_j}\right). \quad (5)$$

[2]The interested reader may find more details in [7].

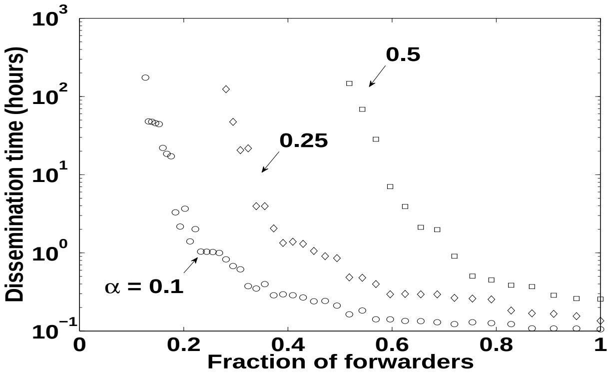

Fig. 1. Dissemination time versus the fraction of forwarding nodes in CAM data. Each mark shows the median value of the dissemination time obtained by taking each node as a source and repeating for 10 random selections of the forwarding nodes.

B. Empirical Dissemination Time

We consider the dissemination time evaluated from real mobility traces. In particular, we consider (CAM) a data trace of human mobility in the area of Cambridge, UK [8] and (SF-TAXI) a data trace of the routes of taxis in the area of San Francisco [9]. The CAM dataset contains information about the contacts between 36 human-carried, Bluetooth-equipped devices over slightly more than 10 days. SF-TAXI contains GPS coordinates for each of about 500 taxis over a month period. For the latter trace, we define a contact between two nodes if the distance between two nodes is smaller or equal to 500 meters [10].

We infer the dissemination times by conducting the following experiment. For given data trace (either CAM or SF-TAXI), we fix a portion of nodes and then pick uniformly at random the given portion of nodes from the set of all nodes and designate them as forwarders. We then inject a message to one of the forwarders at an instance of time and then pass through the trace forward in time, recording the instances at which a forwarder first received the message by encountering a forwarder that already received the message. For the CAM data, we repeat the experiment for each source and 10 random samples of the set of forwarders. Finally, for each given portion of the forwarding nodes, we compute the median dissemination time.

Fig. 1 shows the empirical dissemination time versus the portion of the forwarding nodes for CAM trace. Similar results hold for the SF-TAXI trace; available as Fig. 2 in [7]. In both cases, they confirm that the dissemination time is well fitted by a curve that exhibits diminishing returns with increasing the number of forwarders.

C. Utility Function

We assume that for each channel j there is an underlying utility function $U_j(t_j)$ that specifies the satisfaction of a subscriber with the channel j given that the dissemination time is t_j. It is natural to assume $U_j(t_j)$ is a nonincreasing function of t_j. We will discuss later in this section some additional conditions for a channel utility function.

We denote with $V_j(f_j) = U_j(t_j(f_j))$ the utility function for channel j with respect to the fraction of the users who forward channel j. It is natural to assume that $V_j(f_j)$ is a monotonic nondecreasing function of f_j. This indeed follows if both $U_j(t_j)$ and $t_j(f_j)$ are noincreasing functions which are rather natural assumptions.

It remains to discuss what the system welfare utility is, i.e. when considering all channels together. We admit standard definition that the system welfare is a weighted sum of the utilities over all channels, i.e. for given positive weights $\vec{w} = (w_1, \ldots, w_J)$,

$$V(\vec{f}) = \sum_{j\in\mathcal{J}} w_j V_j(f_j).$$

Two special cases may be of interest, which correspond to different fairness objectives. The former is channel centric, in that it considers each channel as one entity, regardless of the number of subscribers. This utility is obtained by setting all the weights w_j to 1, hence we have

$$V_{CH}(\vec{f}) = \sum_{j\in\mathcal{J}} V_j(f_j) \tag{6}$$

where V_j is a per-channel metric, for example as in Eq.(4) or Eq.(5). The latter is user centric and has the weights such that w_j is proportional to the proportion of j-subscribers, s_j, hence we consider

$$V_{US}(\vec{f}) = \sum_{j\in\mathcal{J}} s_j V_j(f_j) \tag{7}$$

with V_j as before.

In Sec. VI we will show that this utility framework can easily be extended to battery saving.

Sufficient Conditions for a Concave Utility: We discuss a set of *sufficient* conditions that ensure the utility $V_j(f_j)$ is a concave function of f_j. This class of utility functions will be of interest in Section IV-A.

Proposition III.2. *Suppose (C1) $U_j(t_j)$ is a nonincreasing, concave function of t_j and (C2) $t_j(f_j)$ is a convex function of f_j. Then, $V_j(f_j)$ is a concave function of f_j.*

Proof derives from elementary convexity properties and is available in [7]. Condition (C1) says that the utility function $U_j(t_j)$ captures the increasing dissatisfaction of a subscriber of channel j with the dissemination time t_j. This is a realistic assumption that captures the scenarios where users would like to receive fresh content within some time horizon since the content generation and become inreasingly dissatisfied as the delivery time increases beyond the time horizon. Condition (C2) means that the dissemination time $t_j(f_j)$ exhibits diminishing returns with increasing the portion of forwarders f_j. We have already demonstrated cases in Section III-A and Section III-B that support this assumption.

IV. System Welfare Problem

A. The Greedy Algorithm

We consider a system welfare problem where the objective is to optimize the aggregate utility of channel dissemination times subject to the end-user capacity constraints. Solving the system welfare problem amounts to finding an *assignment* of users to channels that solves the following problem:

$$\begin{array}{rl}
\text{SYSTEM} & \\
\text{maximize} & \displaystyle\sum_{j=1}^{J} w_j V_j\left(\frac{1}{N}\sum_{u=1}^{N} x_{u,j}\right) \\
\text{over} & x_{u,j} \in \{0,1\} \\
\text{subject to} & \displaystyle\sum_{j=1}^{J} x_{u,j} \le C_u \\
 & x_{u,j} = 1,\ (u,j):\ j \in S(u).
\end{array}$$

Defining the system welfare utility as a sum of individual utilities is rather standard in the framework of resource allocation. Note that in SYSTEM w_j are positive constants that can be arbitrarily fixed. In particular, it is of practical interest to define w_j proportional to the fraction of subscribers to channel j. In this case, the utility $V_j(\cdot)$ can be interpreted as the utility for channel j for a typical subscriber of channel j.

We show that we can rephrase SYSTEM as an equivalent optimization over the number of users who help forward individual channels. Let H_j be the number of users who help forward channel j and let $\vec{H} = (H_1, \ldots, H_J)$. Let us define $v(A)$ for $A \subseteq \mathcal{J}$ by

$$v(A) = \sum_{u\in\mathcal{U}} \min\left(\sum_{j\in A} 1_{j\in\mathcal{J}\setminus S(u)}, C_u - |S(u)|\right) \tag{8}$$

and let $P(v)$ be the polyhedron defined by

$$P(v) = \{x \in \mathbb{N}_0^J :\ \sum_{j\in A} x_j \le v(A),\ A \subseteq \mathcal{J}\}.$$

We consider the following problem.

$$\begin{array}{rl}
\text{SYSTEM-H} & \\
\text{maximize} & \displaystyle\sum_{j=1}^{J} w_j V_j\left(s_j + \frac{1}{N}H_j\right) \\
\text{over} & \vec{H} \in P(v).
\end{array}$$

The following result establishes a connection between SYSTEM and SYSTEM-H.

Theorem IV.1. *The optimal value of the solution of* SYSTEM *is equal to that of* SYSTEM-H.

Proof: Proof is based on a reduction to a max-flow problem and is available in Appendix [7]. ■

The interested reader is referred to Appendix [7] B where we consider a relaxed version of SYSTEM which allows providing some characterization of the solution to this relaxed version.

We next provide conditions under which SYSTEM-H can be solved by a greedy allocation of helpers to channels. Let us denote with $\Delta_j V(\vec{s} + \vec{H}/N)$ the increment of the aggregate utility by assigning a user to channel j, i.e.

$$\begin{aligned} & \Delta_j V(\vec{s}+\vec{H}/N) \\ = \quad & V(\vec{s}+(\vec{H}+e_j)/N) - V(\vec{s}+\vec{H}/N) \\ = \quad & w_j\left[V_j(s_j+(H_j+1)/N) - V_j(s_j+H_j/N)\right] \end{aligned}$$

where e_j is a vector of dimension $|\mathcal{J}|$ with all the coordinates equal to 0 but the jth coordinate equal to 1. We consider the following greedy assignment.

Algorithm 1 Centralized GREEDY Algorithm for Allocation of Helped Channels.

$H = 0$
while 1 **do**
  Find $I \in \mathcal{J}$ such that $\vec{H} + e_I \in P(v)$
  and $\Delta_I V(\vec{s}+\vec{H}/N) \geq \Delta_j V(\vec{s}+\vec{H}/N)$ for all $j \in \mathcal{J}$
  such that $\vec{H} + e_j \in P(v)$

  if there exists no such I **then** break
  end if
  $H_I \leftarrow H_I + 1$
end while

Theorem IV.2. *Under assumption that $V_j(x)$ is a concave function of x for each $j \in \mathcal{J}$, the solution of* SYSTEM *is obtained by* GREEDY.

Proof is available in [7]; it follows from Theorem IV.1 and showing that SYSTEM-H is a maximization of a concave function over a submodular polyhedron, so by a general result [6], GREEDY provides a solution.

B. Particular Channel Selection Strategies

In this section, we introduce three particular channel selection strategies. Under the assumption of random mixing, the first two strategies correspond to uniform and most solicited strategies in [1]. The third strategy is new and arises from the Metropolis sampling in Sec. VI.

1) Uniform: Under the uniform channel selection, each user u picks a subset of $C_u - |S(u)|$ channels by sampling uniformly at random without replacement from the set of channels that user u is not subscribed to, i.e. from the set of channels $\mathcal{J} \setminus S(u)$.

The uniform channel selection biases to forwarding less popular channels. This is quite intuitive as by the channel selection process the users select channels to which they are not subscribed to. The interested reader is referred to [7] where more discussion is provided along with making a connection to an underlying system welfare problem of the uniform channel selection.

2) Top Popular: Under this scheme, each user u picks channels from the set of channels $\mathcal{J} \setminus S(u)$ without replacement in decreasing order of the channel subscription popularity and random tie break until $C_u - |S(u)|$ channels are picked or there are no channels left. This is a greedy scheme that favours popular channels. We consider this scheme in numerical evaluations in Sec. V.

3) Pick from Neighbour: We consider channel selection strategies under which each user u upon encountering another user u' picks a candidate channel from the user u' and then based on some decision process decides whether to replace a channel to which user u currently helps with the candidate channel. The decision process is assumed to be local, independent of the current assignment of users to channels, which makes these strategies of quite practical interest.

We will construct one such a scheme, in Sec. VI, based on the Metropolis-Hastings sampling. We will see that such a scheme is associated with a system welfare problem with the following objective function: $V_{PFN}(\vec{f}) = \sum_{j \in \mathcal{J}} V_j^{PFN}(f_j)$ with

$$V_j^{PFN}(f_j) = (\alpha_j + C) f_j + D f_j \ln f_j \tag{9}$$

where C and D are system constants and $\alpha_j \geq 0$ is a constant for channel j, which expresses its relative importance (the higher the α_j, the more important the channel j).

The function $V_j^{PFN}(f_j)$ in Eq. (9) is a monotonic nondecreasing function of f_j. Note, however, that $V_j^{PFN}(f_j)$ is a *convex* function of f_j. It is thus not concave and hence does not validate the condition discussed in Sec. III-C, which ensures optimality of the greedy assignment in Sec. IV-A.

V. System Optimum vs. Heuristics

In this section, we demonstrate: A system optimal assignment of channels can yield significantly larger system welfare than some heuristics suggested by prior work. In particular, we compare with the Uniform and Top Popular assignments defined in the preceding section.

We use the subscription assignments of users to channels that we derive from the subscriptions of the users of Zune to podcasts. This dataset consists of 8,000+ distinct podcast feeds and more than a million of users. The data provides us with complete information about subscriptions of users to podcasts. For our evaluations we use the information about user subscriptions to channels. The distribution of the number of subscriptions per user is skewed with the median value of 3 and the mean value of about twice that value [5].

We consider the user-centric system welfare with the channel utility functions $V_j(f_j) = -t_j(f_j)$ where $t_j(f_j)$ is the dissemination time given by Eq. (4). For each user u, we set $C_u = |S(u)| + C$ where $|S(u)|$ is specified by the input data and C is a parameter. We compute optimum assignment by using the algorithm GREEDY (Sec. IV-A). Uniform and Top Popular assignments are computed as prescribed by their respective definitions.

In Fig. 2, we show the dissemination time per subscription versus the per node capacity C. The rate of the access to the infrastructure is fixed to 1 access per day by each user. The rate at which each user encounters other users is fixed to 100 users per day. If the dissemination is solely by direct access to the infrastructure, then the dissemination time is about 13.5 hours. We note that the dissemination time under the system optimum assignment can be reduced for the order of several hours if the dissemination is augmented with the peer-to-peer

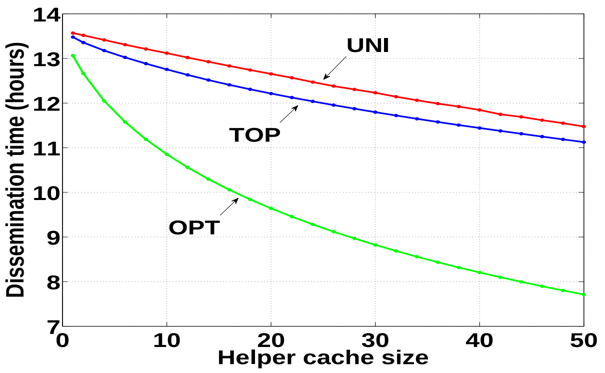

Fig. 2. Dissemination time per subscription versus the size of the public cache C, $C_u = |S(u)| + C$.

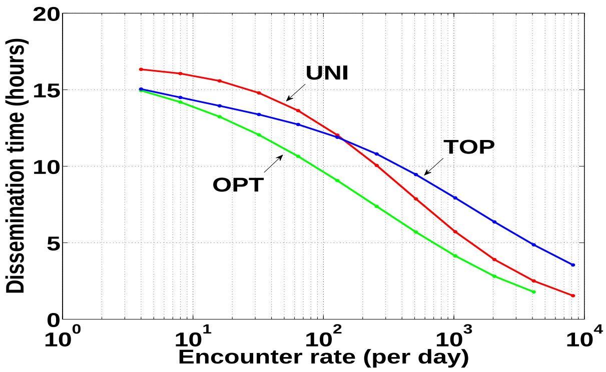

Fig. 3. Dissemination time per subscription versus the rate of encounters η. The cache size for user u set as $C_u = |S(u)| + C$ with $C = 20$.

dissemination. Perhaps even more interestingly, we observe that the gap between the system optimum and that of the Uniform and Top Popular assignments can be significant.

In Fig. 3 we present the results under the same setting as in Fig. 2 but varying the encounter rate and keeping the cache size C fixed to 20. These results show a lack of order for the Uniform and Top Popular assignments – for some cases one is better than the other one and vice-versa in other cases. In any case, system optimum indeed provides best performance.

VI. A Distributed Metropolis Hastings Algorithm

We now consider the problem of designing a distributed algorithm. The goal is for each node to control its set of helped channels so that the resulting global configuration x maximizes a system welfare of the form

$$V(x) = \sum_{j \in \mathcal{J}} w_j V_j(f_j(x)) \tag{10}$$

as discussed in Sec. III (note that, unlike in Sec. III, we make the dependence on the global configuration x explicit).

A. Metropolis-Hastings

We propose to use a Metropolis-Hastings algorithm [11], as it lends itself well to distributed optimization, and was successfully used in distributed control problems in wireless networks [12]. Before giving our distributed algorithm, we first give a short description of a centralized version of the Metropolis-Hastings algorithm:

At every time step, the algorithm picks a tentative configuration x', with probability $Q(x, x')$, where x is the current configuration. We assume that the matrix $Q(x, x')$ has the weak symmetry property:

$$Q(x, x') > 0 \Rightarrow Q(x', x) > 0$$

for all $x \neq x'$. The tentative configuration is accepted (i.e. becomes the new configuration) with probability $p = \min(1, q)$ with

$$q = \frac{\pi(x')Q(x', x)}{\pi(x)Q(x, x')} \tag{11}$$

where $\pi(\cdot)$ is a probability distribution on the set of possible configurations. The algorithm does not converge stricto sensu, however, after a large number of iterations, the probability distribution of the configuration x converges to the a priori distribution $\pi(\cdot)$. Typically, one uses for $\pi(\cdot)$ a Gibbs distribution, given by

$$\pi(x) = \frac{1}{Z} e^{\frac{V(x)}{T}} \tag{12}$$

where T is a system parameter (the "temperature") and Z is the normalizing constant. If T is small, the distribution $\pi(\cdot)$ is very much concentrated on the large values of $V(x)$, so that the algorithm produces random configurations that tend to maximize $V(x)$.

B. A Distributed Rewiring Algorithm

We use Metropolis-Hastings as follows. We use a Gibbs distribution, as in Eq.(12) with $V(\cdot)$ the utility function in Eq. (10). We consider every meeting between two nodes as one step of the algorithm. When two nodes meet, they opportunistically exchange content updates; then one of them, say u is selected as leader and attempts to replace one of its helped channels by one of the channels forwarded from the set held by the other node, say v, as described in Algorithm 2. We now turn to the computation of the acceptance probability

Algorithm 2 Distributed Algorithm for Allocation of Helped Channels

1: **if** $F(u, x) \subset F(v, x)$ **then** do nothing
2: **else**
3: u selects one channel j uniformly at random in the set $H(u, x)$
4: u selects one channel j' uniformly at random in the set $F(v, x) \setminus F(u, x)$
5: compute the acceptance probability $p = \min(1, q)$ with q given by Eq.(15)
6: draw a random number U uniformly in $[0, 1]$;
7: **if** $U < p$ **then** drop channel j and adopt channel j' as a helped channel
8: **end if**
9: **end if**

(line 5 of the algorithm), as given by Eq.(11). First we compute $Q(x, x')$ where $x' = x - 1^{u,j} + 1^{u,j'}$ is the new configuration

($1^{u,j}$ is the configuration vector defined by $1^{u,j}_{u',j'} = 1$ if $u = u'$ and $j = j'$, 0 otherwise):

Proposition VI.1. *The following holds*

$$\frac{Q(x',x)}{Q(x,x')} = \frac{\sum_{v\neq u} \frac{1_{j\in F(v,x)}}{|F(v,x)\setminus F(u,x)|+1_{j'\notin F(v,x)}}}{\sum_{v\neq u} \frac{1_{j'\in F(v,x)}}{|F(v,x)\setminus F(u,x)|}}. \quad (13)$$

Proof is in Appendix [7] D.

$$\frac{Q(x',x)}{Q(x,x')} \approx \frac{f_j(x)}{f'_j(x)}. \quad (14)$$

We note the following result:

Proposition VI.2. *Suppose that for a finite constant $D > 0$, $\lim_{N\to+\infty} NT = D$. Then,*

$$\lim_{N\to+\infty} \frac{V(x') - V(x)}{T} = \frac{1}{D}\left(w_{j'}V'_{j'}(f_{j'}(x)) - w_j V'_j(f_j(x))\right).$$

Proof is available in Appendix [7] F. In view of the last proposition, we have

$$\begin{aligned} q &= \frac{Q(x',x)}{Q(x,x')} e^{\frac{1}{T}\left(V(x')-V(x)\right)} \\ &\approx \frac{Q(x',x)}{Q(x,x')} e^{\frac{1}{NT}\left(w_{j'}V'_{j'}(f_{j'}(x))-w_jV'_j(f_j(x))\right)}. \end{aligned}$$

Combining with (14) we obtain for q the value

$$q = \frac{f_j(x)}{f_{j'}(x)} e^{\frac{1}{D}\left(w_{j'}V'_{j'}(f_{j'}(x))-w_jV'_j(f_j(x))\right)} \quad (15)$$

where $D = NT$ is a global system parameter.

Algorithm 2 requires node u to estimate f_j and $f_{j'}$. This can be done by having the nodes exchange, when they meet, updates of channel popularity for all channels that they know of, and then performing exponential smoothing. A simple, but memory hungry scheme, is as follows. Every node u maintains for every channel j an estimate $\hat{f}_j$. When node u meets node u', for all channels that u' helps or subscribes to, node u does $\hat{f}_j \leftarrow a + (1-a)\hat{f}_j$ and for all other channels $\hat{f}_j \leftarrow (1-a)\hat{f}_j$ where $0 < a < 1$.

Furthermore, all nodes need to share the global system variable D, and know the utility function of each channel (the latter can be contained as meta-information in the channel data). In Section VI-C, we give a simplified algorithm, which does not require such estimations.

C. A Simplified Algorithm

It is possible to entirely avoid the estimation of the f_j quantities, albeit at the expense of imposing a family of utility functions. The idea is to pick a set of utility functions $V_j(\cdot)$ such that f_j and $f_{j'}$ cancel out in Eq.(15). This results in a scheme that belongs to the class of schemes pick from neighbour that was introduced in Section IV-B3.

Theorem VI.1. *If for each channel j, the utility function is $V_j^{PFN}(\cdot)$ in Eq.(9) then q in Eq.(15) is given by*

$$q = \frac{\beta_{j'}}{\beta_j} \quad (16)$$

with $\beta_j = e^{\frac{\alpha_j}{D}}$ and $\beta_{j'} = e^{\frac{\alpha_{j'}}{D}}$. In particular, q is thus independent of $f_j(x)$, $f_{j'}(x)$ and more generally of the configuration x.

Proof: Follows from Eq. (9) and Eq. (15). ∎

With this simplified algorithm, nodes need to know the static parameters $\beta_j > 0$ associated with each channel. There is no global constant, nor it is necessary to evaluate $f_j(x)$. Higher values of j mean that we give more value to disseminating channel j more quickly. Note that only the relative values of β_j matter, as Eq.(16) uses only ratios, and β_j can thus be interpreted as the priority level for channel j. The resulting algorithm is the same as Algorithm 2 but with the acceptance probability q computed using Eq.(16) instead of Eq.(15).

D. A Battery Saving Algorithm

The previous algorithm may be improved to account for battery saving. The motivation is that a node may be reluctant to help disseminate channels if its battery level is low. We address this issue as follows. Assume that every node u knows its battery level $b_u \geq 0$. The battery is empty when $b_u = 0$. Assume to simplify that all nodes measure b_u in the same scale, for example, number of remaining hours of operation at full activity. We can replace the global utility in Eq.(10) by

$$\sum_{j\in\mathcal{J}} w_j V_j(f_j) - \sum_{u\in\mathcal{U}} W_u(b_u)$$

where $W_u()$ is a convex, decreasing function of its argument (for example $W_u(b) = \frac{1}{b^m}$), such that $W_u(b)$ expresses the penalty perceived by user u when its battery level is b. We can apply the Metropolis-Hastings algorithm with this new function. The only difference is in the computation of the acceptance probability. The computation of q in Eq.(16) is replaced by

$$q = \frac{\beta_{j'}}{\beta_j} e^{-[h_u(b_u)-h_{u'}(b_{u'})]} \quad (17)$$

where u and u' are the two nodes involved in the interaction and $h_u(b) > 0$ is the marginal cost of exchanging a channel when two nodes meet, divided by the temperature T (an increasing function of b).

The resulting algorithm is the same as Algorithm 2 with Eq.(15) on line 5 replaced by Eq.(17). The required configuration is (1) every channel j has a static priority level $\beta_j > 0$ and (2) every node u knows its own function $h_u(b)$ for the cost of exchanging one channel with a neighbour when this node's battery level is b.

VII. Simulation Results

In this section, we present simulation results that address the following goals: (i) demonstrate the concentration of the distributed Metropolis-Hastings algorithm to the optimum system welfare and (ii) demonstrate that optimizing a system

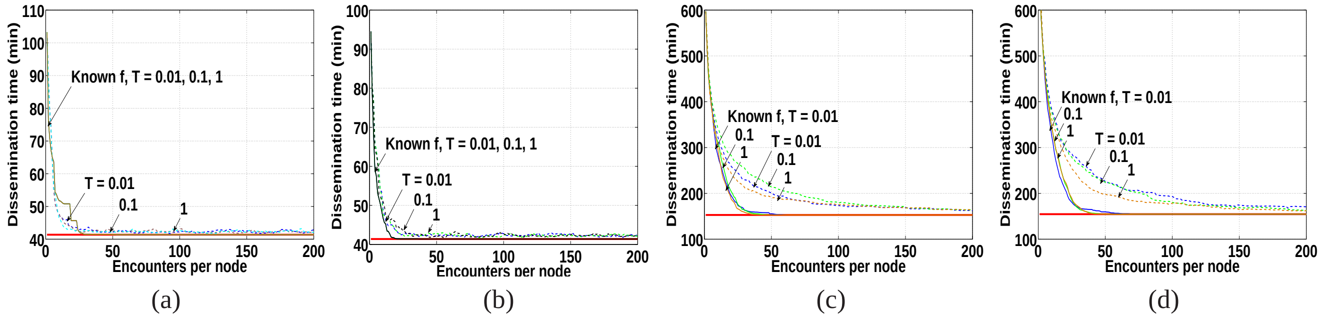

Fig. 4. Convergence of the Metropolis-Hastings algorithm under channel-centric system welfare: (a) small-scale, Zipf-2/3, (b) small-scale, Zipf-1, (c) large-scale, Zipf-2/3, (d) large-scale, Zipf-1. Small-scale refers to $(N, J) = (20, 20)$ and the large-scale refers to $(N, J) = (200, 100)$. The y-axis is the mean dissemination time over all channels. The thick horizontal line denotes the system optimum mean dissemination time. Other solid curves are for the Metropolis-Hastings algorithm with the portions of nodes that forward any given channel known ($\vec{f}$). The dashed lines denote the same but with $\vec{f}$ locally estimated.

welfare under real-world mobility produces good forwarding assignments of channels to users.

In order to cover a broad set of parameters, we conducted simulations by varying the parameters along the following dimensions: (i) node mobility either random mixing or using a real mobility trace, (ii) small and large system scale with respect to the number of users and the number of channels, (iii) different distributions for the subscriptions per channel, (iv) the fractions of nodes forwarding or subscribed to a channel either known or locally estimated, and (v) a range of the temperatures for the Metropolis-Hastings algorithm. We consider random mixing mobility in order to provide results for scenarios for which we have a good understanding of the relation between the channel dissemination time and the fraction of the forwarding nodes. We used our own discrete-event simulator.

A. Random Mixing Mobility

We simulate random mixing mobility where each user encounters other users uniformly at random. In such a system, we indeed have that the dissemination time for any channel depends only on the portion of the nodes that forward the channel (Section III-A).

We consider a small- and a large-scale system where for the former the number of users and the number of channels are both set to 20 while for the latter the number of users is 200 and the number of channels is 100. For the fractions of subscribers per channel ($\vec{s}$), we assume a Zipf distribution with the scale parameter equal to either $2/3$ or 1. For the objective of the system welfare, we consider both the channel- and user-centric cases with the utility function $V_j(f_j) = -t_j(f_j)$ for channel j where $t_j(f_j)$ is the dissemination time and f_j is the fraction of forwarding nodes. In particular, we admit Eq. (4). In cases when $\vec{f}$ or $\vec{s}$ are locally estimated, each node uses an exponential weighted averaging with the smoothing constant (weight of a sample) set as follows. For the estimation of $\vec{f}$, the constant is set to 0.9. For the estimation of $\vec{s}$, the constant is equal to 0.1 and 0.02 for the channel- and user-centric case, respectively.

In Fig. 4, we present the results obtained for the channel-centric case. The graphs show the mean dissemination time

TABLE I
PER-CHANNEL AND PER-USER DISSEMINATION TIMES IN MINUTES FOR CAM TRACE.

Channel-centric	UNI	TOP	OPT
Median	70.2500	133.1000	52.1429
Mean	70.4700	137.1250	57.2000
User-centric	UNI	TOP	OPT
Median	70.4028	97.4528	56.9333
Mean	70.0578	102.7284	59.4089

per channel, i.e. $(\sum_{j \in \mathcal{J}} t_j(f_j))/J$, versus the number of encounters per node. We show the results for the Metropolis-Hastings with $\hat{f}$ assumed to be either known or locally estimated by individual nodes. We observe that the system welfare under the Metropolis-Hastings algorithm concentrates near the optimum system welfare. The results in Fig. 4 indicate a faster concentration in cases when $\vec{f}$ is globally known. We obtained qualitatively same results for the user-centric case which we omit for space reasons; the reader is referred to Fig. 9 in [7]. In summary, our results support the following claim: The system welfare under the Metropolis-Hastings algorithm concentrates near the optimum system welfare with $\vec{f}$ (and $\vec{s}$ in the user-centric case) either globally known or locally estimated.

B. Real Trace Mobility

We compare the system performance under the assignment of channels to users that optimizes a system welfare (OPT) with that of heuristics Uniform (UNI) and Top Popular (TOP), respectively introduced in Sec. IV-B1 and Sec. IV-B2. Our goal is to demonstrate that OPT can do a better job compared to the heuristics UNI and TOP.

We define the system welfare using the dissemination function $t_j(f_j)$ inferred from the mobility trace CAM and letting $V_j(f_j) = -t_j(f_j)$ as in the preceding section. Specifically, we define the logarithm of $t_j(f_j)$ by a concatenation of linear segments that closely follow the empirical data (available in Fig. 7 [7]). We consider a scenario with $J = 40$ channels, ten subscriptions per each user, and ten channels helped by each user. We assume that the channel subscription rates follow a Zipf distribution with the scale parameter equal to $2/3$. For each setting of the simulation parameters, we repeat the experiment five times, each time injecting a message of a

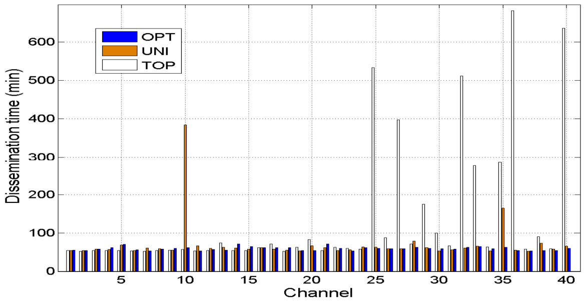

Fig. 5. Mean channel dissemination time under CAM mobility with channel-centric system welfare (similar results hold for user-centric case; see [7]). Channels are enumerated in decreasing popularity (channel 1 is most popular, etc).

channel to a user picked uniformly at random from the users who are either subscribers or helpers for the channel at the beginning of the trace. Note that there are 36 distinct users in the CAM data and that the encounter rate η is equal to 0.001 per second, i.e. 1.2 users every two minutes.

In Table 1 we present the median and mean dissemination time per channel, and per user, for the channel- and user-centric cases, respectively. For both mean and median dissemination time, OPT substantially outperforms UNI and TOP for either channel-centric or user-centric case. In particular, in the channel-centric case, OPT achieves over 70 minutes less dissemination time than TOP and over 10 minutes less dissemination time than UNI for both mean and median dissemination time. In the user-centric case,OPT achieves over 40 minutes less dissemination time than TOP and over 10 minutes less dissemination time than UNI for both mean and median dissemination time. Furthermore, in Fig. 5, we show the mean dissemination time for each channel. We note the following. First, under the channel assignment UNI, some intermediate popular channels may be penalized with a high dissemination time. In particular, in Fig. 5, we note that the tenth most popular channel gets as much as five hours larger dissemination time than under other channel assignments. Second, same can happen under TOP where the results conform to the expected bias against less popular channels. To be specific, many less popular channels get as much as several hours larger dissemination time than under other channel assignment. The results demonstrate cases where assigning channels by optimizing a system welfare avoids penalizing some channels which can occur under the heuristics such as UNI or TOP.

VIII. Related Work

In this section we discuss the work that is most closely related to our work; more discussion is available in [7]. [1] proposed several heuristics for the content exchange between devices based on the inferred preference of the user of a device and that of encountered devices. Each device is assumed to forward an unlimited number of feeds and prioritizes the download of pieces of the content feeds from the encountered devices. Feeds subscribed by a device take priority over other feeds. Each device uses a solicitation strategy to decide which pieces to fetch from the encountered devices. The considered solicitation strategies include the *most solicited* and *uniform* which essentially correspond to the top popular and uniform channel assignment considered in our paper. The approach in [1] is different from ours in that they evaluate a set of a priori defined strategies while we first formulate a global system welfare objective and then identify a channel selection strategy to optimize the given system objective. Another closely related work is CarTorrent [2], a peer-to-peer file sharing designed for vehicular network scenarios using epidemic-style content dissemination. Further related work is [13]. Our work is generally different from the state-of-the art results on epidemic-style information dissemination in that our goal is efficient dissemination by controlling the epidemic spread of *multiple* content streams.

IX. Conclusion

We proposed a framework for optimizing multi-channel information in wireless ad-hoc networks. The problem amounts to optimizing assignment of users to channels for forwarding of the content with respect to a global system welfare objective. We showed that system optimal assignments can be found by a centralized greedy algorithm. Moreover, we showed that the optimal assignment can be well approximated by a distributed algorithm based on the Metropolis-Hasting sampling. We also discussed how to optimize other resources such as the battery power. There are several interesting directions for future work including the convergence analysis of the proposed two-timescale control and extending the framework to heterogenous scenarios with respect to the node mobility.

References

[1] V. Lenders, G. Karlsson, and M. May. Wireless Ad Hoc Podcasting. In *IEEE SECON*, 2007.

[2] K. C. Lee, S. H. Lee, R. R. Cheung, U. Lee, and M. Gerla. First Experience with CarTorrent in a Real Vehicular Ad Hoc Network Testbed. *Mobile Networking for Vehicular Env.*, pages 109–114, 2007.

[3] MIT Reality Mining. Social serendipity. Downloaded from http://reality.media.mit.edu/serendipity.php, July 2009.

[4] Microsoft. Zune social. www.zune.net, December 2008.

[5] Dinan Gunawardena, Thomas Karagiannis, Alexandre Proutiere, and Milan Vojnović. Characterizing podcast services: Publishing, usage, and dissemination. Technical Report MSR-TR-2009-58, Microsoft Research, May 2009.

[6] A. Federgruen and H. Groenevelt. The greedy procedure for resource allocation problems: Necessary and sufficient conditions for optimality. *Operations Research*, 34(6):909–918, 1986.

[7] Liang Hu, Jean-Yves Le Boudec, and Milan Vojnović. Optimal channel choice for collaborative ad-hoc dissemination. Technical Report MSR-TR-2009-26, Microsoft Research, March 2009.

[8] J. Scott, R. Gass, J. Crowcroft, P Hui, C. Diot, and A. Chaintreau. CRAWDAD data set cambridge/haggle (v. 2006-09-15). Downloaded from http://crawdad.cs.dartmouth.edu/cambridge/haggle, September 2006.

[9] M. Piorkowski, N Sarafijanovic-Djukic, and M. Grossglauser. CRAWDAD data set epfl/mobility (v. 2009-02-24). Downloaded from http://crawdad.cs.dartmouth.edu/epfl/mobility, February 2009.

[10] T. Karagiannis, J.-Y. Le Boudec, and M. Vojnović. Power-Law and Exponential-Decay of Inter-Contact Time Between Mobile Devices. In *ACM MOBICOM*, 2007.

[11] P. Brémaud. *Markov Chains: Gibbs Fields, Monte Carlo Simulation and Queues*. Springer, 1999.

[12] B. Kauffmann, F. Baccelli, A. Chaintreau, V. Mhatre, K. Papagiannaki, and C. Diot. Measurement-Based Self Organization of Interfering 802.11 Wireless Access Networks. In *IEEE INFOCOM*, 2007.

[13] A. Chaintreau, J.-Y. Le Boudec, and N. Ristanović. The age of gossip: Spatial mean-field regime. In *ACM Sigmetrics*, 2009.