

Mobile Data Offloading through Opportunistic Communications and Social Participation

Bo Han^{*}, Pan Hui[†], V. S. Anil Kumar[‡], Madhav V. Marathe[‡]

Jianhua Shao[¶], Aravind Srinivasan[§]

^{*} Department of Computer Science, University of Maryland, College Park, MD 20742, USA

[†] Deutsche Telekom Laboratories, Ernst-Reuter-Platz 7, 10587 Berlin, Germany

[‡] Department of Computer Science and Virginia Bioinformatics Institute, Virginia Tech, Blacksburg, VA 24061, USA

[¶] School of Computer Science, University of Nottingham, Nottingham NG8 1BB, United Kingdom

[§] Department of Computer Science and Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742, USA

Abstract

3G networks are currently overloaded, due to the increasing popularity of various applications for smartphones. Offloading mobile data traffic through opportunistic communications is a promising solution to partially solve this problem, because there is almost no monetary cost for it. We propose to exploit opportunistic communications to facilitate information dissemination in the emerging Mobile Social Networks (MoSoNets) and thus reduce the amount of mobile data traffic. As a case study, we investigate the target-set selection problem for information delivery. In particular, we study how to select the target set with only k users, such that we can minimize the mobile data traffic over cellular networks. We propose three algorithms, called *Greedy*, *Heuristic*, and *Random*, for this problem and evaluate their performance through an extensive trace-driven simulation study. Our simulation results verify the efficiency of these algorithms for both synthetic and real-world mobility traces. For example, the *Heuristic* algorithm can offload mobile data traffic by up to 73.66% for a real-world mobility trace. Moreover, to investigate the feasibility of opportunistic communications for mobile phones, we implement a proof-of-concept prototype, called *Opp-Off*, on Nokia N900 smartphones, which utilizes their Bluetooth interface for device/service discovery and content transfer.

Index Terms

Mobile data offloading, target-set selection, opportunistic communications, mobile social networks, implementation, trace-driven simulation.

I. INTRODUCTION

Due to the proliferation of smartphones (e.g., Apple's iPhone and Nokia N95), mobile operating systems (e.g., Google's Android and Symbian OS), and online social networking services, Mobile Social Networks (MoSoNets) have begun to attract increasing attention in recent years. The development of MoSoNets has already evolved from the simple extensions of online social networking sites to powerful mobile social software and applications [17], [20], [32], [34], [36]. Currently, a large percentage of mobile data traffic is generated by these mobile social applications and mobile broadband-based PCs [2]. A side effect of the explosion of these applications, along with other mobile applications, is that 3G cellular networks are currently *overloaded*. According to AT&T's media newsroom, its network experienced a 5,000 percent surge of mobile data traffic in the past three years¹. Thus, it is imperative to develop novel architectures and protocols for this critical problem.

Mobile data offloading, also referred to as mobile cellular traffic offloading, is the use of complementary network communication technologies to deliver mobile data traffic originally planned for transmission over cellular networks. In the original delay-tolerant approach, delay is usually caused by intermittent connectivity [19]. For instance, motivated by the fact that the coverage of WLAN hotspots may be very limited, and thus mobile users may not always be able to connect to the Internet through them, Pitkänen et al. [40] explore opportunistic web access via WLAN hotspots for mobile phone users. We propose to *intentionally* delay the delivery of information over cellular networks and offload it through the free opportunistic communications, with the goal of reducing mobile data traffic.

In this paper, we study the *target-set selection* problem as the first step toward bootstrapping mobile data offloading for information delivery in MoSoNets. The information to be delivered in mobile networks may include multimedia newspapers, weather forecasts, movie trailers etc., generated by content service providers. As an example, in addition to traditional written text and photos, multimedia newspapers may contain news video clips, music, and small computer games. Benefiting from the delay-tolerant nature of non-real-time applications, service providers may deliver the information to only a small fraction of selected users (i.e., target-users), to reduce

¹<http://www.att.com/gen/press-room?pid=4800&cdvn=news&newsarticleid=30838> (AT&T Launches Pilot Wi-Fi Project in Times Square, verified in Oct. 2010)

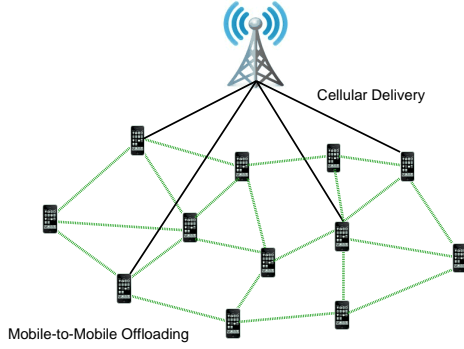


Fig. 1. A snapshot of the contact graph for a small group of subscribed mobile users.

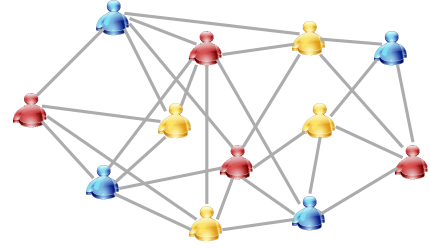


Fig. 2. The social graph of mobile users shown in Figure 1.

mobile data traffic and thus their operation cost. As shown in Figure 1, target-users can then help to further propagate the information among all the subscribed users through their social participation, when their mobile phones are within the transmission range of each other and can communicate *opportunistically*. Non-target-users can also disseminate the information after they get it from target-users or others. The major advantage of this mobile data offloading approach is that there is almost no monetary cost associated with opportunistic communications, which are realized through either WiFi or Bluetooth technology.

We investigate how to choose the initial target set with only k users, such that we can minimize the amount of mobile data traffic. We can translate this objective into maximizing the expected number of users that can receive the delivered information through opportunistic communications². The larger this number is, the less the mobile data traffic will be. To offload other downstream and upstream data (that may contain private information) through the others phones as relay, we must pay special attention to protect the users' privacy. Thus, we focus on popular information delivery in this paper.

It follows from the work of Nemhauser et al. [37] that if the information dissemination function that maps the initial target set to the expected number of infected users is *submodular* (discussed in detail in Section IV), a natural greedy algorithm can achieve a provable approximation ratio of $(1 - 1/e)$ (the best known result so far), where e is the base of the natural logarithm. Thus, if we can prove the submodularity of the information dissemination function, we will be able to apply the greedy algorithm to our target-set selection problem. Our first contribution is that

²We call these users the *infected* users, similar to the infected individuals in the Susceptible-Infected-Recovered (SIR) epidemic model for the transmission of communicable disease through individuals.

by extending the result of Kempe, Kleinberg, and Tardos [29] we prove that the information dissemination function is submodular for the contact graph of mobile users, which changes dynamically over time. However, although this greedy algorithm achieves the best known result, it requires the knowledge of user mobility in the future, which may not be practical.

Our second contribution is to exploit the *regularity* of human mobility [23], [33] and apply the target set identified using mobility history to future information delivery. For example, we determine the target set using the greedy algorithm based on today’s user mobility history of a given period, and then use it as the target set for tomorrow’s information delivery during the same period. We show through an extensive trace-driven simulation study that this heuristic algorithm always outperforms the simple random selection algorithm (wherein the k target users are chosen randomly), and can offload up to 73.66% of mobile data traffic for a real-world mobility trace. The simulation results also indicate that social participation is a key enabling factor for opportunistic-communication based mobile data offloading.

The third contribution of this paper is that we evaluate the feasibility of opportunistic communications for moving phones by building a prototype, called `Opp-Off`, as the first step in the implementation of the proposed information dissemination framework. We compare the energy-consumption performance of both Bluetooth and WiFi interfaces for device discovery and choose Bluetooth as the candidate technology for `Opp-Off`. We also evaluate the device discovery probability, the amount of transferred data, and the duration of data transfer between a static phone and a moving phone. The experimental results show that these two phones can exchange up to 1.48 MB data during their short contacts.

This paper is organized as follows. We briefly review related work in Section II. We then present the system model and formulate the problem in Section III. In Section IV, we propose three algorithms for the target-set selection problem. We investigate the feasibility of opportunistic communications for mobile phones through a prototype implementation in Section V. In Section VI, we evaluate the performance of the three proposed algorithms through trace-driven simulations. We discuss several practical issues in Section VII, and then conclude.

II. RELATED WORK

In this section, we review the existing related work in three categories: cellular traffic offloading, information diffusion/dissemination and mobile social networks.

A. Cellular Traffic Offloading

There are two types of existing solutions to alleviate the traffic load on cellular networks: offloading to femtocells and WiFi networks.

1) *Femtocell for Indoor Environments*: Originally, the femtocell technology (i.e., access point base stations) was proposed to offer better indoor voice and data services of cellular networks. Femtocells work on the same licensed spectrum as the macrocells of cellular networks and thus do not require special hardware support on mobile phones. But customers may need to install short-range base stations in residential or small-business environment, for which they will provide Internet connections. Due to their small cell size, femtocells can lower transmission power and achieve higher signal-to-interference-plus-noise ratio (SINR), thus reducing the energy consumption of mobile phones. Cellular operators can reduce the traffic on their core networks when indoor users switch from macrocells to femtocells. A literature review about the technical details and challenges of femtocells can be found in Chandrasekhar et al. [13].

2) *Cellular Traffic Offloading to WiFi Networks*: Compared to femtocells, WiFi networks work on the unlicensed frequency bands and thus cause no interference with 3G cellular networks. As a result, cellular network operators, including AT&T, T-Mobile, Vodafone, and Orange, have deployed or acquired WiFi networks worldwide [1]. Meanwhile, there are already several offloading solutions and applications proposed from the industry. For instance, the Line2 iPhone application (available at <http://www.line2.com/>) clones the phone's own software and can initiate voice calls over WiFi networks. iPassConnect³ enables end users to switch between 3G cellular and WiFi connections smoothly, and provides them a one-touch login for more than 100,000 hotspots operated by 100+ providers. Recently, Balasubramanian et al. [4] have proposed a scheme called Wiffler to augment mobile 3G networks using WiFi for delay-tolerant applications.

Offloading cellular traffic to femtocells and WiFi networks is limited by their network deployment and relies on the availability of Internet access. Instead, we offload mobile data traffic through opportunistic communications for information dissemination, in metropolitan areas.

B. Information Diffusion/Dissemination

Social networks can be thought of as the carrier of information flows in communities. Various wireless communication technologies can effectively help the propagation of information

³<http://www.ipass.com/>

among mobile users. As a result, information diffusion/dissemination has been widely studied in traditional social networks [15], [29], [41] and wireless networks [31], [36], [38].

1) *Traditional Social Networks*: Information diffusion has been extensively investigated through viral marketing [41] and social networks [15], [29]. Domingos and Richardson [15], [41] introduce a fundamental algorithmic problem of information diffusion: what is the initial subset of k users we should target, if we want to propagate the information to the largest fraction of the network? Kempe et al. [29] prove that for the influence maximization problem in social networks, the information dissemination function is submodular for several classes of models. They also leverage the co-authorship graph from arXiv in physics publications to demonstrate that the proposed algorithm outperforms heuristics based on node centrality and distance centrality, which are well-known metrics in social networks. Although our proof of the submodularity of information dissemination function for the target-set selection problem investigated in this paper is an extension of the result in Kempe et al. [29], we enhance the independent cascade model to make it more realistic to study the information dissemination process in mobile social networks (discussed in details in Section IV-A). Gruhl et al. [24] study the dynamic of information diffusion in blogspace. They characterize information propagation along two dimensions: macroscopic diffusion of topics, based on long-term changes of primary focus and short-term behavior of fixed topic; and microscopic diffusion between individuals, using the theory of disease spreading. In this paper, we exploit social participation and interaction to offload mobile data traffic.

2) *Opportunistic Networks*: There are also several existing works for information dissemination in wireless networks. 7DS [38] is a peer-to-peer data dissemination and sharing system for mobile devices, aiming at increasing the data availability for users who have intermittent connectivity. Due to the heterogeneity of access methods and the spatial locality of information, when mobile devices fail to access Internet through their own connections, they can try to query data from peers in their proximity, who either have the data cached, or have Internet access and thus can download and forward the data to them. Lindemann and Waldhorst [31] model the epidemic-like information dissemination in mobile ad hoc networks, using four variants of 7DS [38] as examples. They consider the spread of multiple data items by devices with limited buffers and use the least recently used (LRU) approach as their buffer management scheme.

Similar to our work, Vukadinović and Karlsson [44] propose to utilize mobility-assisted wireless podcasting to offload the cellular operator's network. However, aiming to minimize the

spectrum usage in cellular networks, they simply select $p\%$ of the subscribers with the strongest propagation channels as target users which may include inactive users. Ioannidis et al. [28] study the dissemination of content updates in MoSoNets, investigating how service providers can optimally allocate bandwidth to keep the content updated as early as possible and how the average age of content changes when the number of users increases. Compared to the above works, we focus on the target-set selection problem to reduce mobile data traffic.

3) *Other Wireless Networks*: Diffusion has also been widely studied in wireless sensor networks and cellular networks. Directed diffusion [27] is a data-centric dissemination paradigm for sensor networks, in the sense that the communication is for named data (attribute-value pairs). It achieves energy efficiency by choosing empirically good paths, and by caching data and processing it in-network. The parametric probabilistic sensor network routing protocol [6] is a family of multi-path and light-weight routing protocols for sensor networks. It determines the forwarding probability of intermediate sensors based on various parameters, including the distance between these sensors, and the number of traveled hops of a message. Zhu et al. [49] propose solutions to prevent the spread of worms in cellular networks by patching only a small number of phones. They construct a social relationship graph of mobile users where the weights of edges are determined by the amount of traffic between two mobile phones and use this graph to represent the most likely spreading path of worms. After partitioning the graph, they can select the optimal set of phones to separate these partitions and block the spreading of worms.

C. Mobile Social Networks

A recent trend for online social networking services, such as Facebook, is to turn mobile. Meanwhile, native MoSoNets have been created, for example, Foursquare and Loopt. Motivated by the fact that people are usually good resources for location, community, and time-specific information, PeopleNet [36] is designed as a wireless virtual social network that mimics how people seek information in real life. In PeopleNet, queries of a specific type are first propagated through infrastructure networks to bazaars (i.e., geographic locations of users that are related to the query). In a bazaar, these queries are further disseminated through peer-to-peer communications, to find the possible answers. WhozThat [7] is a system that combines online social networks and mobile smartphones to build a local wireless networking infrastructure. It utilizes wireless connections to online social networks to bind social networking IDs with location.

WhozThat also provides an entire ecosystem to build complex context-aware applications.

Micro-Blog [20] is a social participatory sensing application that can enable the sharing and querying of content through mobile phones. In Micro-Blog, mobile phones periodically send their location information to remote servers. When queries, for example, about parking facilities around a beach, cannot be satisfied by the current content available on the server, they will be directed to users in the specific geographic area who may be able to answer these queries. CenceMe [34] is a people-centric sensing application that infers individual's sensing presence through off-the-shelf sensor-enabled mobile phones and then shares this information using social network portals such as Facebook and MySpace. In this paper, we study how social participation can help to disseminate information among mobile users.

III. SYSTEM MODEL AND PROBLEM STATEMENT

In this section, we describe the system model of MoSoNets and the target-set selection problem we propose to solve.

A. Model of MoSoNets

No matter which online social networking service we are using now, we are going to see only a piece of our actual social network. However, MoSoNets can integrate not only friends from all the major social networking sites, but also work colleagues and family members who are hidden from these online services. Moreover, MoSoNets can also provide a platform to signal face-to-face interactions among nearby people who probably should know each other [17]. There are two kinds of typical connections in MoSoNets, similar to the small-world networks [46]:

- Local connections realized by short-range communications, through WiFi or Bluetooth networks. When two mobile phones are within the transmission range of each other, their owners may start to exchange information, although they may not be familiar with each other. This opportunistic communication heavily depends on the mobility pattern of users and usually we can construct *contact graphs* (as shown in Figure 1, as a snapshot) for them. Their major advantage is that they do not require infrastructure support and there is no monetary cost.
- Remote connections realized by long-range communications, through cellular networks (e.g., EDGE, EVDO, or HSPA). This communication happens only between friends in real life. It

may be used sporadically, compared to the short-range communications. Usually users need to pay for such data transmissions. We can construct a *social graph*, as shown in Figure 2, based on the social relationship of mobile users. Users connected by an edge are friends of each other. There are three communities depicted by different colors. Users in the same community form a clique. There are also connections between different communities. The friend relationship within a community is not shown here for clarity.

The study of traditional social networks focuses on social graph, and contact graph has been extensively investigated for opportunistic communications. We advocate that *MoSoNets can be viewed as a marriage of traditional social networks with emerging opportunistic networks*. We can exploit both types of communication to facilitate information dissemination in MoSoNets. On one hand, friends can actively forward (push) information whenever they want. On the other hand, mobile users that are in contact can also pull information from each other locally. We note that Chierichetti et al. [14] recently study a similar push-pull strategy for rumour spreading. Meanwhile, Burleigh proposes Contact Graph Routing [11] for delay-tolerant networks, where connectivity changes are scheduled and planned, rather than discovered or predicted.

B. Problem Statement

As we mentioned in Section I, we aim to study how to choose the initial target set with only k users, such that we can maximize the expected number of infected users. This number will translate into the decrease of mobile data traffic. If there are totally n subscribed users and m users finally receive the information before the deadline, the amount of reduced mobile data traffic will be $n - (k + (n - m)) = m - k$. For a given mobile user, delivery delay is defined to be the time between when a service provider delivers the information to the k users until a copy of it is received by that user. Service providers will send the information to a user directly through cellular networks, if he or she fails to receive the information before the delivery deadline.

How the information is propagated is determined by the behavior of mobile users, and we exploit a probabilistic dissemination model in this paper. We define the *pull probability* to be the probability that mobile users pull the information from their peers during one of their contacts. The value of pull probability p may not be the same for different types of information and might change as time goes on, which reflects the dynamics of information popularity. After mobile users receive the information from either the service providers or their peers, they may also forward

it, through cellular networks (e.g., MMS, Multimedia Messaging Service), to their friends with probability q . Usually, $p > q$, because users may prefer the free opportunistic communications. Moreover, short-range communications consume much less energy, in terms of data transmission, than long-range ones. For example, it was reported in a measurement study that to download 10 KB data, WiFi consumes one-sixth of 3G's energy and one-third of GSM's energy [5]. We do not consider the push-based approach for opportunistic communications in this paper and leave it as a future work.

The modeling of information dissemination through opportunistic communications can be viewed as a combination of three sub-processes. First, to protect their privacy, mobile users have the control of whether or not to share a piece of information with their geographical neighbors and share it with probability p_1 . Second, mobile users may want to explore the information in their proximity only when they are not busy and mobile phones may not always be able to discover each other during their short contacts. Thus they can find the meta-data of a piece of information with probability p_2 . Finally, based on these meta-data, mobile users will decide whether or not to fetch the information from their geographical neighbors and pull it with probability p_3 . As a result, $p = p_1 \cdot p_2 \cdot p_3$.

IV. TARGET-SET SELECTION

We first prove the submodularity of the information dissemination function for the contact graph of mobile users, which leads to the greedy algorithm. The information dissemination function is the function that maps the target set to the expected number of infected users of the information dissemination process. Then we present the details of the greedy algorithm and the proposed heuristic algorithm.

A. Submodularity of the Information Dissemination Function

If we can prove that the information dissemination function is submodular, we can then apply the well-known greedy algorithm proposed by Nemhauser et al. [37] to identify the target set. For any subset S of the users, the information dissemination function $g(S)$ gives the final number of infected users when S is the initial target set. The function $g(\cdot)$ is submodular if it satisfies the *diminishing returns* rule. That is, the marginal gain of adding a user, say u , into the target

set S is greater than or equal to that of adding the same user into a superset S' of S :

$$g(S \cup \{u\}) - g(S) \geq g(S' \cup \{u\}) - g(S'),$$

for all users u and all pairs of sets $S \subseteq S'$. We prove the submodularity of the information dissemination function by extending the approach developed in Kempe et al. [29].

Our proof of the submodularity differs from that in Kempe et al. [29] in two ways. First, Kempe et al. [29] prove that the information diffusion function is submodular for the independent cascade model [22] of influence maximization. In that model, when a node u becomes active, it has a *single* chance to activate any currently inactive neighbor v with probability $p_{u,v}$. In comparison, in our extended independent cascade model, mobile users have the chance to pull/exchange information for *every* contact. There are also several other diffusion models in the literature [43] and some of them were derived from another basic model, the linear threshold model [29]. Our enhanced independent cascade model is more realistic than these previous models, as it can account for multiple contacts among mobile users.

Second, as we mentioned in Section III, compared to the information diffusion in traditional social networks [29], the contact graph of MoSoNets changes dynamically and mobile users can pull information from their peers at every contact. To solve this problem, we generate a *time-stamped* contact graph, which is also called time-expanded graph in the literature, e.g., in Hoppe and Tardos [25]. Note that the delay-tolerance threshold (i.e., the delivery deadline) determines the information dissemination duration (from when service providers deliver information to target users to the delivery deadline). As a result, only edges whose corresponding contacts occur before the threshold will be included in this time-stamped graph.

Generally it is hard to compute exactly the underlying information dissemination function $g(\cdot)$ and obtain a closed form expression of it. However, as in Kempe et al. [29], we can estimate the value of $g(\cdot)$ by Monte Carlo sampling. For each pair of users u and v , if they are in contact ℓ times during the information dissemination process, there will be ℓ time-stamped edges in the graph, one for each contact. Suppose u 's pull probability for v during a given contact t is $p_{u,v,t}$.⁴ We can view this random event as flipping a coin of bias $p_{u,v,t}$. Note that whether we flip the coin at the beginning of information dissemination or when u and v are in contact t will not

⁴We can define the pull probability $p_{v,u,t}$ accordingly.

affect the final results. Thus, we can assume that for every contact t of each pair of users u and v , we flip a coin of bias $p_{u,v,t}$ at the beginning of the process and save the result to check later.

After we get all the results of coin flips, we mark the edges with successful pulling of information as *active* and the remaining edges as inactive. Since we already know the results of the coin flips (i.e., whether a mobile user can infect his/her peers for a given contact) and the initial target set, we can calculate the number of infected users at the end of the information dissemination process. In fact, one possible set of results of the coin flips stands for a sample point in the probability space. Suppose z is a sample point and define $g_z(S)$ to be the number of infected users when S is the initial target set. Then $g_z(S)$ is a deterministic quantity for a fixed contact trace. Further define $I(u, z)$ to be the set of users that have a path from u , for which all the edges on it are active and their time-stamps satisfy the *monotonically increasing* requirement⁵. We have

$$g_z(S) = \cup_{u \in S} I(u, z).$$

We now prove that the function $g_z(S)$ is submodular for a given z . Consider two sets S and S' , $S \subseteq S'$. $g_z(S \cup \{u\}) - g_z(S)$ is the number of users in $I(u, z)$ that are not in $\cup_{v \in S} I(v, z)$. Note that $\cup_{v \in S'} I(v, z)$ is at least as large as $\cup_{v \in S} I(v, z)$. We then have

$$g_z(S \cup \{u\}) - g_z(S) \geq g_z(S' \cup \{u\}) - g_z(S').$$

Since $g(S) = \sum_z \text{Prob}(z) \cdot g_z(S)$, we thus obtain that $g(\cdot)$ is submodular, because it is a non-negative linear combination of a family of submodular functions.

B. Greedy, Heuristic, and Random Algorithms

We present three algorithms for the target-set selection problem, called `Greedy`, `Heuristic`, and `Random`. For the `Greedy` algorithm, initially the target set is empty. We evaluate the information dissemination function $g(\{u\})$ for every user u , and select the most active user (i.e., the one that can infect the largest number of uninfected users) into the target set. Then we repeat this process, in each round selecting the next user from the rest with the maximum increase of $g(\cdot)$ into the target set, until we get the k users. Target-set selection is an NP-hard problem for both the independent cascade model and the linear threshold model [29]. Let S^* be the optimal

⁵This requirement reflects the temporal evolution of the information dissemination process along the paths.

target set, Nemhauser et al. [37] show that if the function $g(\cdot)$ is non-negative, monotone and submodular, and at each time we select a user that gives the maximum marginal gain of $g(\cdot)$ to get a target set S with k users, then $g(S) \geq (1 - 1/e) \cdot g(S^*)$. Thus, given that the information dissemination function satisfies the above requirements, the `Greedy` algorithm approximates the optimum solution to within a factor of $(1 - 1/e)$. However, we note that the limitation of the `Greedy` algorithm is that it requires the knowledge of user mobility during the dissemination process, which may not be available at the very beginning.

To make the `Greedy` algorithm practical, we propose to exploit the regularity of human mobility [23], [33], which leads to the `Heuristic` algorithm. Based on a six-month trace of the locations of 100,000 anonymized mobile phone users, Gonzalez et al. [23] identify that human mobility shows a very high degree of temporal and spatial regularity, and that each individual returns to a few highly frequented locations with a significant probability. Benefiting from the regularity of human mobility, the `Heuristic` algorithm identifies the target set using the history of user mobility, and then uses this set for information delivery in the future. That is, for a given period $[s, t]$ of a day d , we apply the `Greedy` algorithm to determine the target set S of the same history period $[s, t]$ of the day $d - c$ based on mobility history, where c is a small integer (usually 1 or 2), and then for information delivery of $[s, t]$ of the day d , service providers send the information to mobile users in S at the beginning to bootstrap the dissemination process. To enable the `Greedy` algorithm, the information dissemination protocol can collect the contact information of the subscribed users. At the end of a day, users can upload the information to the service providers through either their PCs or the WiFi interfaces on their phones.

Finally, for the `Random` algorithm, the service providers select k target users randomly from all the subscribed users. As we will show in Section VI, although this algorithm is simple, it is still effective in the offloading process. Before presenting the simulation results, we introduce our prototype implementation in the next section, which verifies the feasibility of mobile data offloading through opportunistic communications in practice.

V. IMPLEMENTATION

In this section, we evaluate the feasibility of opportunistic communications for moving mobile phones through a proof-of-concept prototype implementation that we built for the proposed information dissemination framework, called `Opp-Off`. In a recent work, Zahn et al. [48]

investigate the content dissemination for devices mounted on moving vehicles. However, there is very little work about whether it is feasible to support opportunistic communications on mobile phones. Since it is hard to deploy the proposed offloading solution on a large user base (e.g., more than 100 users), we focus on the feasibility of opportunistic communications between two mobile phones in this section and evaluate the performance of proposed target-set selection algorithms through trace-driven simulation for large data sets in Section VI.

A. Bluetooth or WiFi

The two common local wireless communication technologies available on most smartphones are Bluetooth and WiFi (a.k.a., IEEE 802.11). There are three major phases during opportunistic communications: device discovery, content/service discovery, and data transfer. In the following, we discuss how to support opportunistic communications using Bluetooth and WiFi separately.

The Bluetooth specification (Version 2.1) [9] defines all layers of a typical network protocol stack, from the baseband radio layer to application layer. It operates in the 2.4 GHz frequency band, shared with other devices [30] (e.g., IEEE 802.11 stations and microwave ovens). Thus, for channel access control Bluetooth uses Frequency-Hopping Spread Spectrum (FHSS) to avoid interference with coexisting devices.

For two Bluetooth devices to discover each other, one of them (inquiring device) sends out inquiry messages periodically and waits for responses; another one (scanning device) listens to the wireless channels and sends back responses after receiving inquiries [16]. The duration of a Bluetooth time slot is $625 \mu\text{s}$. During the device discovery procedure, an inquiring device uses two trains of 16 frequency bands each, selected from 79 frequency bands of 1 MHz width in the range 2402-2480 MHz. The 32 bands are selected based on a pseudo-random scheme and the device switches trains every 2.56 seconds. The inquiring device sends out two inquiry messages on two different frequency bands in every time slot and waits for responses on the same frequency bands during the next time slot. Two parameters, scan window and scan interval, control the duration and frequency of scanning devices. After the device receives an inquiry message, it will wait for $625 \mu\text{s}$ (i.e., the duration of a time slot) before sending out a response on the same frequency band, which completes the device discovery procedure. To increase the device discovery probability and reduce the discovery time, an interlaced inquiry scan mode was proposed in Bluetooth Version 1.2.

Bluetooth defines a Service Discovery Protocol (SDP) to allow devices to discover services provided by others. SDP determines the Bluetooth profiles (e.g., Headset Profile and Advanced Audio Distribution Profile) that are supported by the devices. Bluetooth uses a 128-bit Universally Unique Identifier (UUID) to identify each service. When a device installs a new service, it will register the service with its local SDP server. To discover services supported by others, a device can connect to their SDP servers and search through their service records [47].

There are two types of commonly used Bluetooth transport protocols, L2CAP (Logical Link Control & Adaptation Protocol) and RFCOMM (Radio Frequency Communications). L2CAP is built upon Asynchronous and Connection-less Link (ACL) and multiplexes data transmissions from higher-level protocols and applications. RFCOMM is on top of L2CAP in the protocol stack. It is designed to emulate RS-232 serial ports and supports services similar to TCP. The nominal data rate of Bluetooth Version 2.0 + EDR (extended data rate) is 3 Mbps and can be up to 24 Mbps for Version 3.0 + HS (high speed). Since the Bluetooth specification has more than 1400 pages [9], we refer interested readers to Smith et al. [42] and Drula et al. [16] for detailed introduction of the Bluetooth protocol stack.

The key concepts of WiFi-based device discovery are well understood. IEEE 802.11 standard defines several operation modes, including infrastructure and ad hoc modes. Stations in these modes will periodically send out Beacon messages to announce the presence of a network. A Beacon message contains timestamp (for synchronization), Beacon interval, capability information, service set identifier etc. The default Beacon interval for most of WiFi device drivers in Linux kernel is 100 ms. To support opportunistic communications, the WiFi interface of mobile phones need to operate in ad hoc mode, since they cannot form a network if they both operate in infrastructure mode. Besides sending out Beacon messages, they will also scan the wireless channels to discover peers. For two mobile phones to communicate with each other, they are required to form an Independent Basic Service Set (IBSS). Compared to Bluetooth, there is no service discovery protocol defined in IEEE 802.11 standard. After setting up a wireless connection, mobile phones can exchange data using either UDP or TCP. The maximal data rate of WiFi is 54 Mbps for 802.11g and can be up to 600 Mbps (theoretically) for 802.11n.

We conducted two groups of experiments to study battery life of mobile phones for different Bluetooth inquiry and WiFi scanning intervals, 1, 3, 10 and 30 seconds. We use Nokia N900 smartphones for the measurement study. Its default OS, Maemo 5, is an open source Linux

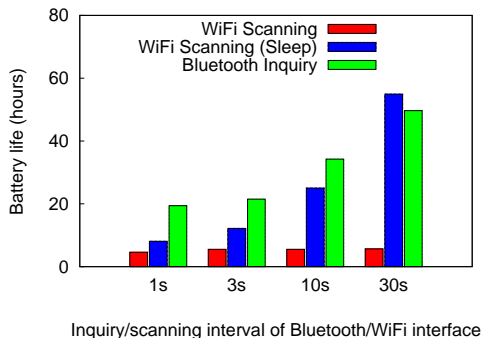


Fig. 3. Battery life of Nokia N900 smartphones for different inquiry and scanning interval of Bluetooth and WiFi interfaces.

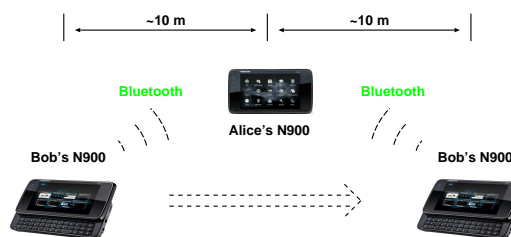


Fig. 4. The emulation of opportunistic communications between Alice and Bob's mobile phones.

distribution with kernel version 2.6.28. The WiFi chipset is Texas Instruments WL1251 which runs with the `wl12xx` device driver⁶. Its Bluetooth chipset is Broadcom BCM2048 and we use the default `BlueZ`⁷ Linux Bluetooth protocol stack for it. We show the measurement results in Figure 3. To save battery life, we also did another group of experiments for WiFi-based device discovery, during which we turned on the WiFi interface only when scanning, shown as WiFi Scanning (Sleep) in Figure 3. For Bluetooth, the inquiry lasts for 10.24 seconds (i.e., 4 trains), which is the standard value [26]. We measure the WiFi scanning duration on Nokia N900 smartphones using a scheme based on the implementation of `iwlist` command. During our experiments, the WiFi interface run in station mode and was not associated with any access point. The experimental results show that the duration is always less than 1 second for 802.11g, which has 11 channels in North America.

There are two observations from Figure 3. First, WiFi scanning will reduce battery life of a fully charged new N900 phone from longer than 310 hours to around only 5 hours. Although turning off the WiFi interface when not scanning can increase battery life, it will also decrease the device discovery probability, because a mobile phone with WiFi interface off cannot be discovered by others. Given the fact that the WiFi scanning duration is less than 1 second, if we turn off the WiFi interface when not scanning, for large scanning intervals most of the time the mobile phones stay in a non-discoverable state. Second, Bluetooth inquiry will not drain battery very quickly. Even when the inquiry interval is 1 second, the battery life is still around 20 hours

⁶<http://wireless.kernel.org/en/users/Drivers/wl12xx/>

⁷<http://www.bluez.org/>

and thus mobile users may not need to charge their phones during the daytime. Although WiFi can provide longer communication range and higher bandwidth, the high energy consumption makes it not suitable for device discovery. Thus, these results indicate that compared to WiFi, Bluetooth may be a better candidate for `Opp-Off`.

B. *Opp-Off Implementation*

We implement a simplified version of `Opp-Off` using Bluetooth for two reasons. First, as we have demonstrated that the energy consumption of WiFi scanning is much higher than that of Bluetooth inquiry. Second, Bluetooth is available on almost all the modern mobile phones. Whereas, only relatively few smartphones have WiFi interface. When mobile phones run `Opp-Off`, the program first starts a content server as a thread and then the main part is a loop that performs device discovery using Bluetooth inquiry (`hci_inquiry` function call in BlueZ). After two phones discover each other, they will start another client thread to connect to the content servers on the remote phones. If they can establish a connection, they will start transmitting data packets until the content transfer is finished or the connection is broken (e.g., because they are not within the Bluetooth communication range of each other due to movement).

In our current implementation, the inquiry duration is 10.24 seconds. We use the default values for other Bluetooth parameters, such as scan window and scan interval. The RFCOMM data packet length is 1000 bytes, which is smaller than the 1017-byte default MTU on our N900 phones. To guarantee that the data transfer is not affected by inquiry procedure, the device discovery procedure will be skipped once the connection is established. That is, a mobile phone cannot connect to multiple peers simultaneously. However, note that the connection is bi-directional. For example, after Alice's client connects to Bob's server, Alice's server can still accept connections from Bob's client.

C. *Evaluation*

We evaluate the device discovery probability, the number of transferred bytes and transfer duration between two mobile phones for different inquiry intervals. During the experiments, we emulate the opportunistic communication between Alice and Bob's mobile phones, as shown in Figure 4. In this scenario, the position of Alice and her N900 phone is fixed and Bob passes by Alice with his N900 phone. The communication range of a Class 2 Bluetooth device is around

Inquiry Interval (s)	1	3	10	30
Device Discovery Probability	40%	20%	50%	100%

TABLE I

THE DEVICE DISCOVERY PROBABILITY FOR 10 RUNS OF EXPERIMENT FOR DIFFERENT BLUETOOTH INQUIRY INTERVALS.

10 meters, which is the case for most mobile phones. Nokia N900 phones are equipped with Class 1 Bluetooth interface and thus the communication range is much longer than 10 meters. Thus we choose two points in a line with the distance from each end-point to the location of Alice’s phone to be around 10 meters. For a single experiment, Bob walks from one end-point to another, starting `Opp-Off` at one end-point and terminating it at another. Considering the moderate human walking speed is around 1 m/s, the contact duration of these two phones is about 20 seconds. If two Bluetooth devices enter inquiry status simultaneously with the same inquiry duration and interval, they will not be able to discover each other, because a Bluetooth device in inquiry status cannot respond inquiry messages from other devices. We start `Opp-Off` on these two phones randomly to avoid such synchronization.

We choose 4 Bluetooth inquiry intervals, 1, 3, 10 and 30 seconds and run the experiments 10 times for each interval. We show the device discovery probability in Table I. Due a single experiment, if Alice and Bob’s phones cannot discover each other, we say that they miss an opportunistic-communication opportunity. As we can see from this table, increasing the inquiry interval may increase the device discovery probability. When the inquiry interval is 30 seconds, these two phones can discover each other for all the 10 runs. One of the possible reasons is that during the inquiry interval, mobile phones are in inquiry scan status, and with fixed inquiry duration, longer inquiry scan duration can increase the chance that mobile phones receive the inquiry messages on the channel they are monitoring.

For the 30-second Bluetooth inquiry interval, we also plot the number of transferred bytes and the duration of data transfer for both directions of the 10 runs in Figure 5 and Figure 6. For 3 out of the 10 runs, both clients on Alice and Bob’s phones can connect to the servers running on the other phone. The maximum number of transferred bytes for both directions is 1,517.58 KB during the short contact of these two mobile phones. The average number of transferred bytes is 563.25 KB and the average duration of data transfer is 12.02 seconds. Note that, in practice if mobile phones have longer communication range, as Nokia N900 phones, they might be able

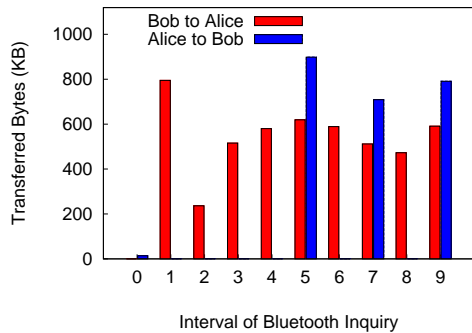


Fig. 5. The number of transferred bytes of 10 runs of experiment, for both directions. The inquiry interval is 30 seconds.

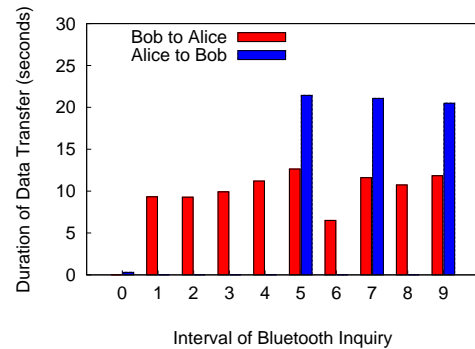


Fig. 6. The data transfer duration of 10 runs of experiment, for both directions. The inquiry interval is 30 seconds.

to exchange much more data through opportunistic communications.

To summarize, the above study shows that it is feasible to exploit the free opportunistic communications for mobile data offloading, even for the short contact during of moving mobile phones. In our current *Opp-Off* prototype, we only implemented the opportunistic communication part and we are currently working on the whole information dissemination framework.

D. Extensions

The unique features of various wireless technologies make them suitable for different tasks. For example, Bluetooth may be suitable for device discovery due to its low energy consumption, but WiFi may be a better solution for content transfer, because the energy consumption per bit for WiFi is lower than that of Bluetooth. To increase battery lifetime, CoolSpots [39] investigates the policies to enable mobile devices to automatically switch between multiple radio interfaces (e.g., WiFi and Bluetooth). Ananthanarayanan and Stoica propose Blue-Fi [3], a system that predicts WiFi availability through Bluetooth contact patterns and cell-tower information. It allows devices to intelligently turn on their WiFi interface only when there is a WiFi access point in its proximity, thus reducing the energy consumption for discovery.

Motivated by the success of these previous works, we are evaluating the pros and cons of WiFi and Bluetooth for different phases. Sticking with one technology may not be the best solution and thus we plan to combine them to make the whole procedure more effective, that is, using Bluetooth for device and service discovery and WiFi for content transfer. Another possible extension is to adaptively adjust the inquiry interval according to human mobility patterns and

develop energy conscious device discovery protocols [16], [45]. If mobile phones send out inquiry messages infrequently, the device discovery probability may be very low. On the other hand, frequent inquiries may be energy inefficient. We are investigating the trade-off between device discovery probability and energy consumption for mobile phones.

Since it is hard to find a large number of mobile users for the performance evaluation of opportunistic-communication based offloading in the wild, we use trace-driven simulation to evaluate the three proposed algorithms in the next section.

VI. SIMULATION

We now introduce the mobility traces that we use for performance evaluation, and then present the results from a trace-driven simulator developed in C. The simulator first loads contact events from real-world traces or generates contact events based on the movement history from the synthetic traces. It then replays the contact events for the given information dissemination periods. At the beginning of each contact, the simulator determines randomly whether a mobile user can get the information from the peer based on the pre-configured pull probability.

A. Mobility Traces

1) *Synthetic Mobility Trace*: We use the SIGMA-SPECTRUM simulator [8] to generate a synthetic mobility trace in the region of Portland, Oregon. The simulator combines different real-world data sources and realistic models, including an urban mobility model, synthetic population (according to U.S. Census data) and road-network data of Portland. The trace records the location of mobile users every 30 seconds. We randomly choose 10,000 people from the entire population of the city (around 1,600,000 people) as the subscribed users. The information dissemination periods start from 7:00AM with different durations. Note that the duration of the information dissemination period is, in fact, also the delay-tolerance threshold for mobile users (i.e., the maximum delay they need to tolerate). We use this trace to evaluate the performance of the Random algorithm for different pull probabilities and delay-tolerance thresholds.

2) *Traces From Real-World Experiments*: To evaluate the performance of the Heuristic algorithm, we need the mobility traces of different days, which is not available in the SIGMA-SPECTRUM simulator. To this end, we exploit two real-world mobility traces from the Huggle project [12] and the Reality Mining project [18].

	History	Delivery
#1	2006-04-24 11:00AM	2006-04-25 11:00AM
#2	2006-04-25 11:00AM	2006-04-26 11:00AM
#3	2006-04-25 12:00PM	2006-04-26 12:00PM

TABLE II

THE START TIME OF THREE SELECTED 1-HOUR PERIODS
FROM INFOCOM06 TRACE.

	History	Delivery
#1	2004-10-25 12:00PM	2004-10-28 12:00PM
#2	2004-11-15 12:00PM	2004-11-22 12:00PM
#3	2004-12-06 12:00PM	2004-12-07 12:00PM

TABLE III

THE START TIME OF THREE SELECTED 6-HOUR PERIODS
FROM REALITY MINING TRACE.

We use the INFOCOM06 trace collected by the Huggle project for 4 days (from 2006-04-24 to 2006-04-27) during INFOCOM 2006 in Barcelona, Spain. This trace recorded the mobility of students and researchers attending the student workshop, using 78 iMotes which had a communication range of around 30 meters. We select 3 pairs of 1-hour periods from the trace as shown in Table II. Thus, the delay-tolerance threshold is 1 hour for this trace. To exploit the 24-hour regularity of human mobility and evaluate the performance of the `Heuristic` algorithm, we use the target set identified by the `Greedy` algorithm for the periods in the second column (“History”) to predict the mobility of users for the periods in the third column (“Delivery”) of the same row. We define active users as those who have at least 1 contact with others during the delivery periods. As a result, the numbers of active users for these periods are 70, 66 and 66. We can also use other thresholds instead of 1. But they may exclude some inactive users for the simulation and thus reduce the (already small) number of simulated users.

The Reality Mining trace was collected using 100 Nokia 6600 smartphones carried by people from the MIT Media Laboratory and Sloan Business School, from 2004-07-26 to 2005-05-05. The information in this trace includes call logs, neighboring Bluetooth devices, and associated cell-tower IDs, etc. The contact trace of these users identified by the Bluetooth scanning is very sparse and thus is not suitable for the simulation. As in Ioannidis et al. [28], we instead consider that two mobile users are in contact of each other if their phones are associated with the same cell tower. Even this cell-tower based contact trace is sparse: this is the reason that we use 6-hour periods for the simulation. Therefore, the delay-tolerance threshold is 6 hours for this trace. Similar to Table II, we show the 3 pairs of 6-hour periods from the trace in Table III. Benefiting from the long duration of the Reality Mining project, we can also exploit the 3-day (#1 of Table III) and 1-week (#2 of Table III) regularity of human mobility. The numbers of active users for these three periods are 61, 71 and 53 for the Reality Mining trace. For both

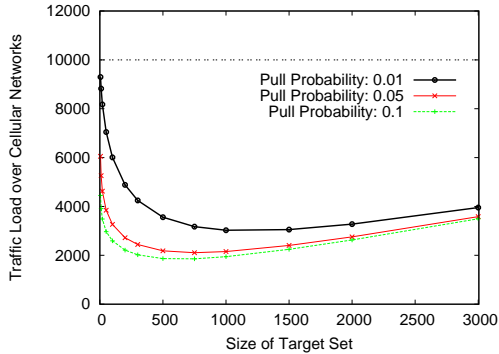


Fig. 7. Performance of Random algorithm for different pull probabilities (Portland city data set).

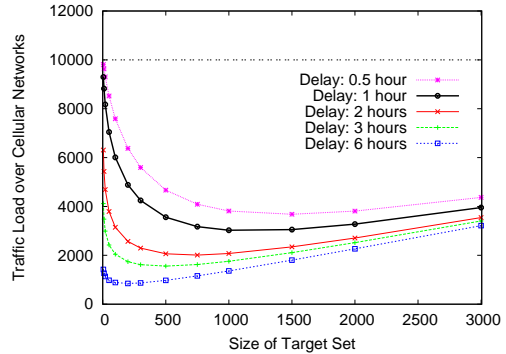


Fig. 8. Performance of Random algorithm for different delay-tolerance thresholds (Portland city data set).

traces, we use only active users in the simulation.

B. Simulation Results

In this section, we present the simulation results of the Random, Heuristic, and Greedy algorithms. In the simulation, we emulate the information delivery of multimedia newspapers (with size around several MB). Each direct cellular delivery consumes one message containing the newspaper and for simplicity we assume there is no further packetization. The simulated duration of a single run is determined by the corresponding delay-tolerance threshold. Our goal here is to determine the target set which leads to the most efficient mobile data offloading.

1) *Pull Probability*: We first evaluate the performance of Random algorithm for different pull probabilities using the Portland trace. We show the mobile data traffic load for different sizes of target set, from 5 to 3,000, and pull probabilities, 0.01, 0.05 and 0.1, in Figure 7. The x-axis is the size of target set and the y-axis show the mobile traffic load, in terms of the number of cellular messages. Every user who fails to receive the information before the delivery deadline will consume a cellular message. Moreover, each user in the target set will also consume a cellular message. The delivery deadline is 1 hour. For each combination of the size of target set and pull probability, we run the simulation 10,000 times and report the average value. The horizontal dotted line shows the amount of cellular messages without offloading, which is the same as the total number of subscribed users. As we can see from this figure, even for the very simple random algorithm, it can reduce the amount of mobile data traffic by up to 81.42% when the pull probability is 0.1. When we reduce the pull probability to 0.01, it can still offload mobile

data traffic by up to 69.73%.

There are two main observations from this figure. First, the amount of mobile data traffic decreases as the pull probability increases. It is because when mobile users are all active in information propagation, a large number of users can get the delivered information from their peers through opportunistic communications, and thus avoid the data transmissions over cellular networks. Hence, active *social participation* is a key enabling factor of efficient information delivery. Second, as the size of target set increases, the amount of mobile data traffic first decreases and then increases. The reasons are: (1). when the size of target set is small, the expected number of users that can receive the information through opportunistic communications is also small and thus a large number of users need to get the information through cellular networks; (2). when the size of target set is large, although it can make more users receive the information through opportunistic communications, the users in the target set will directly generate a large amount of mobile data traffic.

For the three curves in Figure 7, the pull probability is fixed for all the contacts of these mobile users. We also tried different probabilities for different contacts, uniformly and randomly selected between 0.01 and 0.1. The result looks very similar to the curve with pull probability 0.05. Thus, we omit that result for clarity. Note that, since information service providers will deliver information to those users who cannot receive it before delay-tolerance threshold, the delivery percentage is always 100% in our mobile data offloading solutions.

2) *Delay-Tolerance Threshold*: We then evaluate the performance of Random algorithm for different delay-tolerance thresholds for the Portland trace. We show the traffic load over cellular networks for five delay-tolerance thresholds, 0.5, 1, 2, 3 and 6 hours, in Figure 8, as different types of data have different delay-tolerance requirements. The pull probability is 0.01. We also run the simulation 10,000 times for a point in that plot and report the average value. As we can see from this figure, if mobile users are willing to tolerate longer delay we may be able to offload more traffic from cellular networks. However, the benefit of increasing the delay-tolerance threshold from 2 hours to 3 hours is not very significant, compared to that from 1 hour to 2 hours. One possible reason is that when we increase the threshold to 2 hours, most of the active users can receive the delivered information through opportunistic communications and thus the improvement of increasing it to 3 hours is limited.

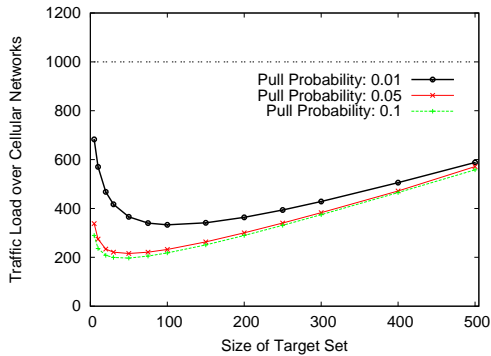


Fig. 9. Performance of Random algorithm for different pull probabilities (Utah state data set).

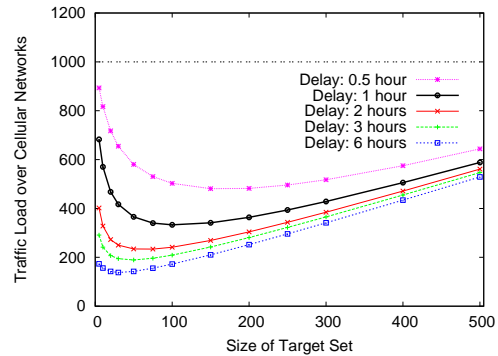


Fig. 10. Performance of Random algorithm for different delay-tolerance thresholds (Utah state data set).

3) *Another Synthetic Mobility Trace*: We also validate the simulation results about pull probability and delay-tolerance threshold on a smaller synthetic mobility trace, again generated by the SIGMA-SPECTRUM simulator [8]. This time, we randomly choose 1,000 people around the Salt Lake City area as subscribed users. Other settings are similar to those of the Portland trace. We plot the results in Figure 9 and Figure 10, which show comparable trends as in Figure 7 and Figure 8.

4) *Comparing Random, Heuristic, and Greedy*: We compare the performance of Random, Heuristic and Greedy algorithms using the two real-world traces. To verify the regularity of human mobility, we show in Table IV the IDs of the top 5 most active users for 2 pairs of selected periods, for the INFOCOM06 trace and the Reality Mining trace. The numbers in the parentheses are the expected number of infected users when each of the active users is selected as the single user in the target set. From this table, we can see that the most active user (with ID 43) for the period 2006-04-25 11:00AM-12:00PM is the second most active user for the period 2006-04-26 11:00AM-12:00PM for the INFOCOM06 trace. For the Reality Mining trace, the most active user for the period 2004-12-06 12:00PM-06:00PM is also the most active one for the period 2004-12-07 12:00PM-06:00PM. For almost all the other periods, the most active user of the *History* period is in the top 5 most active users of the *Delivery* period. We summarize the two traces and the parameters used in the simulation in Table V.

We plot in Figure 11 and Figure 12 the traffic load over cellular networks for the 6 pairs of periods listed in Table II and Table III. Due to the small number of mobile users in the traces, we set the size of target set to be 5. For the Random and Heuristic algorithms, we simulate

Start at	No. 1	No. 2	No. 3	No. 4	No. 5
2006-04-25 11:00AM	43 (31.18)	53 (31.17)	40 (30.77)	73 (29.46)	78 (29.31)
2006-04-26 11:00AM	68 (18.08)	43 (16.67)	69 (15.78)	60 (14.98)	30 (14.86)
2004-12-06 12:00PM	94 (34.07)	15 (34.03)	80 (34.01)	97 (33.61)	7 (33.57)
2004-12-07 12:00PM	94 (26.22)	95 (26.07)	15 (25.97)	92 (25.79)	7 (25.31)

TABLE IV

THE TOP 5 MOST ACTIVE USERS FOR DIFFERENT PERIODS AND THE EXPECTED NUMBER USERS THAT THEY CAN INFECT.

Trace	Haggle	MIT Reality
	INFOCOM06	Mining
Network type	Bluetooth	Bluetooth
Device type	iMote	Nokia 6600
Number of devices	78	100
Duration of trace	4 days	9 months
Regularity	1 day	1, 3, 7 days
Simulated duration	1 hour	6 hours
Pull probability	0.01	0.001
# of Active users	≤ 70	≤ 71

TABLE V

SUMMARY OF TWO REAL-WORLD TRACES.

the information dissemination process 100,000 times and report the averaged values. For the Greedy algorithm, we run the simulation 10,000 times to determine the marginal gain for each user. After we identify the target users, we also run the simulation 100,000 times and report the averaged values. In these figures, the Base shows the amount of mobile data traffic without offloading, which is the same as the number of active users during these periods.

The performance of these algorithms depends on the pull probability. The pull probability is 0.01 for the INFOCOM06 trace and 0.001 for the Reality Mining trace. For high pull probabilities, there is no significant difference among them. As we can see from these figures, Greedy performs the best, followed by the Heuristic algorithm, for all the cases. Compared to the Base, the Random algorithm can reduce the amount of mobile data traffic by up to 53.91% for the INFOCOM06 trace and 70.72% for the Reality Mining trace. Owing to the regularity of human mobility, Heuristic can further reduce the amount of mobile data traffic of Random by up to 18.95% for the INFOCOM06 trace and 12.25% for the Reality Mining trace. Although Greedy and Heuristic perform better than Random, the difference is not very significant. One of the possible reasons is that due to the small number of mobile users and their limited active area, even if we choose the target users randomly, with high probability the information will be disseminated to some very active users quickly, who will then affect a large number of other users. Compared to the Greedy and Heuristic algorithms, a unique advantage of the Random algorithm is that information service providers can avoid collecting the contact information from subscribed users, which may make them feel comfortable to participate in the

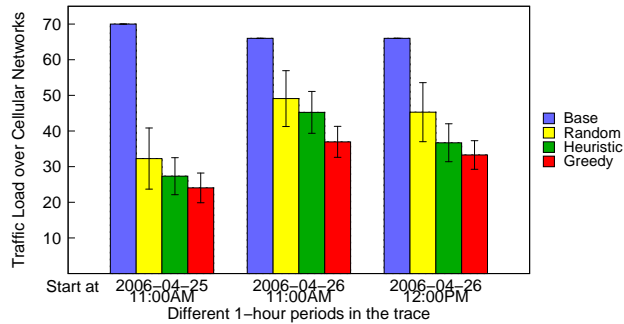


Fig. 11. Performance comparison of Random, Heuristic, and Greedy algorithms for the INFOCOM06 data set.

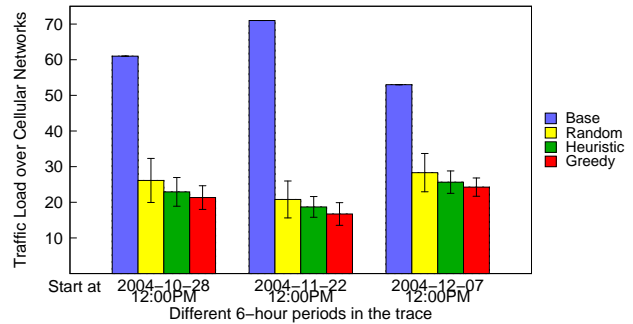


Fig. 12. Performance comparison of Random, Heuristic, and Greedy algorithms for the Reality Mining data set.

information dissemination.

We note that due to the incompleteness of the real-world traces (e.g., caused by hardware errors), some users in the target set of the *History* period may not be active during the *Delivery* period (i.e., they have *no* contacts with other users for the delivery period). In these cases, we replace them with randomly selected users. We have not evaluated how the push-based approach can help the information dissemination among friends, because there is no information about the social graph of mobile users for the above traces. However, we note that it is possible to construct the graph through the analysis of traffic between mobile users [49], or historical data of mobile users, such as proximity and location at a given time [18]. We leave the evaluation of push-based approach as a future work.

VII. DISCUSSION

In this section, we discuss several practical issues for the large-scale deployment of our proposed mobile data offloading solution.

A. Incentives

The integration of effective incentive schemes into mobile data offloading is a challenging problem. For information service providers, with the offloading solution they can decrease the number of cellular messages and thus reduce their operation cost. As a result, they may reduce the subscription fee for their customers. To encourage social participation of mobile users, information service providers can also exploit other incentives: see, e.g., the Coupons approach of Garyfalos and Almeroth [21]. This system appends a sorted list of user IDs to a

propagated message, which records the sequence of users who helped to disseminate the message. Similarly, information service providers can ask mobile users to optionally report when they got the delivered information and from where. Then they can offer discounts to mobile users who actively help the information delivery process. Recently, Misra et al. [35] propose a solution that provides incentives for peer-assisted services. Their goal is to develop an economic framework that creates the right incentives for both users and providers. They exploit a cooperative game theory approach to determine the ideal incentive structure through fluid Shapley value. Applying this scheme into our mobile data offloading solution is our ongoing work.

B. Energy Consumption

Energy consumption may be the most important issue for the deployment of mobile applications. As we mentioned in Section V, the three main phases of opportunistic communications are device and content discovery, and data transfer. Although we have chosen Bluetooth for device discovery, we use fixed parameters (e.g., inquiry duration and interval, and inquiry scan window and interval) in our current implementation. We believe dynamically changing these parameters according to user mobility patterns may make the device discovery procedure more energy efficient. For example, when users are not moving, larger inquiry interval may be a better choice. Since device discovery is a common component for several mobile applications like Social Serendipity [17] and Media Sharing [32], its energy consumption can also be amortized by them. Moreover, for data transfer, we suggested to replace Bluetooth using WiFi to save battery life.

C. Privacy

Unlike some existing protocols which determine whether to exchange information during the contact period [10], we aim to provide a general platform for information dissemination among mobile users. It is the users, not the platform, who make the decision about whether or not to share the information with peers and thus can protect their content privacy. They can opt-in and opt-out of the information dissemination process anytime they want, by turning off only the information dissemination application (not necessarily the 3G radio). Moreover, information service providers select only the users in target set and will not dynamically tag them or others as relays. Mobile users will act as relays only when they participate in the information dissemination process and have already held the information that others may be interested in. We finally note

that we require only the contact information among the users and there is no need to track mobile users' locations to enable our proposed solution.

VIII. CONCLUSION

We propose to offload mobile data traffic through opportunistic communications and investigate the target-set selection problem for information delivery in MoSoNets. We present three algorithms for this problem, `Random`, `Heuristic`, and `Greedy`, and evaluate their performance through trace-driven simulation, using both large-scale synthetic and real-world mobility traces. The simulation results show that `Greedy` performs the best, followed by `Heuristic`. Although the `Greedy` algorithm may not be practical, it is the basis of the `Heuristic` algorithm which exploits the regularity of human mobility. We also implement a prototype of the information delivery framework using Nokia N900 smartphones and study its feasibility for moving phones. Our preliminary experimental results show that during their short contacts mobile phones can exchange up to 1.48 MB data.

IX. ACKNOWLEDGEMENT

We thank the anonymous reviewers for their insightful comments. We thank Kan-Leung Cheng and Xiaoyu Zhang for useful discussions. We also thank our external collaborators and members of the Network Dynamics and Simulation Science Laboratory (NDSSL) for their suggestions and comments. Aravind Srinivasan and Bo Han were supported in part by NSF ITR Award CNS-0426683, NSF Award CNS-0626636, and NSF Award CNS 1010789. The work of V.S. Anil Kumar and Madhav Marathe has been partially supported by NSF Nets Grant CNS-0626964, NSF HSD Grant SES-0729441, NSF PetaApps Grant OCI-0904844, DTRA R&D Grant HDTRA1-0901-0017, DTRA CNIMS Grant HDTRA1-07-C-0113, NSF NETS CNS-0831633, DHS 4112-31805, DOE DE-SC0003957, NSF CNS-0845700, US Naval Surface Warfare Center N00178-09-D-3017 DEL ORDER 13, NSF Netse CNS-1011769 and NSF SDCI OCI-1032677. Part of this work was done when Bo Han and Jianhua Shao were summer interns at Deutsche Telekom Laboratories, supported by the MADNet Project.

REFERENCES

- [1] Mobile Data Offload for 3G Networks. White Paper, IntelliNet Technologies, Inc., 2009.

- [2] Mobile data traffic surpasses voice. <http://www.cellular-news.com/story/42543.php>, 2010.
- [3] G. Ananthanarayanan and I. Stoica. Blue-Fi: Enhancing Wi-Fi Performance using Bluetooth Signals. In *Proceedings of MobiSys 2009*, pages 249–262, June 2009.
- [4] A. Balasubramanian, R. Mahajan, and A. Venkataramani. Augmenting Mobile 3G Using WiFi. In *Proceedings of MobiSys 2010*, pages 209–222, June 2010.
- [5] N. Balasubramanian, A. Balasubramanian, and A. Venkataramani. Energy Consumption in Mobile Phones: A Measurement Study and Implications for Network Applications. In *Proceedings of IMC 2009*, pages 280–293, Nov. 2009.
- [6] C. L. Barrett, S. J. Eidenbenz, L. Kroc, M. Marathe, and J. P. Smith. Parametric Probabilistic Routing in Sensor Networks. *Mobile Networks and Applications*, 10(4):529–544, Aug. 2005.
- [7] A. Beach, M. Gartrell, S. Akkala, J. Elston, J. Kelley, K. Nishimoto, B. Ray, S. Razgulin, K. Sundaresan, B. Surendar, M. Terada, and R. Han. WhozThat? Evolving an Ecosystem for Context-Aware Mobile Social Networks. *IEEE Network*, 22(4):50–55, July-Aug. 2008.
- [8] R. Beckman, K. Channakeshava, F. Huang, V. S. A. Kumar, A. Marathe, M. V. Marathe, and G. Pei. Implications of Dynamic Spectrum Access on the Efficiency of Primary Wireless Market. In *Proceedings of DySPAN 2010*, pages 1–12, Apr. 2010.
- [9] Bluetooth Special Interest Group. Specification of the Bluetooth System, Version 2.1 + EDR, 2007.
- [10] C. Boldrini, M. Conti, and A. Passarella. Context and resource awareness in opportunistic network data dissemination. In *Proceedings of WoWMoM 2008*, pages 1–6, June 2008.
- [11] S. Burleigh. Contact Graph Routing. Internet-Draft, draft-burleigh-dtnrg-cgr-00, 2009.
- [12] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott. Impact of Human Mobility on Opportunistic Forwarding Algorithms. *IEEE Transactions on Mobile Computing*, 6(6):606–620, June 2007.
- [13] V. Chandrasekhar, J. G. Andrews, and A. Gatherer. Femtocell Networks: A Survey. *IEEE Communications Magazine*, 46(9):59–67, Sept. 2008.
- [14] F. Chierichetti, S. Lattanzi, and A. Panconesi. Rumour Spreading and Graph Conductance. In *Proceedings of SODA 2010*, pages 1657–1663, Jan. 2010.
- [15] P. Domingos and M. Richardson. Mining the Network Value of Customers. In *Proceedings of SIGKDD 2001*, pages 57–66, Aug. 2001.
- [16] C. Drula, C. Amza, F. Rousseau, and A. Duda. Adaptive Energy Conserving Algorithms for Neighbor Discovery in Opportunistic Bluetooth Networks. *IEEE Journal on Selected Areas in Communications*, 25(1):96–107, Jan. 2007.
- [17] N. Eagle and A. Pentland. Social Serendipity: Mobilizing Social Software. *IEEE Pervasive Computing*, 4(2):28–34, Apr.-June 2005.
- [18] N. Eagle, A. S. Pentland, and D. Lazer. Inferring friendship network structure by using mobile phone data. *Proceedings of the National Academy of Sciences*, 106(36):15274–15278, Sept. 2009.
- [19] K. Fall. A Delay-Tolerant Network Architecture for Challenged Internets. In *Proceedings of SIGCOMM 2003*, pages 27–34, Aug. 2003.
- [20] S. Gaonkar, J. Li, R. R. Choudhury, L. Cox, and A. Schmidt. Micro-Blog: Sharing and Querying Content Through Mobile Phones and Social Participation. In *Proceedings of MobiSys 2008*, pages 174–186, June 2008.
- [21] A. Garyfalos and K. C. Almeroth. Coupons: A Multilevel Incentive Scheme for Information Dissemination in Mobile Networks. *IEEE Transactions on Mobile Computing*, 7(6):792–804, June 2008.

- [22] J. Goldenberg, B. Libai, and E. Muller. Talk of the Network: A Complex Systems Look at the Underlying Process of Word-of-Mouth. *Marketing Letters*, 12(3):211–223, Aug. 2001.
- [23] M. C. González, C. A. Hidalgo, and A.-L. Barabási. Understanding individual human mobility patterns. *Nature*, 453(7196):779–782, June 2008.
- [24] D. Gruhl, R. Guha, D. Liben-Nowell, and A. Tomkins. Information Diffusion Through Blogspace. In *Proceedings of WWW 2004*, pages 491–501, May 2004.
- [25] B. Hoppe and Éva Tardos. The Quickest Transshipment Problem. In *Proceedings of SODA 1995*, pages 512–521, Jan. 1995.
- [26] A. S. Huang and L. Rudolph. *Bluetooth Essentials for Programmers*. Cambridge University Press, 2007.
- [27] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, and F. Silva. Directed Diffusion for Wireless Sensor Networking. *IEEE/ACM Transactions on Networking*, 11(1):2–16, Feb. 2003.
- [28] S. Ioannidis, A. Chaintreau, and L. Massoulié. Optimal and Scalable Distribution of Content Updates over a Mobile Social Network. In *Proceedings of the IEEE INFOCOM 2009*, pages 1422–1430, Apr. 2009.
- [29] D. Kempe, J. Kleinberg, and Éva Tardos. Maximizing the Spread of Influence through a Social Network. In *Proceedings of SIGKDD 2003*, pages 137–146, Aug. 2003.
- [30] U. Lee, S. Jung, D.-K. Cho, A. Chang, J. Choi, and M. Gerla. P2P Content Distribution to Mobile Bluetooth Users. *IEEE Transactions on Vehicular Technology*, 59(1):356–367, Jan. 2010.
- [31] C. Lindemann and O. P. Waldhorst. Modeling Epidemic Information Dissemination on Mobile Devices with Finite Buffers. In *Proceedings of SIGMETRICS 2005*, pages 121–132, June 2005.
- [32] L. McNamara, C. Mascolo, and L. Capra. Media Sharing based on Colocation Prediction in Urban Transport. In *Proceedings of MOBICOM 2008*, pages 58–69, Sept. 2008.
- [33] A. Mei and J. Stefa. SWIM: A Simple Model to Generate Small Mobile Worlds. In *Proceedings of the IEEE INFOCOM 2009*, pages 2106–2113, Apr. 2009.
- [34] E. Miluzzo, N. D. Lane, K. Fodor, R. Peterson, H. Lu, M. Musolesi, S. B. Eisenman, X. Zheng, and A. T. Campbell. Sensing Meets Mobile Social Networks: The Design, Implementation and Evaluation of the CenceMe Application. In *Proceedings of SenSys 2008*, pages 337–350, Nov. 2008.
- [35] V. Misra, S. Ioannidis, A. Chaintreau, and L. Massoulié. Incentivizing Peer-Assisted Services: A Fluid Shapley Value Approach. In *Proceedings of SIGMETRICS 2010*, pages 215–226, June 2010.
- [36] M. Motani, V. Srinivasan, and P. S. Nuggehalli. PeopleNet: Engineering A Wireless Virtual Social Network. In *Proceedings of MOBICOM 2005*, pages 243–257, Aug.-Sept. 2005.
- [37] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions. *Mathematical Programming*, 14(1):265–294, Dec. 1978.
- [38] M. Papadopouli and H. Schulzrinne. Effects of power conservation, wireless coverage and cooperation on data dissemination among mobile devices. In *Proceedings of MOBIHOC 2001*, pages 117–127, Oct. 2001.
- [39] T. Pering, Y. Agarwal, R. Gupta, and R. Want. CoolSpots: Reducing the Power Consumption of Wireless Mobile Devices with Multiple Radio Interfaces. In *Proceedings of MobiSys 2006*, pages 220–232, June 2006.
- [40] M. Pitkänen, T. Kärkkäinen, and J. Ott. Opportunistic Web Access via WLAN Hotspots. In *Proceedings of PerCom 2010*, pages 20–30, Mar.-Apr. 2010.
- [41] M. Richardson and P. Domingos. Mining Knowledge-Sharing Sites for Viral Marketing. In *Proceedings of SIGKDD 2002*, pages 61–70, July 2002.

- [42] T. J. Smith, S. Saroiu, and A. Wolman. BlueMonarch: A System for Evaluating Bluetooth Applications in the Wild. In *Proceedings of MobiSys 2009*, pages 41–53, June 2009.
- [43] T. W. Valente. *Network Models of the Diffusion of Innovations*. Hampton Press, 1995.
- [44] V. Vukadinović and G. Karlsson. Spectral Efficiency of Mobility-Assisted Podcasting in Cellular Networks. In *Proceedings of MobiOpp 2010*, pages 51–57, Feb. 2010.
- [45] W. Wang, V. Srinivasan, and M. Motani. Adaptive Contact Probing Mechanisms for Delay Tolerant Applications. In *Proceedings of MOBICOM 2007*, pages 230–241, Sept. 2007.
- [46] D. J. Watts and S. H. Strogatz. Collective dynamics of ‘small-world’ networks. *Nature*, 393(6684):440–442, June 1998.
- [47] Z. Yang, B. Zhang, J. Dai, A. Champion, D. Xuan, and D. Li. E-SmallTalker: A Distributed Mobile System for Social Networking in Physical Proximity. In *Proceedings of ICDCS 2010*, pages 468–477, June 2010.
- [48] T. Zahn, G. O’Shea, and A. Rowstron. Feasibility of Content Dissemination Between Devices in Moving Vehicles. In *Proceedings of CoNEXT 2009*, pages 97–108, Dec. 2009.
- [49] Z. Zhu, G. Cao, S. Zhu, S. Ranjan, and A. Nucci. A Social Network Based Patching Scheme for Worm Containment in Cellular Networks. In *Proceedings of the IEEE INFOCOM 2009*, pages 1476–1484, Apr. 2009.

Bo Han received the Bachelor’s degree in Computer Science and Technology from Tsinghua University in 2000 and the M.Phil. degree in Computer Science from City University of Hong Kong in 2006. He is currently a Ph.D. candidate in the Department of Computer Science at the University of Maryland, College Park. He worked as research intern at AT&T Labs Research for summers 2007, 2008 and 2009, and Deutsche Telekom Laboratories for summer 2010. His research interests include wireless communication, mobile computing, distributed algorithms and Internet routing.

Pan Hui is a senior research scientist in Deutsche Telekom Laboratories, Berlin. He received his PhD from Computer Laboratory, University of Cambridge. During his PhD, he also affiliated with Intel Research Cambridge. Before that he was with University of Hong Kong for his Mphil and bachelor degree. His research interests include delay tolerant networking, mobile networking and systems, planet-scale mobility measurement, social networks, and the application of complex network science in communication system design. More information about his profile and his research work can be found at <http://www.deutsche-telekomlaboratories.de/~panhui/>

V. S. Anil Kumar is currently an Assistant Professor in the Department of Computer Science and the Virginia Bioinformatics Institute, Virginia Tech. Prior to this, he was a technical staff member at Los Alamos National Laboratory. He received a Ph.D. in Computer Science from the Indian Institute of Science in 1999 and a B.Tech in Computer Science and Engineering from the Indian Institute of Technology, Kanpur in 1993. His research interests include approximation algorithms, mobile computing, combinatorial optimization and simulation of large socio-technical systems.

Madhav V. Marathe is a Professor in the Dept. of Computer Science and the Virginia Bioinformatics Institute at Virginia Tech. He obtained his PhD from University at Albany, SUNY in 1994. He is a Senior member of IEEE. His research interests are in modeling of large complex systems, algorithms, wireless networks, medical and health informatics and applications of computing to societal problems.

Jianhua Shao received the BSc Hons in Computer Science from University of Nottingham (UK) in 2010. He is currently a PhD student in the Doctorial Training Centre in Horizon Digital Economy Hub in the University of Nottingham (UK). He interned at Deutsche Telekom Laboratories for summer 2010. His research interests include mobile network and social context.

Aravind Srinivasan (Fellow, IEEE) is a Professor (Dept. of Computer Science and Institute for Advanced Computer Studies) at the University of Maryland, College Park. He received his degrees from Cornell University (Ph.D.) and the Indian Institute of Technology, Madras (B.Tech.). His research interests are in randomized algorithms, networking, social networks, combinatorial optimization, and related areas. He has published several papers in these areas, in journals including Nature, Journal of the ACM, IEEE/ACM Transactions on Networking, and SIAM Journal on Computing. He is an editor of four journals, and has served on the program committees of various conferences.