This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2018.2867437, IEEE Transactions on Services Computing

Integrating Social Networks with Mobile Device-to-Device Services

Xiaofei Wang, *Member IEEE,* Min Chen*, *Member IEEE,* Victor C. M. Leung, *Fellow IEEE,* Zhu Han, *Fellow IEEE,* and Kai Hwang, *Fellow IEEE*

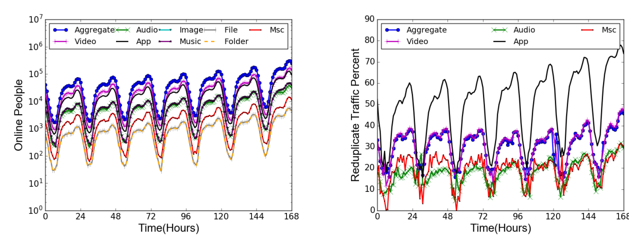
Abstract—In recent years, the rapid growth of traffic becomes a serious problem of mobile network operators. For effectively mitigating this traffic explosion problem, there have been many efforts to research on offloading the traffic from cellular links to direct communications among users. In this paper, we are motivated by users' sharing activities, and hence propose the framework of ***T**raffic* ***O**ffloading assisted by* ***S**ocial network services (SNS) via opportunistic* ***S**haring in mobile social networks (MSNs), *TOSS*, to offload SNS-based cellular traffic by user-to-user sharing. First, a subset of users who are to receive the same content was selected as initial population depending on their content spreading impacts in the online SNSs and their mobility patterns in the offline MSNs. Then users move, encounter and share the content via opportunistic local connectivity with each other, the content via opportunistic local connectivity with each other, e.g., Bluetooth, Wi-Fi Direct, Device-to-Device in LTE. Individual users have distinct access patterns, which potentially allows TOSS to exploit the user-dependent access delay between the content generation time and each user's access time for content sharing purposes. The traffic offloading and content spreading among users are analyzed by taking into account various options in linking SNS and MSN traces. Four mobility traces and online SNS trace for evaluation are analyzed. An extended evaluation over a large-scale data set are further carried out, and the effectiveness of TOSS is further proved.

Index Terms—Traffic Offloading, Mobile Social Networks, Social Network, Opportunistic Networks, Device-to-Device Communication

1 Introduction

Because of the fast development of mobile communication technologies, more and more users tend to download and enjoy multimedia content on mobile terminals. And hence the ever increasing traffic load becomes a serious concern of mobile network operators (MNOs), but the study in [2] points out that a large portion of the traffic load is due to the duplicated downloads of the same popular files. Therefore, how to effectively reduce the duplicated downloads over cellular links by *offloading* the traffic load via local short-range communications directly, so that users may cache and re-share the content to potential neighbors, would be crucial.

Recently there have been many studies to explore the opportunistic user-to-user (device-to-device, D2D) sharing during intermittent encounters/meetings of mobile users for traffic offloading in mobile social networks (MSNs), which can be considered as a special type of the Delay Tolerant Network (DTN) with more emphasis on the social relationship among users [3] [4] [5] [6] [7]. Note that in some studies [8] a DTN/MSN can be regarded as the opportunistic network as well. Users in MSNs are able to discover the adjacent neighbors [9] and thus set up temporary connectivities locally, e.g., Wi-Fi Direct, Bluetooth, Near-Field-Communication, and D2D [10] in LTE, for sharing content with each other, without using any 3G/4G cellular network data, Particularly, D2D has been designed in 3GPP as an underlay to LTE-Advanced networks, by which users can utilize operator's authorized spectrum for direct communications without intervention of the cellular infrastructure, and thus is currently gaining significant momentum [10].



(a) Number of online individuals (b) Reduplicate traffic percentage

Fig. 1. Performance of *Xender*'s filtered trace, the 1st week of Feb. 2016

In addition, recent evidences demonstrate the great potential capacity of opportunistic D2D communications. Particularly, there have been a number of mature D2D sharing applications (APPs) in the mobile markets already, attracting millions of users, e.g., *Xender* [11], *SHAREit* [12] and *Zapya* [13]. And hence the cellular offloading capacity would make the D2D-enabled MSN an effective solution to relieve the MNO's traffic burden and to enter a new stage of optimal Internet service. For instance, Fig. 1 illustrates the time-varying measurements of a filtered subset trace of *Xender*, which includes 5 million mobile users with more than 90 million D2D transmission activities during the first week of February, 2016. For the whole data set during February 2016 [31], *Xender* has served around 9 million active users daily and 100 million active users monthly, as well as 110 million daily content deliveries, in total. Fig. 1(a) represents

• *Some parts of this paper were published in IEEE INFOCOM 2014 [1], and we have extended the content significantly.*
• *X. Wang is with the Department of Computer Science and Technology, Tianjin University, E-mail: xiaofeiwang@tju.edu.cn*
• *M. Chen is with the Department of Computer Science and Technology, Huazhong University of Science and Technology, E-mail: minchen2012@hust.edu.cn (corresponding author)*
• *V. C. M. Leung is with the Department of Electrical and Computer Engineering, The British Columbia University, E-mail: vleung@ece.ubc.ca*
• *Z. Han is with the Department of Electrical and Computer Engineering, University of Houston, E-mail: zhan2@uh.edu*
• *K. Hwang is with the Electrical Engineering and Computer Science, University of Southern California, E-mail: kaihwang@usc.edu*

Manuscript received January 22, 2017; revised XX XX, 2017.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2018.2867437, IEEE Transactions on Services Computing

the number of individual online users, and notably, users who share videos and pictures take the highest portion. Obviously *Xender* helps the mobile networks to offload a large number of large-size videos by D2D sharing activities because it is free and fast. From the plots of portion of duplicate traffic (selected content types) in Fig. 1(b), which is computed as one minus the ratio of the total size of non-duplicate traffic load to the total traffic load; we see that approximately up to 40% aggregated traffic is duplicate, which implies that a huge amount of mobile users request the same popular content files via *Xender*, especially videos and APPs (mobile applciations). Therefore, *Xender*-like D2D services are becoming an important and effective method for mobile video dissemination and APP marketing with great capacity for traffic offloading.

Many studies are done for exploiting D2D sharing by the intermittently encounters/meetings among mobile users for the purpose of traffic offloading in MSNs. It is advocated that by selecting an appropriate initial set of *seeds*, the backbone traffic load can be reduced significantly by 20% to 50% [4]. As proved in [5], the content dissemination though a small number of capable users as initial seeds can efficiently satisfy the delay requirements of users, while reducing a substantial amount of backbone traffic. However, there are still several important issues in related research which are not fully elaborated:

- *How to know, to estimate and to predict the dissemination delays of users' preferred content for their satisfaction?* As studied in [5] [6] [14] [15], same dissemination deadline of the same content for all users are assumed; however, users may indeed have various access delay requirements [16].
- *Why mobile users share content with others?* Studies in [3] [5] [6] assume people always exchange content gratuitously. However, in reality, people mostly share information by "word-of-mouth" propagation [19], and realistic social relationship should be exploited.
- *How to design the seeding strategy for minimizing the mobile traffic load while satisfying the delay requirements of all users?* Effective strategies for selecting initial seeds have been discussed in prior studies [3] [4] [17], most of which focus on user mobility but ignore the social impact properties among users.

Considering above issues, we try to exploit the "social" factors of mobile users, and thus discover that there has been a dramatic increase in the number of mobile users who participate in the online Social Network Services (SNSs), e.g., Facebook, Twitter, Sina Weibo, and so on, where more and more content is recommended and spread widely and rapidly [21] [22]. By investigating related measurements and modeling studies of the MSNs and SNSs, we utilize following key findings, which are leveraged for effective content dissemination and traffic offloading:

- Considering the online SNSs, access patterns of users can be always measured, statistically modeled and hence utilized for our purpose. We can analyze the **access delay** between the content generation time and the user access time [23], which is per-user dependent due to people's various life styles [16] [21].
- A user's social influence in online SNSs, or say **spreading impact** to other users, can be modeled based on the analysis of historical activities, i.e., the probability of forwarding to others.
- The mobility patterns of users in offline MSNs can be measured and modeled [5] [14] [24], and hence we can derive different offline **mobility impact** factors for users to disseminate the content to others.
- User relationships and interests in online SNSs show significant **homophily** and **locality** properties (to be detailed in Sec. 2), which is quite similar to those in offline MSNs [19] [25]. That is, users are mostly clustered both by regions and interests, which can be always exploited for traffic offloading.

In this paper, we are motivated to propose a ***T**raffic **O**ffloading framework by **S**NS-Based opportunistic **S**haring* in MSNs, named ***TOSS***. TOSS initially pushes the content objects to appropriately selected seeding users, who will meet and share the content opportunistically with others, by exploiting their spreading impact in the online SNS and their mobility impact in the offline MSN. From modeling and analysis, as well as trace-driven evaluation, TOSS lessens the cellular traffic load significantly (63.8% - 86.5%), while still satisfying the delay requirements of all users.

We organize the rest of the paper as follows. We review the related work in Sec. 2, and then detail the TOSS modeling and optimization framework in Sec. 3. The trace-driven evaluation and analysis are shown in Sec. 4 and Sec. 5, respectively. We further analyze TOSS over a large-scale realistic trace of social D2D sharing in Sec. 6, followed by the conclusion in Sec. 8.

2 Related Work

2.1 Opportunistic Sharing in MSNs

Offloading studies are classified into two categories, delayed and nondelayed offloading. While many of current studies are for nondelayed offloading, in this paper we mostly focus on the delayed one, e.g., DTN-based ones. The study by Zhang et al. [14] has developed a differentiation-based model to analyze the epidemic content delivery delays. And by the similar modeling methodology, Li et al. [6] also have designed an opportunistic content delivery framework in DTNs focusing on energy-efficiency, while the scalability and optimality of content dissemination by exploiting device-to-device contacts has been explored as a social welfare maximization problem studied in [3]. Similarly, [15] has solved the maximization of traffic offloading utility in DTNs as a knapsack problem. However, epidemic delivery generally suffers from slow start and long completion time, and thus strategic pushing schemes have been studied to expedite the dissemination in [5] [4].

Content dissemination by generic epidemic sharing has been hotly accelerated by leveraging users' social relationships in many studies recently. For instance, BUBBLE Rap [26] utilizes the social grouping characteristics for tieing friends. And interesting study in [17] analyzes the social participation for content dissemination in MSNs based on the optimal initial seeds selection. A friendship is reflected by not only a impact but also tags, and thus the study in [27] proposes to assign interest tags to the users and

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2018.2867437, IEEE Transactions on Services Computing

content objects to identify their preferences of content, and then to utilize users local centrality for efficient content spreading. ContentPlace [28] utilizes social central betweenness of mobile users to optimize the mobile content sharing. Furthermore the user encounter history is explored for getting the friendship to disseminate content in the DTNs/MSNs in [29]. Therefore we are also motivated to extend the epidewmic sharing in MSNs by considering the real social relationships in SNSs. Regarding realistic tests, Bao et al. carried experiments in Manhattan and identified the effectiveness of sharing-based offloading, which reduced about 30% to 70% mobile traffic load [30]. Security and privacy problems are important in sharing-based content dissemination, but we cannot cover them in this framework due to the limited space, however, readers may read more in [32] and [33].

The content dissemination in MSNs highly relies on those **infrastructureless** D2D communication techniques, including Bluetooth, Wi-Fi Direct, NFC and etc. For instance, the Airdrop technique in Apple's devices provides convenient local sharing functionality via Wi-Fi connections. Recently the D2D communication underlaying the 3GPP LTE cellular network in the operator authorized spectrum gains lots of popularity [10], which is an effective enabler of services in proximity with capable and efficient D2D content dissemination but limited impact of interference in the primary cellular networks. But D2D-based opportunistic sharing and offloading cannot provide full guarantee on the deliveries, and thus some studies like [7] propose to use effective approach to offload traffic via D2D opportunistic sharing adaptively if the content spreading progress doesn't catch up with the expectation. However, driving the sharing-based traffic offloading by D2D sharing in the market mostly requires incentive-based business models and proper pricing schemes, which can encourage users to share with each others willingly. There are a few representative pricing and incentive studies like Win-Coupon in [34].

2.2 Information/Content Spreading in SNSs

In the real world, "opinion leaders" who are capable of strong social impacts always perform the key roles for spreading information to most of other people, according to the the effect of "word-of-mouth" [19] [35]. Similarly in online SNSs, a small number of users may significantly influence the most of the other users for the spreading of popular information/content, which is studied as the "power-law" effect. For analyzing the propagation of the content from "opinion leaders" to others as well as the re-sharing activities among users, many studies have utilized the probabilistic modeling, which is proved to be effective and convenient [23] [36]. Measurement studies in [23] [35] further identify that the recommendation from famous people, who have potentially strong social impact to others, may accelerate the topic spreading optimally. The propagation of content via people by people necessarily induce delays, and the measurement studies in [23] [21] [16] point out that delays of re-sharing behaviors as well as the spreading impacts are accumulated hop by hop. This delay between the time of generating the content and that of accessing the content mainly depends on the different life styles of people. Based on measurement races, researchers can analyze, and predict the sharing activities and the access delays of SNS users, and thus utilize them for increasing the potential of offloading [21] [22] [23].

In SNSs, user relationships and interests have significant **homophily** and **locality** characteristics as similar to those in MSNs as reported in studies [19] and [25]. The homophily means the online and offline users are mostly clustered both by regions and interests, which is also called "birds-of-a-feather" effect [20], and is regarded as the tendency of individuals to associate and bond with similar others . And obviously people with similar interests intend to share and transfer the related infomation/content with each other. The locality here means that people may have nearly consistent behaviors of accessing the content and sharing with people who are close geographically. Even users in online SNSs may mostly interact with and thus impact others in proximity as pointed in studies [19] and [25]. Note that, users' online and offline characteristics of homophily and locality have been already utilized to facilitate the content delivery in [2] [25], and thus is further leveraged in TOSS framework for D2D-based traffic offloading.

3 The TOSS Framework

In this section, we will describe the modeling part of the TOSS framework, including modeling, online spreading impact, access delay analysis, offline mobility impact and the optimization algorithm, respectively.

3.1 Preliminaries

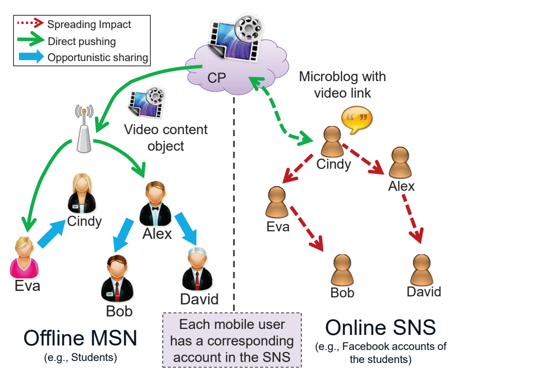


Fig. 2. Illustration of TOSS with two layers: online SNS and offline MSN

As shown in Fig. 2, the TOSS framework entails both an online SNS and an offline MSN. Note that we declare that "online" is for the accounts in the Internet-based virtual world, and "offline" is for the real people in the physical world. Suppose there are total N mobile users, u_i, $i = 1, ..., N$, who have corresponding SNS identities. Because we focus on the content spreading in an online SNS, we use a directional graph to model the SNS, e.g., Twitter, Sina Weibo. TOSS can also work with SNSs based on the bidirectional graph (e.g., Facebook) since it is a subset of the directional graph. The online SNS can thus be represented by, $G(V, E)$, where V is the set of users, and E is the set of directional edges. If u_j follows u_i, u_j is one **follower** of u_i and u_i is one **followee** of u_j. As we focus on the content spreading, the directional edge (represented by an arrow) in

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2018.2867437, IEEE Transactions on Services Computing

Fig. 2 is from u_i to u_j, denoted by v_{ij}. That is, u_i has a direct impact to u_j for content spreading.

We define the home-site, where a user create and shares content in the SNS platform, as the **microblog**, and we define a short message posted by a user containing the content (or link to the content) as a **micropost**, e.g., a tweet in Twitter or a post in Facebook. And then the content file is called a **content object**. Furthermore, we define the **timeline** of a user in online SNS as the serie of all microposts published by a user in his/her microblog, sorted by the publishing time.

At any time, a user may find or create a new interesting article, image, or video, and share it in the SNS as an **initiator**. All his/her followers will then be able to access the content, and some of them will further re-share in their timelines. Making comments will not induce any information spread; thus we only consider the re-share activities. Afterwards, what TOSS seeks to achieve is that, while the micropost with the content is being spread to other users in the online SNS, the content object will be accessed and delivered among user devices in the offline MSN. Note that the TOSS framework is not confined strictly to the dissemination of one popular content to all the users, but applies to general deliveries of any content to a group of potential recipients with any size. That is when we deal with one particular content, we can collect the online and offline social information of those users who will access the content separately from the whole base, and treat them as a one group for applying for TOSS framework. More traffic will be inflowed through SNSs inherently due to the emerging trend of integration of SNSs with Content Providers (CPs). And hence TOSS will benefit, and thus becomes more effective for offloading traffic.

TOSS defines four factors for user u_i: two for the online SNS, (1) the outgoing spreading impact, $I_i^{S\rightarrow}$, and (2) the incoming spreading impact, $I_i^{S\leftarrow}$, which indicate how important the user is for propagating the micropost (to others or from others, respectively); two for the offline MSN, (3) the outgoing mobility impact, $I_i^{M\rightarrow}$, and (4) the incoming mobility impact, $I_i^{M\leftarrow}$, which indicate how important the user is for sharing the content object (to others or from others, respectively) via encounters. We will discuss their calculation in Sec. 3.2 and Sec. 3.4.

Considering the above factors, TOSS seeks to select a proper subset of users as seeds for pushing the content object directly via cellular links, and to exploit the D2D sharing in the offline MSN, while satisfying different access delay requirements of different users. The sharing in TOSS is considered as "prefetching" in advance before users' practically accessing activities. We define a vector $\overrightarrow{p}$ to indicate whether to push the content object to a user via cellular links or not, e.g., $p_i = 1$ means pushing the content object directly to user u_i.

3.2 Spreading Impact in the Online SNS

We extend the previous probabilistic models [35] [36] to quantify the content spreading impact in the SNS. Hereby, we define the $I^{S\rightarrow}$ factor of user u_i to user u_j, denoted by γ_{ij}, $0 \leq \gamma_{ij} \leq 1$, as the ratio of the number of microposts of u_i that u_j accesses and re-shares to the number of all microposts of u_j in u_j's timeline. And thus, γ_{ij} is the probability that u_j will re-share the microposts from u_i.

Based on the SNS graph G, we define U_i^h as the set of h-hop upstream neighbors (followees) of user u_i through all possible shortest h-hop paths without a loop, and likewise D_i^h as that of h-hop downstream neighbors (followers). And we use γ_{ij}^h to denote the $I^{S\rightarrow}$ factor from user u_i to u_j by any h-hop path (inversely γ_{ji}^h as the $I^{S\leftarrow}$ factor from user u_j to u_i). From u_j's point of view over a certain period, we need to consider (1) the number of microposts that u_j has created by himself/herself, c_j, (2) the number of re-shared microposts by u_j from u_i, r_{ij}, and (3) the number of re-shared microposts from all h-hop followees, to calculate $I_i^{S\rightarrow}$ as follows:

$$\gamma_{ij}^1 = \frac{r_{ij}}{c_j + \sum\limits_{u_k \in U_j^1} r_{kj}}, \tag{1}$$

$$\gamma_{ij}^2 = 1 - \prod_{k \in D_i^1 \cap U_j^1} \left(1 - \gamma_{ik}^1 * \gamma_{kj}^1\right), \ldots\ldots \tag{2}$$

$$\gamma_{ij}^h = 1 - \prod_{k \in D_i^{h-1} \cap U_j^1} \left(1 - \gamma_{ik}^{h-1} * \gamma_{kj}^1\right). \tag{3}$$

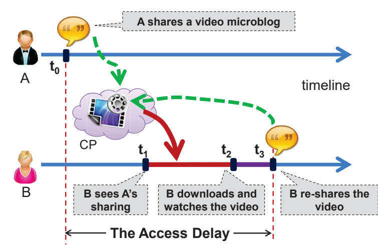
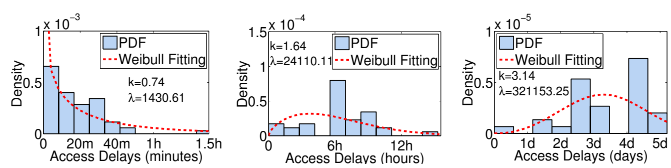


Fig. 3. Illustration of the content access delay



(a) User u_1, Std. Err. of k is 0.078 (b) User u_2, Std. Err. of k is 0.151 (c) User u_3, Std. Err. of k is 0.421

Fig. 4. The access delay distributions of real users with Weibull fitting

We use γ_{ij}^* to denote the impact from user u_i to user u_j via all possible paths with less than or equal to H hops, computed by:

$$\gamma_{ij}^* = 1 - \prod_{n=1}^{H} \left(1 - \gamma_{ij}^n\right), \tag{4}$$

where H is less than or equal to the maximal diameter of the SNS graph G. Then $I_i^{S\rightarrow}$ and $I_i^{S\leftarrow}$ of u_i to and from the whole user base is calculated by, respectively,

$$I_i^{S\rightarrow} = \sum_{j=1}^{N} \gamma_{ij}^*, \quad I_i^{S\leftarrow} = \sum_{j=1}^{N} \gamma_{ji}^*. \tag{5}$$

Note that it is reported in [20] [21] that the average path length in SNSs is normally 4.12 and the spreading impact after 3 hops becomes negligible.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2018.2867437, IEEE Transactions on Services Computing

3.3 Access Delays of Users in the SNS

Access delay between the content generation time and user's access time may be different for each user [16] [21]. As illustrated in Fig. 3, user A creates a microblog for an interesting video in the SNS at t_0. One of A's followers, B, happens to see A's microblog after a certain delay at t_1 due to B's personal business. Once B clicks to play it, a buffering delay is needed until t_2; B will re-share the video at t_3 after watching it. In practice, it is hard to obtain t_1 and t_2 data. So we consider B's access delay as $t_3 - t_0$, which is captured from the SNS measurement trace.

To investigate access delays, we collected the SNS trace data of approximately 2.2 million users from the biggest online SNS in China, Sina Weibo (measurement details will be explained in Sec. 4). The access delay is gathered as the time difference between the generation time of the original microblog and the time of re-sharing by a follower. We pick up four real users from the trace, and plot their access delays by probability distribution function (PDF) in Fig. 4. User u_1 is likely to access the content frequently with short delays. But users u_2 and u_3 have significant delays, on the order of hours and days, respectively.

We use a PDF to model the access delays of each user, say u_i, in terms of the probability to access the content at t, denoted as $A_i\left(t\right)$. As similar to [3], $A_i\left(t\right)$ can be considered as the access utility function, in order to calculate the user satisfaction performance. If the content object is already obtained locally in the user's device when he/she has the highest probability to access the content, he/she will be mostly satisfied.

In order to model the various distributions of access delays with different shapes of PDF curves, we choose to use the Weibull distribution for fitting, which is commonly used for profiling user behaviors in SNSs [37]:

$$A_i\left(t, \beta_i, k_i\right) = \frac{k_i}{\beta_i}\left(\frac{t}{\beta_i}\right)^{k_i-1} e^{-\left(\frac{t}{\beta_i}\right)^{k_i}}, \quad t \geq 0, \tag{6}$$

where the fitting parameters β_i and k_i can identify the access pattern of user u_i (k_i controls the curve shape, and the $beta_i$ value (λ values in Fig. 4is act When $k_i \geq 1$, the Weibull fitting curve can present the distribution of the access delays of keen users; if $k_i < 1$, the Weibull fitting curve has a peak, and thus, can present the distribution of access delays of dull users. It is measured that (to be discussed in Sec. 4) about 2/3 of SNS users are dull ones with large access delays, which is a sufficiently large portion of users that allows TOSS to disseminate the content object by offline opportunistic sharing.

3.4 Mobility Impact in the Offline MSN

It has been studied that mobile users in the offline MSNs (or DTNs), have different mobility patterns [5] [14] [24], and hence different potentials for sharing content. Thus the mobility impact, I^M, is defined to quantify the capability of a mobile user to share a content object with other users via opportunistic meetings, or say **contacts**, while roaming in the MSN. The temporary connectivity with nearby users mostly relies on active discovery mechanisms; thus we assume all mobile users are synchronized with a low duty cycle for probing as proposed by eDiscovery [9] so that phones are aware of each other, or the neighbor information can be also obtained in a centralized manner done by the operator.

Referring to [3] [5] [6] [14] [24] [27], we assume that the inter-contact intervals of any two mobile users follow the exponential distribution. We use λ_{ij} to denote the opportunistic contact rate of user u_i with user u_j. Note that there are many practical methods to measure λ_{ij} values, e.g., centralized measurement by the location management entity in the MNO, or by message exchange during distributed D2D contacts [28]. The contact duration is ignorable in TOSS, because we assume the content delivery is always successfully finished during the contact due to the high bandwidth of local communications [3] [5] [14] [27].

We adopt the epidemic modeling from [6] [14] [5] with the continuous time Markov chain to model the opportunistic sharing in TOSS. For now, we assume that file objects are shared instantly via high-capable device-to-device link when two users contact, and hence the contact duration time is thus not considered. In practical, the D2D links may support up to 10 Mbps transmission speed normally [46] [47] which is enough for most of the files, and we will discuss more in Sec. 7. We let $S_i(t)$ be the probability that user u_i may have the content at t, $0 \leq S_i\left(t\right) \leq 1$, while $1 - S_i\left(t\right)$ is the probability that user u_i has not received the content until t. The increment of $S(t)$ within a period Δt, that is $S_i\left(t+\Delta t\right) - S_i\left(t\right)$, will be calculated as following.

The probability of user u_i to meet user u_j during Δt, is $1 - e^{-\lambda_{ij}\Delta t}$ due to the exponential decay of inter-contact intervals. The probability that user u_i can get the content from another user u_j via opportunistically meeting, denoted by ϵ_{ij}, is calculated by:

$$\varepsilon_{ij} = \left(1 - e^{-\lambda_{ij}\Delta t}\right) \cdot \gamma_{ji}^{*} \cdot S_j\left(t\right), \tag{7}$$

where the $I^{S\rightarrow}$ impact factor from u_j to u_i, γ_{ji}^{*}, is considered as both *(i)* the spreading probability that u_j will re-share microblogs from u_i and *(ii)* the sharing probability that u_i can obtain the content object from u_j.

By summing ϵ_{ij} of u_i from all users, the probably that u_i can get the content from others within Δt is,

$$1 - \prod_{j=1, j\neq i}^{N} \left(1 - \varepsilon_{ij}\right). \tag{8}$$

The increment of the probability that u_i has the content is,

$$S_i\left(t+\Delta t\right) - S_i\left(t\right) = \left(1 - S_i\left(t\right)\right) \cdot \left(1 - \prod_{j=1, j\neq i}^{N} \left(1 - \varepsilon_{ij}\right)\right). \tag{9}$$

Letting $\Delta t \rightarrow 0$, the derivative of $S_i(t)$ will be,

$$\begin{aligned} \dot{S}_i\left(t\right) &= \lim_{\Delta t \rightarrow 0} \frac{S_i(t+\Delta t) - S_i(t)}{\Delta t} \\ &= \left(1 - S_i\left(t\right)\right) \cdot \sum_{j=1, j\neq i}^{N} \lambda_{ij} \cdot \gamma_{ji}^{*} \cdot S_j\left(t\right), \end{aligned} \tag{10}$$

where initially $S_i\left(0\right) = p_i$ from $\overrightarrow{p}$.

Solving the above matrix of the ordinary differential equation system is complicated. However, we can find a numerical solution easily by approximation with power series [38]. Due to the limited space, we skip the details of the procedure for getting numerical solutions.

Given a pushing vector $\overrightarrow{p}$, we can calculate how long it will take for any user u_i to obtain the content by the inverse function of $S_i(t)$ with $S_i(t) = 1$, defined as the *content obtaining delay* of u_i, denoted by t_i^*:

$$t_i^* = S_i^{-1}\left(\left\{\gamma_{ji}^*\right\}, \left\{\lambda_{ij}\right\}, \overrightarrow{p}\right) \quad, \quad j = 1, ..., N, j \neq i, \qquad (11)$$

where $\left\{\gamma_{ji}^*\right\}$ is the series of $I^{S\leftarrow}$ factors from all other users to u_i in the SNS, and $\{\lambda_{ij}\}$ is the series of meeting rates of user u_i to all other users in the MSN. Note that TOSS mainly seeks the optimal $\overrightarrow{p}$ to match the content obtaining delays of all users with their access delay PDFs.

$I_i^{M\rightarrow}$ is actually the same as $I_i^{M\leftarrow}$ since $\lambda_{ij} = \lambda_{ji}$ for any u_i and u_j due to the symmetric nature of contacts. We define the I^M factor for u_i as $I_i^{M\rightarrow} = I_i^{M\leftarrow} = \lambda_i^* = \sum_{j=1}^{N} \lambda_{ij}$. We will only use I^M to denote the mobility impact. We can use approximation methods, e.g., the Newton method, to get the numerical result of the inverse function of $S_i(t)$.

3.5 System Optimization and Heuristic Algorithm

TOSS's main objective (Eq. 12), is to choose proper set of seeds for initial pushing, $\overrightarrow{p}$, by evaluating I^S (both incoming and outgoing) and I^M values of all users, to get the content obtaining delay t^* for each user in order to maximize the sum of the access utilities (access probabilities) for all users, where the number of initial pushing seeds is constrained by C, and we call $\sum A_i(t)$ the total access utility function of the whole user base. This problem is similar to the social welfare maximization problem, discussed in [3]. With power series approximations, we can find the maximum values by general numerical methods. Also we can even tune and find the needed C by given a target total access utility. One of the key remaining future work will be the reduction of the complexity of the equations and thus the optimization problem.

$$\begin{aligned}
\underset{\overrightarrow{p}}{\text{Maximize}} : & \sum_{i=1}^{N} A_i\left(t_i^*, \beta_i, k_i\right) \\
= & \sum_{i=1}^{N} A_i\left(S_i{}^{-1}\left(\left\{\gamma_{ji}^*\right\}, \left\{\lambda_{ij}\right\}, \overrightarrow{p}\right), \beta_i, k_i\right) \\
& (j = 1, ..., N, j \neq i) \\
\text{Subject to} : & |\overrightarrow{p}| \leq C \quad ,
\end{aligned} \qquad (12)$$

We design a heuristic algorithm to find the near-optimal solution $\overrightarrow{p}$ for maximizing $\sum A_i(t)$ numerically, based on the popular hill-climbing method, as shown in Algorithm 1. Initially, it selects the top C users from all users sorted by I^M in descending order ($I^{S\rightarrow}$ or $I^{S\leftarrow}$ works similarly) as the first input of the iteration loop. The assignment of value 1 means selected as the pushing seed, and 0 means not pushing at the beginning. Then the algorithm tries to iteratively exchange the p_i and p_j values of any two users u_i and u_j for a new pushing vector, and calculate the $\sum A_i(t)$ by the new pushing vector, to check whether a larger value of $\sum A_i(t)$ can be achieved; if so, the values of p_i and p_j can be retained. The iteration repeats infinitely until the increment of $\sum A_i(t)$ is smaller than a specified threshold. The algorithm is with complexity around $O(M \cdot N^2)$, where M is the number of iterations. By default, we set M as 30, and the threshold is 0.01, but these two factors are managed regarding the balance between the time cost and the result accuracy under different scenarios and requirements. Note that the above modeling and the heuristic algorithm are carried out in MATLAB.

Algorithm 1 A Hill-climbing algorithm to seek near-optimal initial pushing seeds

```
// Initializing p⃗
for all i = 1 → N do
    p_i=0; v_i = λ_i^*, γ_i^*, or random;
end for
Sort v_i by Descent Order (↓);
for all i = 1 → C do
    p_i=1;
end for
A_sum = Σ_{i=1}^{N} A_i (S_i^{-1} ({γ_ji^*}, {λ_ij}, p_i), β_i, k_i),
        (j = 1...N, j ≠ i);
// Hill-Climbing
repeat
    flag=true;
    for all i = 1 → N do
        for all j = i + 1 → N do
            if (flag==true) AND (p_i + p_j == 1) then
                Exchange(p_i, p_j);
                A'_sum = Σ_{i=1}^{N} A_i (S_i^{-1} (γ⃗_i^*, λ⃗_i, p_i), β_i, k_i),
                        (j = 1...N, j ≠ i);
                if A'_sum > A_sum then
                    δ = A'_sum − A_sum; A_sum = A'_sum;
                    flag=false;
                end if
            end if
        end for
    end for
until δ < Threshold
return A_sum, p⃗
```

4 Analysis of SNS and MSN Trace Results

In this section, we will carry out realistic measurement and analysis over small-scale traces to evaluate the factors in the TOSS framework.

To evaluate the effectiveness of the TOSS framework, we need SNS trace data to quantify the spreading impact factors and access delays, as well as MSN trace data to analyze the mobility impact. However, there is no publicly available trace data that contains both the SNS and the MSN activities. Although there is geographical feature for Weibo posts, after we measured the traces, we discovered that the geographical information of Weibo posts is quite discrete, not continuous, which brings difficulty to identify the mobility impact of the users. Also a portion of posts don't have geographical information. However, rich data of users trajectories is needed for modeling the mobility accurately, and thus we can discover the opportunities for D2D sharing while users are moving. Therefore, the integrated geographical information of the Weibo posts cannot be used unluckily, and thus we have to use the integration method for the online and offline traces. Thus, we will choose to take separate measurements, and combine them by certain mapping schemes as explained in Sec. 5.1.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2018.2867437, IEEE Transactions on Services Computing

We select the most popular online SNS in China, Sina Weibo, and keep track of 2,223,294 users for four weeks during July, 2012. We collected 37,267,512 microposts generated (and partially re-shared) by the users, and further obtained the list of all the re-sharing activities for each micropost. We implemented the data collection software, which starts from 15 famous users of distributing popular video clips, and expands the user base from their followers. Capturing the next hop followers is carried out iteratively. The captured data includes details of owner's account profile, all microposts with timestamps of the owner, all comments and reposts with timestamps, as well as the profile of the users that make comments and reposts to the owner.

4.1 Online SNS - Spreading Impact, γ_{ij} and I^S

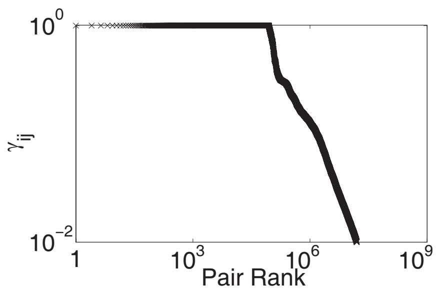
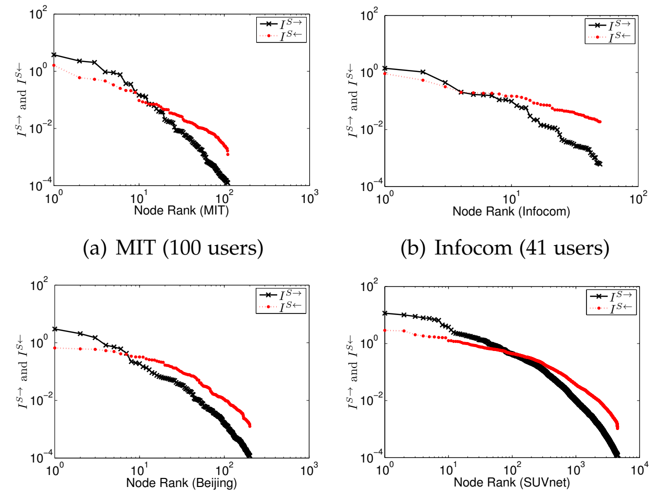


Fig. 5. Analysis of γ_{ij}

Recall that I^S is the overall spreading impact of the user to all users in the SNS, calculated by Eq. (5). However, calculating I^S for the whole user base takes substantially long time. Thus, we analyze the sub-graphs of corresponding number of users from the whole social graph by random walking according to the scale of the mobility traces (to be detailed in Sec. 4.3). We consider up to 4-hop paths ($H = 4$) among the users in the graphs as suggested in [21].

There have been some related measurement studies in [21] [39] and [40] pointing out that: the SNS is a **scale-free** complex network, in which node strength distribution follows the power-law, at least asymptotically. That is a small number of nodes make dominant impact to the network, while many nodes make very small impact, if we consider the node degree or the spreading impact (re-sharing ratio) as the strength of a node to the network [39] and [40]. So due to the nature characteristics of scale-free complex networks, sub-graphs from the whole network graph (with not too small size) by the random walking method can still obtain similar power-law characteristics (power-law-like distribution of node strength).

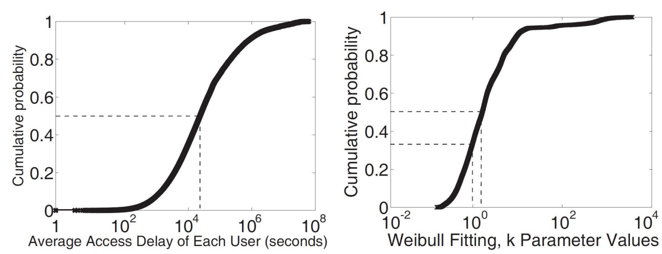
We then check the sub-graphs that we abstract from the online SNS graph with the sizes corresponding to the mobility traces (to be detailed in Sec. 4.3), and for each trace we abstract sub-graphs for five times, and then make average value. We draw the log-log plots for $I^{S\rightarrow}$ and $I^{S\leftarrow}$ of the nodes from the sampled sub-graphs as shown in Fig. 6. We can see that a smaller number of people have significant outgoing impact ($I^{S\rightarrow}$) to the whole SNS, while many users have very small impact. Also we see that many users are more likely to be impacted rather than impacting others ($I^{S\rightarrow} < I^{S\leftarrow}$). All of the figures are able to reflect the asymptotical power-law trend. So conclusively, all of the



(a) MIT (100 users) (b) Infocom (41 users)

(c) Beijing (182 users) (d) SUVnet (4,311 users)

Fig. 6. Measurement values of I^S for sub-graphs sampled from the SNS graph with different sizes corresponding to the mobility traces



(a) Average access delays (b) Weibull parameter, k; Avg. Std. Err. of k values is 172.42.

Fig. 7. Access delays and fitting parameters

sub-graphs with different sizes can still represent the SNS characteristics, and it will be an acceptable methodology to map the SNS sub-graphs to the mobility traces. Note that in the following part, the online spreading impact factor is normalized and then applied.

4.2 Online SNS - Access Delay of u_i, $A_i(t)$

Measurement results of the access delays on the whole user base are shown in Fig. 7. From the cumulative distribution function (CDF) of the average of all the access delays of each user in Fig. 7(a), half of the users have the average access delay larger than 23,880 seconds, which is about 6 hours and 38 minutes. Taking a closer look, we find (1) 3.67% of users have the average access delay small than 10 minutes, (2) 20.38% of users have the delay smaller than 1 hour, and (3) 26.79% of users access the SNS with average delay larger than 1 day. Furthermore, we calculate the Weibull fitting parameters of all users, and the CDF of the shape parameter k of all users is shown in Fig. 7(b). which indicates that 32.63% of users have $k < 1$, who are keen users, while 67.37% of users can be classified as dull users. Therefore, we verify that a large portion of users access the SNSs with sufficiently large delays, which TOSS can exploit to disseminate content by offline opportunistic sharing.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2018.2867437, IEEE Transactions on Services Computing

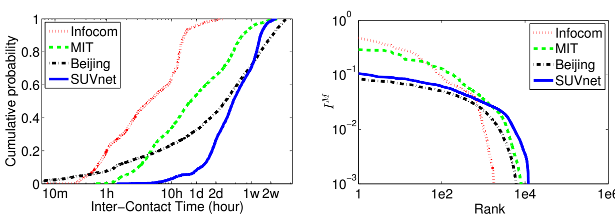
TABLE 1
Mobility Traces (Bl. is for Bluetooth)

Trace	Link	Users	Days	Contacts	Avg.λ
MIT [41]	Bl.	100	246	54,667	0.01532
Infocom [42]	Bl.	41	4	22,459	0.14167
Beijing [43]	/	182	150	8,894	0.00023
SUVnet [44]	/	4,311	30	169,762	0.00131

4.3 Offline MSNs - λ_{ij} and I^M

We choose four mobility traces, MIT [41], Infocom [42], Beijing [43], and SUVnet [44], in order to evaluate the performance of TOSS. These traces record either direct contacts among users carrying mobile devices or GPS-coordinates of each user's mobile route, and traces details are shown in Table 1. The four traces differ in their scales, durations, and mobility patterns; The MIT and the Infocom traces are collected by normal people, but the Bejing and the SUVnet traces are collected by vehicles. Beijing and the SUVnet traces have no contact records, but only GPS coordinates by time; we assume a contact once two users are within 20 meters during 20s.

We analyze the traces and obtain the inter-contact intervals of all user pairs, as shown in Fig. 8(a). The Infocom trace has the highest contact rate because users are at a conference spot, and thus have high contact rates. The MIT trace also has high contact rates since users are friends within the campus. The Beijing and the SUVnet traces have large inter-contact intervals because they have relatively low frequency of GPS records and large user base. I^M values of all users of the traces (values smaller than 0.001 are ignored) are plotted in Fig. 8(b), which indicates the similar trends of the traces as discussed above. Users in the Infocom trace have the highest potentials to obtain the content by sharing, but users in the Beijing trace have the weakest potentials.



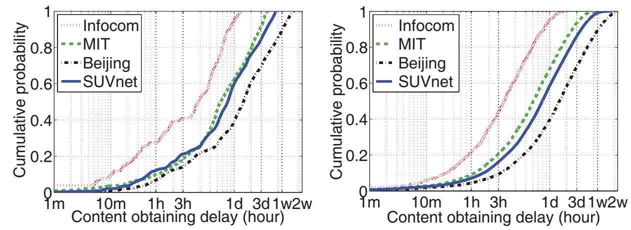
(a) Analysis of $1/\lambda_{ij}$ (b) Analysis of I^M

Fig. 8. λ_{ij} and I^M values

4.4 Content Obtaining Delays, t_i^*

We investigate the content obtaining delays, t^*, of all users by just 1 random initial pushing (averaging 20 runs with different random seeds) for the four traces. And then we put λ values of all pairs extracted from the traces into the proposed model in Sec. 3.4. From the CDFs in Fig. 9(a), the Infocom trace has the smallest obtaining delays mostly within 1 day; the Beijing trace has the longest delays even up to 10 days. The model with practical λ values in Fig. 9(b) shows the similar performance to the real traces.

In order to precisely verify the accuracy of our modeling to the real traces, from the two figures, Fig. 9(a) for the



(a) Real Traces (b) Trace-based Mobility Modeling

Fig. 9. Content obtaining delays by 1 random pushing with modeling

TABLE 2
Correlation regression analysis between the traces and modeling

Trace	Pearson Correlation	Significance
MIT [41]	0.973	0.000
Infocom [42]	0.979	0.000
Beijing [43]	0.976	0.000
SUVnet [44]	0.968	0.000

real traces, and Fig. 9(b) for the modeling, we carry out the bivariate correlation regression analysis on them, in order to get the Pearson correlation coefficients, by using SPSS. As shown in Table. 2, the results of the correlation coefficients between the real traces and by the modeling are in the range of 0.973 to 0.979, which means the simulation and modeling can fit perfectly with a sufficiently high accuracy.

5 Performance Evaluation

In this section, we evaluate the TOSS framework on how the spreading and mobility impact factors (I^S and I^M) affect the total access utility function ($\sum A_i(t)$).

5.1 How Pushing Vector Impacts Total Access Utility

Due to the lack of a trace that contains the activities of the same users in both online SNSs and offline MSNs, we consider three schemes for mapping SNS users to MSN users in each of the four mobility traces: (1) **random**: SNS users are randomly mapped to MSN users; (2) **h-h**: both SNS and MSN users are sorted in descending order of $I^{S\rightarrow}$ and I^M, respectively, and then are mapped correspondingly; (3) **h-l**: both users are sorted as similar to **h-h**, but an SNS user with high $I^{S\rightarrow}$ is mapped to an MSN user with low I^M. As discussed in Sec. 4.1, since the number of SNS users is much larger than that of MSN users in each trace, we pick accounts from the sub-graphs of the SNS by the random walking method to match the number of MSN users in each trace. Note that while assigning the SNS accounts to the MSN users, the corresponding access delay patterns of the SNS accounts will be retained.

Regarding the methodology of mapping a sub-graph of online SNS by random-walk sampling to the offline MSN graph, we carry out following discussion: It is already studied that when we consider the mobility impact (meeting rate) of two users as their vector strength, and the overal mobility impact of one user (sum of all mobility impact to all other users) as the node strength, the MSN is also classified as a scale-free network [42] [24]. That is in the MSN, a small

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2018.2867437, IEEE Transactions on Services Computing

number of users are always moving quickly and meet many components, while many of the users are relatively stable to meet limited number of other users.

So regarding each mobility trace with different amount of mobile users, as we discussed in Sec. 4.1, we take random-walking-based sampling to obtain the subgraphs from the SNS trace with corresponding number of user accounts, and then map one SNS account to one mobile user by above mapping choices. Note that the online spreading impact factor is normalized and then applied. So this is a reasonable methodology to map between online and offline traces in the case of lacking such a trace with both information. To seek or carry out such a measurement study to track both the online SNS activities and offline MSN activities for a group of people is one important future work.

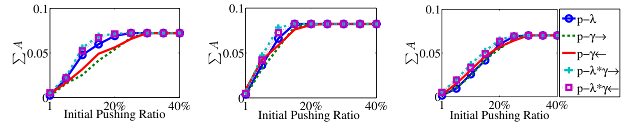
To select the users who will be initial seeds, $\overrightarrow{p}$, constrained by the allowed total number of seeds, C, we consider the following five pushing strategies based on the impact factors:

- **p-**λ: we sort users by I^M ($\sum \lambda_i^*$) in a descending order and choose the top C ones (similar to [3]);
- **p-**$\gamma^{\rightarrow}$: we sort users by $I^{S\rightarrow}$ ($\sum \gamma_{ij}^*$) in a descending order and choose the top C ones (similar to [26] [28]);
- **p-**$\gamma^{\leftarrow}$: we sort users by $I^{S\leftarrow}$ ($\sum \gamma_{ji}^*$) in a descending order and choose the top C ones;
- **p-**$\lambda * \gamma^{\rightarrow}$: we sort users by $I^M * I^{S\rightarrow}$ conjunctively in a descending order and choose the top C ones;
- **p-**$\lambda * \gamma^{\leftarrow}$: we sort users by $I^M * I^{S\leftarrow}$ conjunctively in a descending order and choose the top C ones.

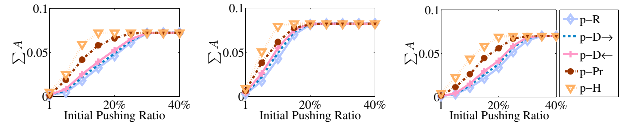
Moreover, there are many **viral marketing** methods to evaluate a SNS user's strength regarding information spreading, for example we can easily qualify by node degree including outgoing degree (number of followees) and incoming degree (number of followers). Note that here the arrow direction is the "following/followed" relationship, reverse to the spreading direction. Furthermore, the PageRank algorithm [45] is also comprehensively used for SNS analysis, which is a link analysis algorithm of Google by assigning a numerical weighting to each element of a hyperlinked set of nodes, with the purpose of "measuring" its relative importance. We apply the general PageRank algorithm on the selected SNS sub-graphs and obtain the PageRank scores. We also consider a random pushing and the heuristic algorithm, and hence we have five more initial pushing strategies based on the graphs:

- **p-R**: we randomly choose C users;
- **p-**$D^{\rightarrow}$: we sort users by outgoing node degree in a descending order and choose C users;
- **p-**$D^{\leftarrow}$: we sort users by incoming node degree in a descending order and choose C users;
- **p-**Pr: we sort users by PageRank score in a descending order and choose top C users;
- **p-H**: we run the hill-climbing heuristic algorithm to obtain the near-optimal pushing vector.

We investigate how $\overrightarrow{p}$ under the 10 pushing strategies impacts the total access utility of all users, $\sum A_i(t)$, with only the MIT trace as shown in Fig. 10, and we skip to show the results of other traces since they show very similar trends. The percentage in the figures is the pushing ratio of C to the number of involved users in each trace. We can



(a) **random - strategies by γ and λ** (b) **h-h - strategies by γ and λ** (c) **h-l - strategies by γ and λ**



(d) **random - strategies by graph** (e) **h-h - strategies by graph** (f) **h-l - strategies by graph**

Fig. 10. As C increases, $\sum A_i(t)$ converges - 3 mappting schemes, 10 pushing strategies - MIT trace as an example

see that as the number of initial seeds increases, $\sum A_i(t)$ increases and converges to the maximum. In all cases **p-H** converges to the maximum the fastest, while **p-**$\lambda * \gamma^{\rightarrow}$ and **p-**$\lambda * \gamma^{\leftarrow}$ as well as **p-**Pr perform very well. **p-R** always performs the worst, but **p-**$D^{\rightarrow}$ and **p-**$D^{\leftarrow}$ also performs poorly. Note that the maximal value of $\sum A$ is capped in different mapping schemes, which means the total user satisfaction is determined by the scenario user nature. The results of different mapping schemes show marginal differences, because TOSS always chooses the users with strong impact strength, and also the access delays provide large a space for sharing.

Although we lack of realistic traces with both online SNS and offline MSN information, the **h-h**, **h-l**, and **random** represents three different mapping schemes with no significant different performance, and hence in following parts, we use the average initial pushing ratio (as well as $\sum A_i(t)$) across the three to reflect general performance under various user parameters in SNSs and MSNs.

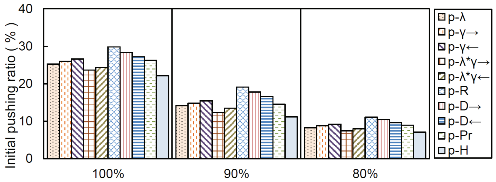
5.2 Satisfying 100%, 90%, and 80% of Users

Recall that the access utility function of u_i is $A_i(t)$. A user is **satisfied**, if he/she can obtain the content when her/his access probability ($A_i(t)$) approaches its maximum (in the fitting Weibull pdf). If we aim to make 100% of users obtain the content by initial pushing and sharing, substantially large delays may take place for certain users (e.g., a user with low γ and λ values). Therefore, we investigate what percentage of users (initial pushing ratio) should be initial seeds to satisfy the access delay requirements of 100%, 90%, and 80% of users depending on different pushing strategies.

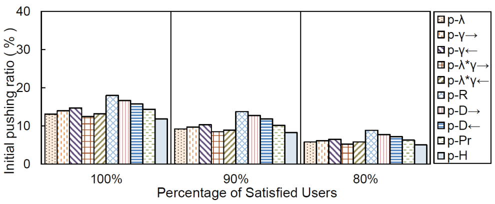
From Sec. 3.5 and Fig. 10, $\sum A_i(t)$ is an increasing function of C (i.e., $|\overrightarrow{p}|$), and the number of satisfied user is also an increasing function of C. The C value that makes $\sum A_i(t)$ approach its maximum will be the standard number of initial pushing seeds for satisfying 100% of user. We examine how C is reduced (for higher offloading gains) if we target the satisfaction of 90% and 80% of users.

From Fig. 11, to satisfy 100% of all users, **p-H** always finds the best initial pushing vector (i.e., the least number of seeds), and **p-R** performs the poorest, while **p-**$D^{\rightarrow}$ and **p-**$D^{\leftarrow}$ also performs poorly, so simply pushing by node degree is not that preferred. In most cases, **p-**$\lambda * \gamma^{\rightarrow}$ and **p-**$\lambda * \gamma^{\leftarrow}$ perform the second best, which implies that we

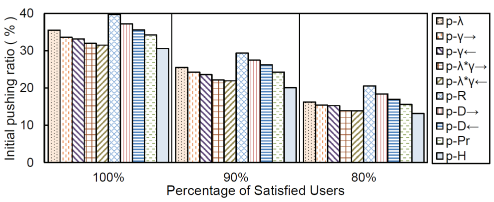
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2018.2867437, IEEE Transactions on Services Computing



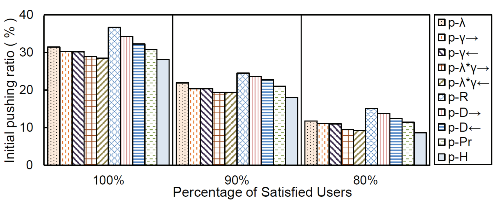
(a) MIT



(b) Infocom



(c) Beijing



(d) SUVnet

Fig. 11. Initial pushing ratios to satisfy 100%, 90%, and 80% of all users

can conjunctively consider the I^S and I^M factor by simple multiplication to achieve near-optimal performance. **p-**Pr achieves not so good performance compared with strategies by impact factors, as it focuses the connections of the network graph but ignores the historical spreading impact, while our proposed factors (γ) make better sense. In MIT and Infocom traces, λ-based strategies perform better than γ-base ones, which means the mobility factor decides more on the sharing process when nodes are with high mobility. In Beijing and SUVnet traces, γ-base ones perform better, which means the social factor controls more when nodes are with low mobility. Note that the Infocom trace always has the best performance; only 13.5% initial pushing ratio can satisfy all users by the **p-H**.

When we target to satisfy 90% of all users, the required initial pushing ratio is reduced significantly. With simple pushing strategies, for the MIT and the Infocom traces, only 15.4% and 10.5% of users need to be the initial seeds on average. The number of initial seeds is further dramatically reduced, when satisfying 80% of users. Approximately 10% initial pushing ratio is needed for all traces except the Beijing trace, which requires about 17% initial pushing ratio. The Beijing and SUVnet traces always need relatively larger number of initial seeds due to their low contact rates and large user bases. The implication from Fig. 11 is that, when the users have relatively higher mobility patterns, the mobility impact will mostly decide the content obtaining delays, but when people are not moving and meeting frequently within a large user base, the online spreading impact needs to be enhanced for initial pushing to offload traffic effectively. Also some worse-case users bring ineffectiveness for opportunistic sharing, but it may be better to push the content to them in the beginning, if they have keen access delay requirement, or it will be better to let them to carry out on-demand fetching when they approach the peaks of their access delay PDF. Generally, **p-H** is about 15-24% better than **p-R**, and 12-16% better than **p-**λ and **p-**γ, and the multiplication of **p-**λ and **p-**γ will be quite a good solution in practical. It is a balance between performance and complexity, and the implication is that, if we focus on the best performance, we can run the heuristic algorithm; if we want a balance between complexity and performance, we can evaluate user online spreading impact and offline mobility impact, and choose proper strategy (e.g., for offloading. **p-R** can still offload certain amount of traffic, which indicates that the sharing-based offloading can work very well in practical actually, because this is mainly due to the potential of the user access delays as discussed in Sec. 3.3.

TABLE 3
Percentage (%) of Traffic Reduction With On-Demand Delivery - Average of 9 Simple Pushing Strategies / Heuristic

Trace	100%	90%	80%
MIT [41]	73.6 / 76.3	**74.6 / 76.9**	70.9 / 72.2
Infocom [42]	**85.3 / 86.5**	79.5 / 80.4	73.4 / 74.1
Beijing [43]	65.3 / 68.4	**65.0 / 68.9**	63.8 / 65.2
SUVnet [44]	68.5 / 70.3	**68.7 / 71.0**	68.3 /70.7

5.3 On-Demand Delivery

If a user who has not obtained the content (by initial pushing or sharing) until he/she actually accesses it, we have to deliver it over a cellular link, which is called on-demand delivery. Also for instance, as our proposal is based on opportunistic modeling, and there is the possibility that the content spreading in realistic scenarios doesn't match with the expectation, and thus the system must trigger the on-demand deliveries correspondingly to make sure of the quality of service, i.e., users' delays to obtain content. Then the cellular traffic of the content delivered on-demand is not offloaded. We now compare the three target percentages of satisfied users (investigated above) in terms of total offloaded traffic. Table 3 shows how much traffic is offloaded from cellular links for the three cases, where the offloaded traffic ratios of the nine pushing strategies are averaged, which are followed with the results of **p-H** strategy after "/". Note that boldfaced numbers are the highest amount of traffic reduction for each trace across the three target satisfaction cases (i.e., 100%, 90% and 80%). When lowering the percentage of satisfied users from 100% to 90% and to 80%, although the initial pushing ratios become reduced, in some cases, the on-demand delivery for the remaining 10% and 20% of users may induce the increment of the total cellular traffic instead. In the MIT, Beijing and SUVnet traces, initial pushing for satisfying the 90% of users plus on-demand delivery for the 10% of users actually reduces the cellular traffic the most.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2018.2867437, IEEE Transactions on Services Computing

6 Analysis of Social D2D Sharing Traces

The above evaluation over the four traces has proved the effectiveness of TOSS in single small-scale scenarios. For carrying out a much realistic investigation, in this section, we further utilize a large-scale social D2D sharing trace from *Xender*, which has a large user base and a high diversity of formations of users, activities and mobility properties. *Xender* has been one of the most popular D2D sharing application services in China and India (as well as many countries in Asia and South America), and it provides users with the convenience of sharing various types of content files without using 3G/4G cellular network, across a large diversity of system platforms, at a speed around several MBps, mainly based on Wi-Fi tethering technique. During February 2016, *Xender* has served totally around 9 million daily and 100 million monthly active users, as well as notably 110 million daily content sharing interactivities [31], while the service is still attracting a substantially increasing number of users for dissemination content by D2D sharing. We then test TOSS over *Xender*'s large-scale offline trace by integrating with Sina Weibo online trace appropriately.

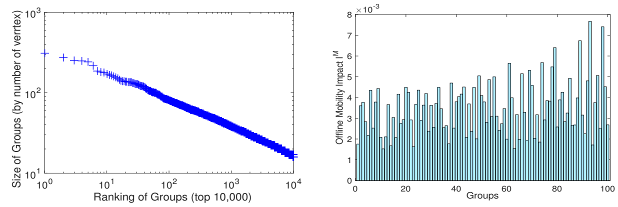
6.1 Description of *Xender*'s Trace

We capture *Xender*'s trace for four weeks from 01/02/2016 to 28/02/2016. Since around 70% users are from India [31], we are motivated to concentrate on analyzing Indian users' activities only. After cleaning invalid and uncompleted entries, the target data set includes 30,485,335 users, 4,434,440,043 transmission interactivity timestamps conveying 16,785,175 content files, with the data entry format: $<$**sender; receiver; content**$<$**MD5, size**$>$**; GPS; timestampi**$>$. To process our large-scale data (with total size of 843 GB) efficiently and reliably, we use Python 2.7 along with Anaconda scientific package and graph-tool library (v2.18), on a cluster of four Dell PowerEdge R730 servers, each of which has two E2630v3 CPUs, 32 cores, 64G RAM and 16T SAS hard disk.

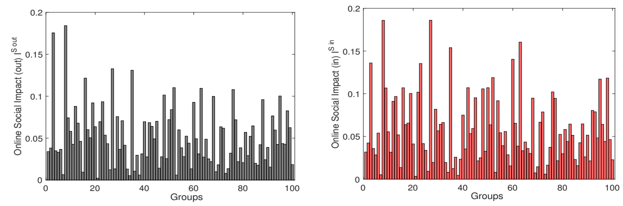
6.2 Integrating *Xender*'s Trace with Sina Weibo Data

Directly mapping equivalent amount of offline users with the whole user base of online users (2,223,294 Sina Weibo users) induces huge amount calculation workload, which is beyond the purpose of this paper. Because the social graph composed by *Xender*'s user interactivities is not a densely connected one, we first discover groups (i.e., connected components of the graph) and then choose the top 100 largest groups from the 883,772 groups in total with respect to the number of vertices, i.e., involved users, as representative groups for experiments. The largest group has 309 vertices (users), and there are 11,948 users in the 100 selected groups. As shown in Fig. 12(a), the sizes of groups follows nearly a perfect power law effect, which indicates in practice, there exists a small number of large groups with many users, but most of the groups (long tail) have small number of users.

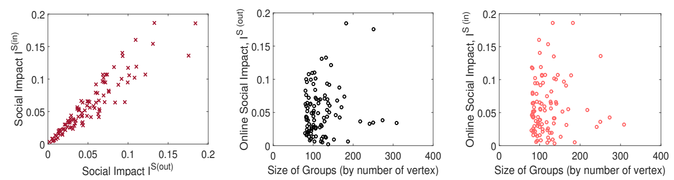
The evaluation of the average offline mobility impact (I^M) of users in each group is shown in Fig. 12(b), while the ranking is based on the group size by descending order, corresponding to the group ranking in Fig. 12(a). Although it seems that there is a trend that smaller groups may have roughly larger I^M, we cannot confirm this effect but will take in-depth investigation for clues and reasons as a future work. Similarly to the evaluation procedure in Sec. 5, we map the online Sina Weibo user dataset with the selected 100 *Xender*'s groups according to aforementioned 3 schemes (**h-h**, **h-l**, **random**) and we make sure of the connectivity of each graph of online users obtained by random walking corresponding to the sizes of offline *Xender*'s groups, while all SNS graphs are non-overlapped, with no common vertex. Now, groups have different scales (sizes), offline mobility impact factors, and various combinations of online impact (outgoing and incoming) factors, and thus our experiments can reflect TOSS's performance over *Xender*'s trace. Fig. 12(d) and Fig. 12(c) show the averaged online social incoming and outgoing impact of each group, while the ranking is still corresponding to the ranking in Fig. 12(b).



(a) Group Size by Vertex (top 10,000) (b) Offline Mobility Impact (top 100 *Xender* groups)



(c) Online Social Impact - Incoming (of the groups of SNS users corresponding to the top 100 *Xender* groups) (d) Online Social Impact - Outgoing (of the groups of SNS users corresponding to the top 100 *Xender* groups)



(e) Correlation between $I^{S\rightarrow}$ and $I^{S\leftarrow}$ of Each Group (f) Correlation between Group Size and Averaged $I^{S\rightarrow}$ (g) Correlation between Group Size and Averaged $I^{S\leftarrow}$

Fig. 12. Properties of *Xender* Groups

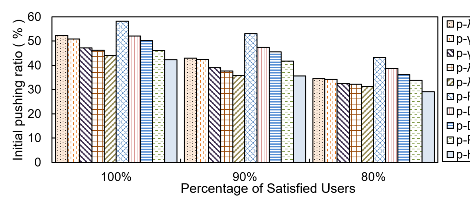


Fig. 13. Initial pushing ratios to satisfy 100%, 90%, and 80% of all users - *Xender*'s trace

6.3 Offloading Performance Evaluation

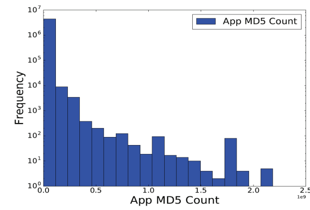
By applying the ten pushing strategies (i.e., **p-**λ, **p-**$\gamma^{\rightarrow}$, **p-**$\gamma^{\leftarrow}$, **p-**$\lambda * \gamma^{\rightarrow}$, **p-**$\lambda * \gamma^{\leftarrow}$, **p-R**, **p-**$D^{\rightarrow}$, **p-**$D^{\leftarrow}$, **p-**$Pr$, and

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2018.2867437, IEEE Transactions on Services Computing

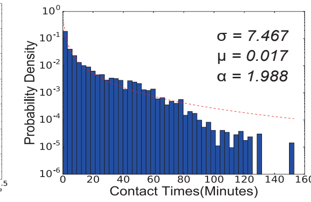
p-H), we evaluate the initial pushing ratio over the 100 groups for satisfying the 80%, 90% and 100% of all the users (considering all users as a whole user base), while the results are averaged across the three mapping schemes (**randomly, h-h, h-l**). As shown in Fig. 13, initial pushing ratio for *Xender* trace is much larger than previous four traces, especially for 100% satisfaction case. Even using the heuristic strategy, **p-H**, around 20.1%, 33.6% and 42.3% initial pushing ratios are needed for satisfying 80%, 90%, and 100% users in all groups, respectively. Because there are various types of users and groups in *Xender*'s trace, while the groups are also not densely connected compared to Infocom and MIT traces that are measured from restricted spots of users with frequent meetings, more initial pushing ratios are needed. Note that **p-**$\lambda * \gamma^{\leftarrow}$ performs slightly better than **p-**$\lambda * \gamma^{\rightarrow}$, which provides the same implication as evaluation results in Fig. 11(c) and Fig. 11(d) that scenarios with low mobility impact may benefit from the strategy emphasizing more on incoming social impacts.

Furthermore, by considering the on-demand pushing to fill up the remaining users, the average ratios of final traffic reduction are, 52.4%, 47.3%, and 49.8%, for 80%, 90%, and 100% satisfaction cases, respectively. With **p-H** pushing strategy, TOSS may achieve 59.9%, 56.4%, and 57.7% final traffic reduction in 80%, 90% and 100% satisfaction cases, respectively, which is worse than previous evaluation over the four traces, but still soundful to be able to half the mobile traffic load by D2D sharing.

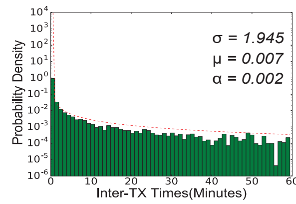
Due to the locality nature of human-beings, TOSS framework still performs well even facing to the scenarios with a very large user base; although people move and travel sometimes, they still meet most of friends in most cases, which is the **clustered effect** for a group of users, which will not be impacted by the whole user base. In another word, people are constrained by our life style and location due to the inherent nature of **locality**.



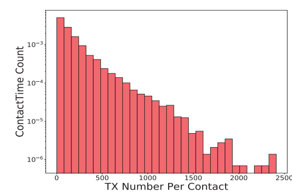
(a) Probability distribution of file sizes



(b) Probability distribution of contact durations



(c) Probability distribution of inter-TX times within the contact



(d) Probability distribution of transmissions per contact

Fig. 14. The size of files and durations of contacts

7 Contact Durations and File Sizes

In this section, we try to elaborate the possibility of D2D sharing considering the potential terminations of encounters. In previous discussions of the TOSS framework, we have assumed that as long as two users' encounters, they can "instantly" share the content file, which is actually not true. In the real scenarios, the transmissions may be interrupted if two people have short encounter duration (contact duration) or if the link quality is not good enough to support the high bandwidth for transmitting large files. Therefore, it is important to verify the disconnections of the D2D sharing activities.

However, as far as we know, there is no any D2D trace for tracking whether a transmission is done or not, and even in the Xender trace, there is no related log for the transmission terminations. Therefore, we carry out an estimation based on the attracted information of file size distribution, contact-time distribution, and some measurement results of D2D bandwidth. From related studies [46] and [47], the Wi-Fi-based transmission among devices can achieve a throughput from 7 Mbps to 10 Mbps, within the distance of 20 metres. and therefore we assume an averaged typical value as 8.5Mbps for estimation. In Xender trace, we collect the file size distribution as shown in Fig. 14(a) and collect the contact duration distribution in Fig. 14(b). Furthermore, in the trace, during each contact, users may intend to share more files and thus there may be more transmissions. So we plot the distribution of inter-transmission times (i.e., the interval among transmissions within one contact) as shown in Fig. 14(c), and the distribution of the number of transmission per contact in Fig. 14(d) which proves the meeting time has great potential of sharing not only one but also more files.

From the traces and figures, we discover that: the average file size is around 16 MBytes, and the average contact time is around 493s, which indicates that on average, the aforementioned bandwidth can easily support the completion of the file sharing (i.e., 16 MBytes * 8 / 8.5 Mbps = 15s, which is much smaller than 493s). However, we cannot justify by only average values, so we carry out more detailed comparison based on the statistics in the Table 4. Sampled file size at particular percentile is estimated for checking whether it can be transmitted completely in the involved encounters (contacts). In most cases, the contact duration is with larger value than the required time. That is, from the Xender's services, the practical D2D transmission may take place by just using a small portion of the contact duration. And even they share multiple files during one contact, as shown in Fig. 14(d). The most of the contacts have only one transmission, while there are still a large number of contacts with several transmissions.

Practically, when two people share some large files, in most cases they need to request and get admissions, and thus they have an agreement of making the transmission done. Because of the satisfiable throughput of D2D, they don't have to terminate, but can still share more files at one time. Therefore, the assumption in Sec. 3.5 can be supported.

For evaluating TOSS on real traces with more consideration on the termination of D2D sharing, we apply one "disrupting factor", notated by pi, as the probability of successful transmission. The Eq. 13 is further improved into:

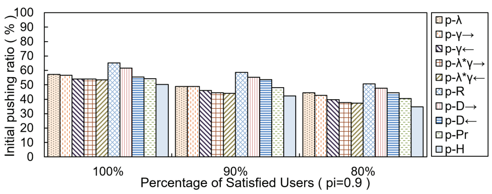
$$\varepsilon_{ij} = \left(1 - e^{-\lambda_{ij}\Delta t}\right) \cdot \gamma_{ji}^{*} \cdot S_j\left(t\right) * \pi, \tag{13}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2018.2867437, IEEE Transactions on Services Computing

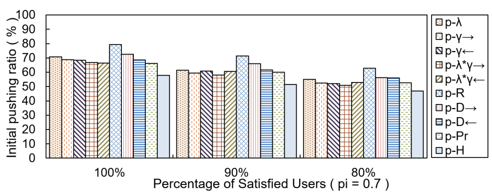
TABLE 4
Statistics of file sizes and contact durations

Items	File Size (MBytes)	Estimated TX Time (Size*8/8.5 seconds)	Contact (seconds)
Average	15.97	15.0	493
10%	0.14	0.13	10
30%	3.74	3.52	43
50%	6.33	5.96	128
70%	16.81	15.82	393
90%	44.42	41.81	1512

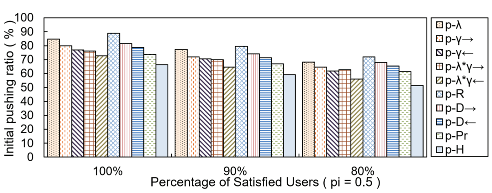
Since we have evaluated the cases when $\pi = 1$, we further check the performance when $\pi = 0.5, 0.7 and 0.9$ regarded as the successful ratios of D2D sharing activities. The same test settings in last section are used, and it can be seen that the termination can significantly impact the offloading ratio. When the probability of successful transmission is 0.9, we need a bit more initial pushing; but when there are more terminations, e.g., π decreases to 0.7 or 0.5, more seeds for pushing are needed shown in Fig. 15.



(a) Probability of successful transmission $\pi = 0.9$



(b) Probability of successful transmission $\pi = 0.7$



(c) Probability of successful transmission $\pi = 0.5$

Fig. 15. Initial pushing ratios to satisfy 100%, 90%, and 80% of all users, considering the probability of successful transmission

8 Conclusion and Discussion

In this paper, we proposed the TOSS framework to offload the mobile cellular traffic by leveraging device-to-device local communications, with discussions on the pushing strategies to select the appropriate initial seeds depending on their spreading impact in the online SNS and their mobility impact in the offline MSN. Also the diverse user access delays are exploited and utilized for content sharing. Trace-driven evaluation reveals that TOSS can reduce a large portion (63.8% - 86.5%) of the cellular traffic while guaranteeing the access delay requirements of all users. Also we carried out a large-scale measurement-based evaluation by using *Xender*'s trace, and further proved the effectiveness of TOSS for being able to reduce up to 59.9% of the mobile traffic under much complex scenario settings of 100 groups.

From practical perspective, TOSS framework depends on a collaboration among the mobile operators, content providers and social networks, which will induce significant optimization of the traffic by offloading. An easy beginning of TOSS is a mobile SNS application, with extra functions for discovering nearby SNS friends, friends of friends, and even strangers, for exploring and transmitting files among them by both active "request-to-share" and proactive "background-share" mechanisms. Because it is expected to be able to obtain the content object before users may access it with high probability, the sharing can be carried out in the background, which can be considered as prefetching. When the user wants to directly access the object, which is not prefetched yet, on-demand delivery will be carried out then. Practically, D2D sharing alone cannot be the major methodology of content dissemination in mobile networks, but as there has been a rising number of new services of D2D, it should be integrated and utilized together with the cellular-based content on-demand delivery services to bring better quality of service and higher quality of experience to mobile users.

References

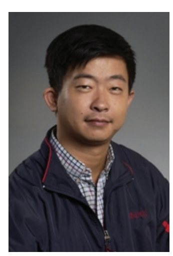
[1] X. Wang, M. Chen, Z. Han, D. O. Wu, T. T. Kwon, "TOSS: Traffic Offloading by Social Network Service-Based Opportunistic Sharing in Mobile Social Networks," *IEEE INFOCOM*, 2014.
[2] S. Scellato, C. Mascolo, M. Musolesi, and J. Crowcroft, "Track Globally, Deliver Locally: Improving Content Delivery Networks by Tracking Geographic Social Cascades," *WWW*, 2011.
[3] S. Ioannidis, A. Chaintreau, and L. Massoulie, "Optimal and Scalable Distribution of Content Updates over A Mobile Social Network," *IEEE INFOCOM*, Rio de Janeiro, Brazil, April, 2009.
[4] F. Malandrino, M. Kurant, A. Markopoulou, C. Westphal, and U. C. Kozat, "Proactive Seeding for Information Cascades in Cellular Networks," *IEEE INFOCOM*, USA, March, 2012.
[5] X. Wang, M. Chen, Z. Han, T. Kwon, and Y. Choi, "Content Dissemination by Push & Share in Mobile Cellular Networks," *MASS*, Las Vegas, USA, October, 2012.
[6] Y. Li, Y. Jiang, D. Jin, L. Su, L. Zeng, and D. Wu, "Energy-efficient Optimal Opportunistic Forwarding for Delay-Tolerant Networks," *IEEE TVT*, vol.58, no.9, pp.4500–4512, November, 2010.
[7] N. Wang and J. Wu, "Opportunistic wifi offloading in a vehicular environment: Waiting or downloading now?" in *INFOCOM*, 2016.
[8] L. Pelusi, A. Passarella, and M. Conti, "Opportunistic Networking: Data Forwarding in Disconnected Mobile Ad-hoc Networks," *IEEE Communications Magazine*, 2006.
[9] B. Han and A. Srinivasan, "eDiscovery: Energy Efficient Device Discovery for Mobile Opportunistic Communications," *IEEE ICNP*, Austin, USA, October, 2012.
[10] C. Yu, K. Doppler, C. Ribeiro, and O. Tirkkonen, "Resource Sharing Optimization for Device-to-Device Communication Underlaying Cellular Networks," *IEEE TMC*, vol.10, no.8, pp.2752–2763, 2011.
[11] *Xender*, http://www.xender.com/
[12] *SHAREit*, http://www.ushareit.com/
[13] *Zapya*, http://www.kuaiya.cn/
[14] X. Zhang, G. Neglia, J. Kurose, and D. Towsley, "Performance Modeling of Epidemic Routing," *Computer Networks*,
[15] Y. Li, G. Su, P. Hui, "Multiple Mobile Data Offloading Through Delay Tolerant Networks," *ACM MobiCom, CHANT*, 2011.
[16] J. Yang and J. Leskovec, "Patterns of Temporal Variation in Online Media," *ACM WSDM*, Hongkong, August, 2011.
[17] B. Han, P. Hui, V. S. A. Kumar and A. Srinivasan, "Mobile Data Offloading through Opportunistic Communications and Social Participation," *IEEE TMC*, vol.11, no.5, pp.821–834, May, 2011.
[18] F. Hao, G. Min, J. Chen, F. Wang and L. T. Yang, "An Optimized Computational Model for Multi-Community-Cloud Social Collaboration," *IEEE Transactions on Services Computing*, vol.3, no.3, 2014.

[19] T. Rodrigues, F. Benvenuto, M. Cha, and V. Almeida, "On Word-of-Mouth Based Discovery of the Web," *ACM IMC*, 2011.
[20] M.D. Choudhury, H. Sundaram, A. John, D.D. Seligmann, and A. Kelliher, "Birds of a Feather: Does User Homophily Impact Information Diffusion in Social Media?" *CoRR*, 2010.
[21] H. Kwak, C. Lee, H. Park, and S. Moon, "What is Twitter, a Social Network or a News Media?" *ACM WWW*, USA, April, 2010.
[22] G. V. Steeg and A. Galstyan, "Information Transfer in Social Media," *ACM WWW*, Lyon, France, 2012.
[23] G. Comarela, M. Crovella, V. Almeida, and F. benevenuto, "Understanding Factors that Affect Response Rates in Twitter," *ACM HT*, Milwauke, USA, June, 2012.
[24] T. Karagiannis, J. Y. Le Boudec, and M. Vojnovic, "Power-Law and Exponential Decay of Inter Contact Times between Mobile Devices," *ACM MobiCom*, Montreal, Canada, September, 2007.
[25] M. P. Wittie, V. Pejovic, L. Deek, K. C. Almeroth, and B. Y. Zhao, "Exploiting Locality of Interest in Online SNS," *ACM CoNEXT*, Philadelphia, USA, November, 2010.
[26] P. Hui, J. Crowcroft, and E. Yoneki "BUBBLE Rap: Social-Based Forwarding in Delay-Tolerant Networks," *ACM MobiHoc*, Hongkong, December, 2008.
[27] W. Gao and G. Cao, "User-Centric Data Dissemination in Disruption Tolerant Networks," *IEEE INFOCOM*, Shanghai, China, 2011.
[28] C. Boldrini and A. Passarella, "ContentPlace: Social-aware Data Dissemination in Opportunistic Networks," *ACM MSWiM*, 2008.
[29] D. Rothfus, C. Dunning and X. Chen, "Social-Similarity-based Routing Algorithm in Delay Tolerant Networks," *IEEE ICC*, 2013.
[30] X. Bao, Y. Lin, U. Lee and R. Choudhury, "Exploiting Naturally Clustered Mobile Devices to Offload," *INFOCOM*, 2013.
[31] Global File Sharing Industry White Paper, Yeahmobi Tech. Rep., http://cn.yeahmobi.com/pdf/2015-2016_global_P2P_file_transfer_business_whitepaper_India.pdf
[32] R. Lu, X. Lin, T. H. Luan, X. Liang, X. Li, L. Chen and X. Shen, "PReFilter: An Efficient Privacy-preserving Relay Filtering Scheme for Delay Tolerant Networks," *INFOCOM*, USA, 2012.
[33] R. Schlegel, C. Y. Chow, Q. Huang and D. S. Wong, "Privacy-Preserving Location Sharing Services for Social Networks," *IEEE Transactions on Services Computing*, vol.10, no.5, 2017.
[34] X. Zhuo, W. Gao, G. Cao, and Y. Dai, "Win-Coupon: An Incentive Framework for 3G Traffic Offloading," *IEEE ICNP*, 2011.
[35] V. Chaoji, S. Ranu, R. Rastogi and R. Bhatt, "Recommendations to Boost Content Spread in Social Networks," *ACM WWW*, 2012.
[36] R. Xiang, J. Neville, and M. Rogati, "Modeling Relationship Strengthin Online Social Networks," *ACM WWW*, 2010.
[37] L. Gyarmati and T. A. Trinh, "Measuring User Behavior in Online Social Networks," *IEEE Network Magazine*, vol.24, pp.26–31, 2010.
[38] H. Aminikhah and J. Biazar, "A New Analytical Method for System of ODEs," *Numerical Methods for Partial Differential Equations*, vol.26, no.5, pp.1115–1124, June, 2010.
[39] R. P. Satorras and A. Vespignani, "Epidemic Spreading in Scale-Free Networks," *Phys. Rev. Letter*, vol.86, issue.14, April, 2001.
[40] D. Towsley, "A Walk in the Dark: Random Walks and Network Discovery," *IFIP Networking, Keynote*, New York, USA, May, 2013.
[41] N. Eagle and A. Pentland, "Reality Mining: Sensing Complex Social Systems," *Pers. Ubiquitous Comput.*, vol.10, pp.255–268, 2006.
[42] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott, "Impact of Human Mobility on Opportunistic Forwarding Algorithms," *IEEE TMC*, vol.6, no.6, pp.606–620, June, 2007.
[43] Y. Zheng, L. Zhang, X. Xie, and W. Ma, "Mining Interesting Locations and Travel Sequences from GPS Trajectories," *ACM WWW*, Madrid, Spain, April, 2009.
[44] H. Huang, P. Luo, M. Li, D. Li, X. Li, W. Shu, and M. Wu, "Performance Evaluation of SUVnet With Real-Time Traffic Data," *IEEE TVT*, vol.56, no.6, pp.3381–3396, November, 2007.
[45] S. Brin and L. Page, "The Anatomy of A Large-scale Hypertextual Web Search Engine," *Computer Networks and ISDN Systems* , vol. 30, no.1-7, pp. 107–117, April, 1998.
[46] Xinyu Xing, Jianxun Dang, Shivakant Mishra and Xue Liu, "A Highly Scalable Bandwidth Estimation of Commercial Hotspot Access Points," in IEEE INFOCOM, June 2011.
[47] Alyaa Syaza Azini, Muhammad Ramlee Kamarudin, Muzammil Jusoh, "Transparent antenna for WiFi application: RSSI and throughput performances at ISM 2.4 GHz," in Springer Telecommunication Systems, Vol.61, No.3, pp. 569-577, 2016.

Xiaofei Wang got M.S. and Ph.D. degrees from the School of Computer Science and Engineering, Seoul National University in 2008 and 2013, respectively. He is currently a professor in School of Computer Science and Technology, Tianjin University, China. He got IEEE Communications Society Fred W. Ellersick Prize in 2017. His research interests are social-aware multimedia service in cloud computing, cooperative backhaul caching, and traffic offloading.

Min Chen Min Chen is the director of Embedded and Pervasive Computing (EPIC) Lab at Huazhong University of Science and Technology. His Google Scholars Citations reached 12,300+ with an h-index of 54. His top paper was cited 1280+ times. He is an IEEE Senior Member since 2009. His research focuses on 5G Networks, Edge Computing, Healthcare Big Data, Emotion Detection and Robotics, etc.

Victor C. M. Leung is a Professor of Electrical and Computer Engineering and holder of the TELUS Mobility Research Chair at the University of British Columbia, Vancouver, Canada. He has co-authored more than 1100 technical papers in the areas of wireless networks and mobile systems. In 2017, Dr. Leung received the Canadian Award for Telecommunications Research, the IEEE Communications Society Fred W. Ellersick Prize and an IEEE Systems Journal Best Paper Award. Dr. Leung is a Fellow of the Royal Society of Canada, the Canadian Academy of Engineering and the Engineering Institute of Canada.



Zhu Han received B.S. and Ph.D. degrees from Tsinghua China and the University of Maryland, College Park, USA, in 1997 and 2003, respectively. He is currently a John and Rebecca Moores Professor with the Electrical and Computer Engineering Department, University of Houston. He is IEEE Fellow since 2014 and IEEE Distinguished Lecturer since 2015. He is 1% highly cited researcher according to Web of Science 2017.

Kai Hwang is currently a Presidential Chair Professor, Chinese University of Hong Kong, Shenzhen, China. He received the Ph.D. from UC Berkeley in 1972. He has published in computer architecture, parallel processing, cloud computing, and network security. His Google Scholar citation was 17,400 with an h-index of 55 in 2018. His latest two books: Cloud Computing for Machine Learning and Cognitive Applications (The MIT Press) and Big Data Analytics for Cloud/IoT and Cognitive Computing (Wiley, U.K.) were published in 2017.