Design and Analysis of an Efficient Friend-to-Friend Content Dissemination System

Kanchana Thilakarathna, Aline Carneiro Viana, Aruna Seneviratne, Henrik Petander

Design and Analysis of an Efficient Friend-to-Friend Content Dissemination System

Kanchana Thilakarathna,* Aline Carneiro Viana,† Aruna Seneviratne,* Henrik Petander*

Abstract—Opportunistic communication, off-loading and decentrlaized distribution have been proposed as a means of cost efficient disseminating content when users are geographically clustered into communities. Despite its promise, none of the proposed systems have not been widely adopted due to unbounded high content delivery latency, security and privacy concerns. This paper, presents a novel hybrid content storage and distribution system addressing the trust and privacy concerns of users, lowering the cost of content distribution and storage, and shows how they can be combined uniquely to develop mobile social networking services. The system exploit the fact that users will trust their friends, and by replicating content on friends' devices who are likely to consume that content it will be possible to disseminate it to other friends when connected to low cost networks. The paper provides a formal definition of this content replication problem, and show that it is NP hard. Then, it presents a community based greedy heuristic algorithm with novel dynamic centrality metrics that replicates the content on a minimum number of friends' devices, to maximize availability. Then using both real world and synthetic datasets, the effectiveness of the proposed scheme is demonstrated. The practicality of the the proposed system, is demonstrated through an implementation on Android smartphones.

Index Terms—Opportunistic Content Dissemination; User Generated Content Sharing; Mobile Social Networking; Dynamic Centrality Metrics

1 Introduction

The past few years have seen a rapid growth in the use of free social networking applications such as Facebook, Twitter and Google+. In addition, other centralized content hosting services that support the distribution of user generated content (UGC) such as YouTube and Flickr are also becoming widely used. The UGC that is shared through these services are increasingly being generated and consumed by users using their mobile devices, necessitating the transfer of content between mobile devices and the content hosts [8]. This will impact the users as well as the mobile operators as it will (a) exacerbate congestion of the mobile networks [8] and (b) make the problems associated with privacy and data ownership even more acute. As the service providers will not only have full control of the user data [5], but have access to more private information that are available through the mobile devices, such as location.

There have been numerous proposals for dealing with increasing mobile data traffic by taking advantage of ubiquitous availability of mobile devices and access to different types networks [2], [13], [25], [40]. Majority of the proposals for using mobile devices exploit the possibility of using short-range communication to communicate with each other when they are in close proximity, i.e. *opportunistic communication* [2], [13]. The proposals that exploit the availability of different type of networks, transfer data through the least congested and/or cheapest network whenever possible, i.e. *offloading*. In doing so, both solutions attempt to lower the congestion in any given network and minimize the cost.

Despite offering additional advantages for mobile users, such as providing connectivity when there is no direct access to a network [23], opportunistic communication solutions have not seen wide spread adoption, primarily for two reasons. Firstly, due to the inherent reluctance by users to interact with strangers or third parties, despite security and privacy concerns are being partially addressed by the use of encryption. Secondly, due to the unbounded latencies of data transport.

In contrast, offloading offer a generic solution for reducing congestion in a given network. Offloading between cellular and WLANs have been adopted by a number of operators. However, they still suffer from the lack of universal availability of WLANs provided by the same cellular network operator and the time taken for authentication. Importantly, neither of these solutions directly address loss of control of data and privacy.

There is a separate body of work that address the issues of loss of control of data and privacy. These works have led to the development of *distributed decentralized storage and delivery*, especially for social networking applications [10], [14], [32], [35]. In decentralized storage and delivery systems, individual users or a community of users host their data, thus providing the users full control and preventing third parties mining private information. However, these systems lead to increase in data traffic, as they require the replication of user data on distributed servers [40], and increases the cost and complexity for the user as they have to manage the hosting of data.

In this paper, we propose a new hybrid content storage and distribution system for user generated content (UGC). Despite the obviousness of developing the a hybrid scheme, that exploits the advantages of opportunistic networks, offloading and decentralized storage and delivery, to date such a scheme has not yet been proposed again

**Kanchana Thilakarathna, Aruna Seneviratne and Henrik Petander are with NICTA, Australia and School of EE&T, UNSW, Australia.*
†Aline Carneiro Viana is with INRIA, France.

due to two primary reasons. First, because of the high communication and energy cost of decentralized content storage and distribution for mobile systems. Second, due to the inability of guaranteeing delivery latencies in opportunistic networks. In this paper, we address these two fundamental limitations by 1) exploiting the possibility of content replication, (2) considering initial encounter time and duration of users' encounters, and (3) utilizing existing social networking services for opportunistic dissemination. The proposed solution creates a distributed decentralized storage system with intelligent content replication, which reduces mobile data traffic and provides the users full control at minimal cost which can be used to provide mobile social networking services.

The paper makes the following contributions;

- Presents a hybrid content dissemination system for mobile social networks, which takes advantage of trusted social networking friends.
- Provides a formal definition of content replication, which maximizes content availability and minimizes replication and shows this to be NP-hard.
- Presents a community based greedy algorithm for efficient content replication by taking advantage of routine behavioral patterns of mobile users.
- Proposes dynamic centrality metrics to identify the most influential users within communities to minimize the content replication and delivery delay.
- Shows that it is possible to provide delivery success rates of 80% with less than 10% replication, through extensive data driven simulations using both real world and synthetic datasets.
- Demonstrates the practicality of the proposed hybrid content dissemination through the implementation of the system on Android smartphones.

The remainder of the paper is organized as follows: Section 2 presents the related work. The formalization of the problem of content replication is presented in Section 3 and the overview of the proposed system is presented in Section 4. Section 6 presents the dynamic centrality metrics and community based content replication algorithm. Section 7 evaluates the performance of proposed metrics and content replication algorithm followed by details of the implementation of the system on real devices in Section 8. Finally, Section 9 concludes the paper.

2 Related Work

There is a vast body of work in mobile content storage and distribution. The work related to this paper can be broadly categorized into three areas, namely social networks, opportunistic content dissemination and hybrid content dissemination using decentralized storage.

2.1 Social Networks using Decentralized Storage

These systems address the issue of user privacy and loss of control of data, by enabling users to host their own data. Diaspora [14] was one of the only widely used social networking system using distributed storage. To host data, Diaspora users need to set up their own server. If a user's mobile device is used to host the content, either the availability of the content has to be compromized or the user will incur increased communications costs [40]. If a cloud based hosting solution is used, the users will incur the costs of uploading/downloading, and will increase the traffic on the network as well as the cost of hosting. Safebook [10] is based on the concept of decentralization and collaboration among friends to create a secure social network. Similar to our system, friends are assumed to be cooperative and their devices are used for storage to increase availability. SuperNova [35] is another recently proposed decentralized social networking system that uses content/profile replication. The idea is to increase the online availability of content, by using a super-peer based network of volunteer agents. In both systems, to increase the availability of content, the number of replicas of data need to be increased, which leads to increased communication costs and energy consumption when used with mobile devices.

Tribler [32] is a peer-to-peer file sharing system, where peers are clustered into social groups and replicate their contextual information. Although there are similarities to our solution, Tribler does not consider methods to increase the availability of the content or minimize communication costs. Sharma et al. [36], similar to the proposed scheme, presents a friend-to-friend content replication strategy to ensure minimal replication and maximal availability. Again it leads to increased communication costs and energy consumption when used with mobile devices. A erasure coding based friend-to-friend storage system is proposed in [12]. The coding based redundancy techniques are generally not suitable for social networking content due to their smaller size and frequency of access as discussed in [35].

2.2 Opportunistic Content Dissemination and Decentralized Storage

There has been considerable work on opportunistic routing protocols [17]. However, this work is concerned with the routing of information between the source and destination. As the focus of this paper is efficient content storage and distribution, opportunistic routing schemes are not considered in this section.

There have been a number studies that investigate the possibility of improving the content delivery latency using social behavioral patterns of users. In [31], the authors have analysed the effectiveness of temporal communities in storage and dissemination of content opportunistically. As expected, the results show that the users in contact with a large number of users that are mobile, are mostly effective in opportunistic content dissemination. In [3], [29], the authors attempt to predict users future contact patterns and use them as "content transporters". However, only using such users is likely to increase delivery delays. Reich et al. [33] study the effects of user impatience in content dissemination, when content is disseminated only when two users meet other. In this paper, we investigate effectiveness of generic content dissemination and show that it is inefficient when the number of *consumers* is lower. Then, we propose how these temporal communities can be leveraged to minimize the delivery delays.

In [27], [37], the authors present a social community based cooperative content caching and retrieval schemes. It is aimed at minimising the cost of content distribution to users with common interests that are physically colocated by introducing the notion of ”familiar strangers”. Yungki et al. [26] propose a continuity-aware cooperator detection method for an ambient monitoring system again based on the historical contact patterns of users. All these cooperative content storage and dissemination methods, do not address the trust, security and privacy issues, requires users to interact with strangers. GameOn [46] is another recently proposed solution for peer-to-peer mobile gaming platform for the passengers in public transport. Although the proposed concepts can be applicable to other application scenarios, it is not applicable to disseminating content to a group of user who are not always in the vicinity of each other as proposed in this paper.

2.3 Hybrid Content Dissemination and Decentralized Storage

There have been many proposals of which make use of the networking infrastructure to replicate content in a selected set of users. Han et al. [13] proposed the use of a target set users for content replication and opportunistic communication for content propagation. The focus of this work is on the dissemination of data to and from a centralized data store. Thus, it does not address the privacy and trust issues of users as they are again required to interact with strangers.

Ioannidis et al. [18] proposed a distributed caching mechanism for the purpose of social welfare where users cache content downloaded through the networking infrastructure. However, the solution is not for sharing UGC. Similarly, Whitebeck et al. [45] proposed a hybrid content delivery system with a control loop through which users send acknowledgements of delivery to a central service provider. Even though this has been proposed for general mobile users, the main focus and simulation results are for a vehicular network. VIP delegation [2] replicates data on a few “socially important” users in a mobile network. Moreover, none of the proposals again consider the trust and privacy of the users and consider the dynamic aspects of contact time. Microcast [21] is another hybrid scheme that allows group of users to reduce their cellular network usage when downloading the same content at the same time. Each phone in the vicinity downloads different parts of the same content and then the pieces of content will be shared locally. Microcast is designed for the case where all users want the file at the same time and all users are at the same location, which can not be used for disseminating content among a group of users.

TABLE 1
Content creation and access model.

Content creation model	
Amount per week	142MB [8]
File Size	Gamma(scale=2,mean=4MB) [1]
Inter-arrival time	Exponential (mean=3.5 hrs) [25]
Content access model	
No. of consumers	Pareto Type II (80-10 rule) [44]
Consumer location	Random distribution
Transfer rate	2 Mbps [25]
Delivery deadline	3 days

MobiTribe [40], [41] presents a hybrid content sharing overlay for existing centralized social networking services. MobiTribe provides the user more control over their own data leveraging distributed storage on social networking friends rather than the centralized service providers. In [41], it does not consider geographical proximity of the friends to deliver content locally.

3 Motivation

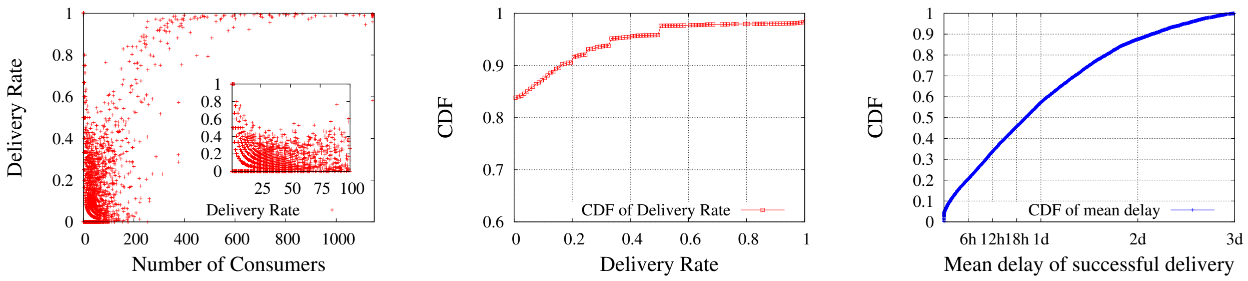
We investigate the effectiveness of opportunistic communication for sharing UGC, when only interactions among a content *creator* and an interested set of *consumers* are considered, assuming that there is only one content *creator*. This highlights the benefits of initial content replication and formally define the problem of content replication.

3.1 Evaluation setup

Real-world Dartmouth campus dataset [24] that consists of two months of data from January to March 2004 with contact patterns of 1146 users are used. We consider two users to be in contact with each other when they are connected to the same WiFi access point, as described in [4]. As the dataset only describes mobility of users, to simulate opportunistic dissemination, we assign the users *content creation* and *content access* patterns which are summarized in TABLE 1.

Content creation model: We assume that each user in the dataset generates content over two months. Cisco has predicted that an average smartphone will consume 2.6GB of data per month by 2016 [8]. Though the amount of UGC is predicted to increase, the ratio between upload and download is expected to be 25% to 75% [8]. Similarly, we assume that the average smartphone user generates 142MB per week, considering 25% of 2.6GB is evenly distributed among 30 days. The inter-content generation-time, or the inter-arrival-time of content is reported to be exponentially distributed [25] and therefore these 142MB of content items are assumed to be distributed throughout the week with a mean inter-arrival-time of 3.5 hours. The size of generated content is characterized by a Gamma distribution with 4MB mean as described in [1].

Content access model: The *consumers* are randomly selected among the total user population of the dataset. Hence, the locations of the *consumers* are also randomly distributed according to their geographic coordinates. As content popularity is reported to be follow a Pareto distribution [1], and degree distribution in Facebook follow power-law distribution [44], we assume that the



(a) Content delivery rate (b) CDF of content delivery (c) CDF of mean delay

Fig. 1. Effectiveness of opportunistic communication in content dissemination.

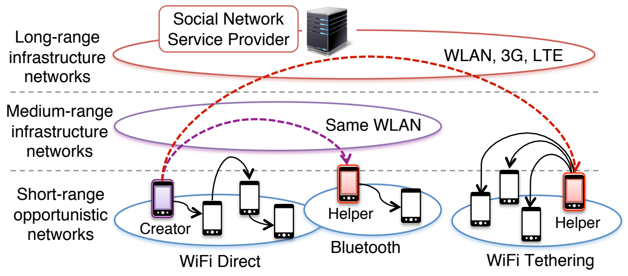

Social Network Service Provider
Long-range infrastructure networks
WLAN, 3G, LTE
Medium-range infrastructure networks
Same WLAN
Short-range opportunistic networks
Creator
Helper
Helper
WiFi Direct
Bluetooth
WiFi Tethering

Fig. 2. Overview of the proposed concept of hybrid content replication and dissemination.

number of *consumers* can also be modelled by a Pareto distribution. Therefore, the number of *consumers* are selected using Pareto distribution where 80% of content requests accounts for 10% of contents. If one of the friends is allowed to re-share it will be considered as a newly generated content.

Opportunistic dissemination model: Since WLANs is the most pervasive network for opportunistic communication, we consider a practical data transfer rate of 2Mbps [25]. Further, we take content delivery deadline to be 3 days, to evaluate the best case scenario in terms of delivery success rate. In fact, 50% of Facebook users login every day[1] indicating that delivery delay should be less than 1 day for at least 50% of the users. It is further assumed that (1) no initial content replication is performed and (2) the *consumers* are collaborative and once content is downloaded, they unconditionally share it with *consumers* nearby. We assume that the content is disseminated when a user carrying content (an infected user) meets a user who does not have the content (an uninfected user), similar to epidemic dissemination, with unlimited bandwidth and infinite buffers.

3.2 Effectiveness of opportunistic dissemination

If content can be opportunistically disseminated from a *creator* to a *consumer* before 3 days, it is considered to be a successful delivery otherwise a failure. The main factor that affects the delivery rate is the number of *consumers* as shown in Fig. 1a. For a large *consumer* population, the delivery rate is almost 100% as there are enough *consumers*. However, the number of *consumers* are often low in social networking, e.g. mean number of friends per user in Facebook is approximately 100 [44], which makes the successful delivery rate very low. As can be seen in Fig. 1b, the delivery rate is zero for approximately 84% of content. Fig. 1c shows the content delivery delay from the time of creation. The probability of mean delivery delay being less than 1 day is ∼60%.

The results show that using opportunistic communication, without initial content replication, is ineffective when the number of *consumers* is less than 150. Furthermore, due to the power-low distribution, majority of the groups contain a very low number of *consumers*. In such cases, it is possible to increase the number of seed nodes distributing the content to selected set of *consumers* using pre-existing communication infrastructure. Thus, there is an obvious trade-off between delivery performance and content replication overhead, and the challenge is to *maximize the content delivery rate with limited content replication.*

4 The System Architecture

The proposed system takes advantage of low-cost network connectivity among wirelessly connected friends, and the storage and processing power of mobile devices to address the aforementioned problem of maximising the content delivery rate with limited content replication. The use of a lowest cost network is dependent on the location of the source and destination devices. It has been observed in the literature that people have strong correlation between friendship and user mobility [6], [7]. If they are within the communication range of each other, opportunistic dissemination could be used. If not, the available lowest-cost infrastructure communication, e.g. off-peak cellular, could be used. The system exploits the availability of these hierarchical heterogenous networks to disseminate content depending on their location as shown in Fig. 2.

Consider the case where a *creator* wants to share content with a set of users who have previously been identified as *friends* through a social networking service (e.g. Facebook). Assume that potential *consumers* among these friends can be predicted based on their history of content consumption patterns. Then, we can propagate the content only to these *consumers* and let other *friends* to fetch the content from the *creator* or one of the *consumers*. If this predictive pushing and fetching can be scheduled to use low-cost networks depending on the location of the users (Fig. 2), it will help to minimize the communication

[1] http://blog.kissmetrics.com/facebook-statistics/

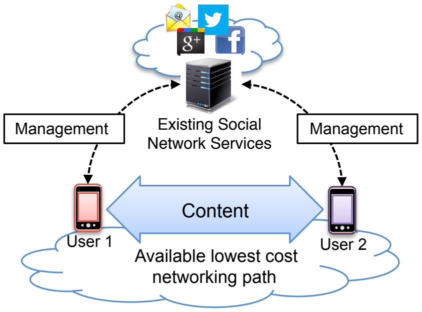

Management
Existing Social Network Services
Management
Content
User 1
User 2
Available lowest cost networking path

Fig. 3. Rely on existing services for advertising content.

costs and energy usage. Moreover, predictive pushing reduces the number of redundant transfers and reduces the storage costs of the mobile devices. We contend that despite it is difficult to predict the content consumption of a user with a high degree of accuracy, a low prediction accuracy will still generate a significant impact because of the increase in content availability due to content replication on the *consumers*.

If opportunistic communication are used for disseminating content, the main challenge is guaranteeing the timeliness of delivery. The delivery time can be reduced by replicating the content on carefully selected *consumers*, namely *helpers*, as shown in Fig. 2. The viability of the proposed system depends on the selection of *helpers* and minimization of replication. We consider that a central entity selects the *helpers* based on the users' connectivity information, as described in Section 6. Since *helpers* are only selected from the *friends*, the privacy of the users is better preserved. As shown in Fig. 2, initial content replication is carried out by pre-existing networking infrastructure and/or opportunistic communication.

For content dissemination among the users, a modified version of BitTorrent peer-to-peer (P2P) protocol is used. In particular, a *consumer* becomes a *propagator* only after the *consumer* has completely downloaded the content. This ensures a user can first download the full content for its own use before helping others.

4.1 Leveraging on existing social network services

The idea is to decouple the distribution of shared notification from the actual content transfer as shown in Fig. 3. The proposed system will be developed to a mobile app (application), namely *Yalut*, as an overlay service that runs on top of the existing social networking services. The app requires the read and write permissions to social networking feeds of the particular service, e.g. Facebook, if the user want to send shared notification through that particular service, e.g. Facebook.

Content sharing. When a *creator* selects to share a content item, the *Yalut* app advertises the shared notification on the selected social networking service using the available lowest-cost networking infrastructure and the *friends* of the *creator* will be notified. The *Yalut* app will not upload the actual content to the social networking service. At each device, the *Yalut* app monitors the social networking feeds in the background and identifies the content that the user is likely to be consumed based on the history of content access patterns. This allows the *Yalut* service to identify the potential *consumers*.

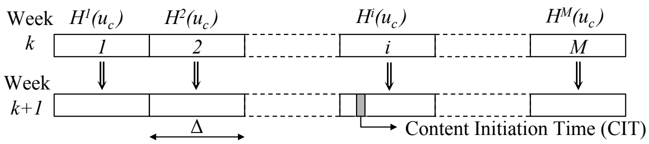


Fig. 4. Periodic weekly helper selection.

TABLE 2
Summary of symbols for content replication.

Symbol	Description
c	content creator
u, v	users
V	set of users
G_t	dynamic contact graph at time t
E_t	set of edges at time t
e	an edge in E_t
Δ	delivery deadline
P_t	set of propagators at time t
$\alpha(u)$	minimum contact duration required for user u to receive full content from content propagators
$\sigma(c)$	consumers covered only by the creator c
$H(c)$	set of selected helpers for the creator c
$\lambda(c)$	limit of helpers for the creator c

Content downloading. Once the potential *consumers* are identified, the content will be initially replicated on to the *helpers* either a pre-existing networking infrastructure or opportunistic communication as stated earlier. Then, the *Yalut* app rely on *helpers* to disseminate the content opportunistically to all other *consumers*.

We have successfully integrated the *Yalut* app with the most popular existing services, e.g. Facebook, Google+ and Twitter, on Android devices. Section 8 provides further details of the practical implementation and the developed *Yalut* app is available on the Google Play Store.

4.2 User Privacy and Incentives

Content sharing with friends inherently reduces the privacy of the user through direct and indirect channels [9]. Thus, it is impossible to develop a system that fully guarantees privacy and confidentiality of users. Our aim is to minimize loss of privacy and confidentiality threats.

In general, content replication compromises privacy. The aim is to minimize privacy leakage by using trusted social networking friends, keeping data away from the centralized hosts and not propagating content through strangers. In addition to the privacy preservation, the reduction of users cellular data usage can also be considered as an incentive to use the system. We believe that this is sufficient for users to collaborate with their friends for the benefit of the community of friends as a whole. In addition, it is possible to incorporate a credit scheme where *helpers* accumulates credits in return for propagating others content, as proposed in [34].

5 Problem Formulation

The effectiveness of the proposed system in terms of content delivery latency and delivery success rate depend on the selection of *helpers* and the minimization of *helpers*. Therefore, we first formally define the content replication problem and followed by the content replication algorithms.

5.1 Formal Definition of Content Replication

Consider a dynamic contact graph $G_t = (V, E_t)$, where V is a set of users and E_t is an edge set at time slot $t \in (1, 2, \cdots, n)$. An edge $e \in E_t$ exists among two users, if they are within the communication range at time t. Without loss of generality, the length of each time slot is considered as one unit, which represents the minimum duration in which there is no change in the topology. Suppose a *creator* $c \in V$ wants to share content via a mobile social networking application, i.e. *consumers*, in $V \setminus c$. A *consumers* is *covered* if it receives the full content within the content delivery deadline of Δ time slots. *Consumers* are assumed to be collaborative and they become content *propagators* only after being covered. Let $\alpha(u)$ be the minimum total contact duration required by a *consumer* u to receive the full content from *propagators*. We denote $P_t \in V$ as the set of *propagators* at time t. When there is no initial replication, $P_1 = c$. Then, if a user u has to receive the full content, the aggregated contact duration of u with *propagators* should be greater than $\alpha(u)$,

$$\text{i.e.} \sum_{t=1}^{\Delta} \mathbb{I}((u, v) \in E_t \text{ for some } v \in P_t) \geq \alpha(u), \tag{1}$$

where the indicator function,

$$\mathbb{I}(\textit{statement}) = \begin{cases} 1 & \text{if } \textit{statement} = \text{true} \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

Hence, the set of *consumers* covered by a *creator* c is:

$$\sigma(c) = \left\{ u \in V : \sum_{t=1}^{\Delta} \mathbb{I}((u, v) \in E_t \text{ for some } v \in P_t) \geq \alpha(u) \right\} \tag{3}$$

Consider a set of helpers $H(c) \in V$ for a *creator* c. Thus, the *creator* and the helpers are the initial set of *propagators*, $P_1 = H(c) \cup c$. The objective is to cover all *consumers* with minimum number $\lambda(c)$ of helpers. Then, our CONTENT REPLICATION (CR) problem is to minimize the cardinality of the set P_1 such that it covers all *consumers* in V, formally;

$$\begin{array}{ll} \text{Minimize} & |P_1| \\ \text{subject to} & \sigma(c) = V \setminus P_1 \end{array} \tag{4}$$

Here, we show that the CR problem is computationally NP-Hard even for a simple instance of a static social graph, where $E_t = E \; \forall \; t$ and $\alpha(u) = 1 \; \forall \; u \in V$, i.e. the full content can be transferred in a single contact. This is similar to best case scenario where there is unlimited bandwidth and zero bit error rates.

Theorem 1. *CR is NP-Hard even when $E_t = E \; \forall \; t$ and $\alpha(u) = 1 \; \forall \; u \in V$.*

Proof: We show that *minimum dominating set* is polynomial time reducible to CR problem. Let $G'' = (V'', E'')$ be an undirected graph. A *dominating set* of the graph G'' is a $D \subseteq V''$ such that every vertex $u'' \notin D$ is adjacent to at least one member of D. The dominating number $\gamma(G'')$ is the cardinality of the smallest dominating set. For a given positive integer k, the decision problem of whether there exist a $\gamma(G'') \leq k$ is one of the well-known NP-Complete problems [20].

Recall the dynamic contact graph $G_t = (V, E_t)$. Since we assume that $E_t = E \; \forall \; t$, G_t becomes a static undirected contact graph $G = (V, E)$. In addition, $\alpha(u) = 1 \; \forall \; u \in V$ makes that a vertex u can be covered if there is at least one edge to the set of initial *propagators* in P_1. Then, the decision problem of CR is to find whether there is a set of initial *propagators* P_1 of size at most $|P_1| = k$ which covers maximum number of *consumers* $\sigma(c)$. The coverage is maximized only when $|\sigma(c)| = |V \setminus P_1|$, i.e P_1 has to be a minimum dominating set of size k.

This follows immediately that if there is a solution to the decision problem of CR, there should be a solution to the minimum dominating set problem. Since the decision problem of the minimum dominating set is NP-Complete, the hardness of the optimisation problem of CR becomes NP-Hard. □

6 Replication Algorithms

The probability of using users encounters is highly dependent on their social behavior. Hence, there is a diurnal correlation of opportunistic encounters among users. These patterns have been extensively analyzed [2]. Usually, social behavior of the majority of users have weekly routines. Further, there is higher probability that a user meets the same people at the same day and time in every week. This predictive regularity of encounter patterns can be leveraged of efficient content replication. In order to allow instant content dissemination, *helpers* can be selected in advance: i.e., helpers for the week $k+1$ can to be selected during the week k, as shown in Fig. 4.

Consider the week (Δ_k) is divided into Δ time slots, where the Δ is the content delivery deadline. Since when a *creator* c is going to generate a content during the week $k+1$ is unknown, we select several sets of *helpers* for week $k+1$ during the week k as $H_{k+1}(c) = \left\{ H_k^j(c) : u \in V \text{ and } j \in [1 : \Delta_k/\Delta] \right\}$. At the end of every week, the central management entity performs *helper* selection and informs all *creators* the prospective sets of *helpers*. A *creator* will be assigned a new set of *helpers* only if they change their behavioral patterns significantly.

6.1 Greedy Helper Selection Algorithm

We utilize a greedy algorithm GREEDY-HELPERS for content replication (CR). The most influential user in the network is the one who has contact with the maximum number of *consumers*, i.e. has max$|\sigma(\cdot)|$, which can be intuitively used as a greedy choice property.

Algorithm 1 presents the naive greedy algorithm to select a set of *helpers* $H(c)$ for the *creator* c. D is the set of *consumers covered* by the *creator* and the selected *helpers*. After calculating $\sigma(u)$ for all $u \in V$, i.e. line 3, D will

TABLE 3
Summary of symbols for dynamic centrality metrics.

Symbol	Description
D	set of consumers covered by c and the selected helpers $H(c)$
$\alpha^t_{u,v}$	contact duration between u, v at time t
G	aggregated weighted graph of $G_t \forall t$ with $\alpha^t_{u,v}$ as edge weights
E	total set of edges $\forall t$
$N(u)$	set of neighbours of u
$C_{LD}(u)$	local metric: $\|N(u)\|$
$I(u,v)$	initial contact time for u, v for a given Δ
$D(u,v)$	aggregated contact duration for u, v over Δ
$w(u,v)$	weight of edge (u,v) equal to $I(u,v) + (1/D(u,v))$
$C_{LID}(u)$	local improved metric: $\|N(u)\| + \frac{\|N(u)\|}{\sum_{v \in N(u)} w_{u,v}}$
$p(u,v)$	binary parameter for the existence of path between u and v
$G'(V', E')$	directed aggregated contact graph
$C_{GP}(u)$	global metric: $\sum_{v \in V} p(u,v)$
$sp(u,v)$	shortest path between u and v in terms of $w(u,v)$
$C_{GIP}(u)$	global improved metric: $\sum_{v \in V'} p(u,v) + \frac{\sum_{v \in V'} p(u,v)}{\sum_{v \in V'} sp(u,v)}$
$com(u)$	set of users in u's community

be equal to the set of *consumers* covered by the *creator* $\sigma(c)$ (i.e. line 4) and the *helper* set $H(c)$ will be equaled to the *creator* c (i.e. line 5). Then, we loop through until we *cover* all devices or reach the threshold of replication $\lambda(c)$ while selecting the *consumer* with highest $|\sigma(u)|$ from the remaining *consumers*.

```
Algorithm 1 GREEDY-HELPERS($G_t, \Delta, \lambda, c$)
1.  $D \leftarrow H(c) \leftarrow \emptyset$
2.  for all $u \in V$ do
3.      Find $\sigma(u)$
4.  $D \leftarrow \sigma(c)$
5.  $H(c) \leftarrow c$
6.  while $|H(c)| \leq \lambda(c)$ or $D \neq V$ do
7.      Let $u \in (V \setminus (H(c) \cup D))$ maximizing $|\sigma(u)|$
8.      $H(c) \leftarrow u$
9.      $D \leftarrow D \cup \sigma(u)$
10. return $H(c)$
```

This set-covering flavoured solution has considerably high level of approximation factor. Kempe et al. [22] shows that this type of greedy algorithm is $(1 - 1/e)$ approximation, where e is the base of the natural algorithm. Even though, this provides an acceptable approximation algorithm for CR problem, finding $\sigma(u)$ for all $u \in V$ is computationally too complex in a dynamic network under resource constraints. In our previous work [42], we proposed computationally simple dynamic centrality metrics that exploit the temporal and spatial regularity of social wireless connectivity patterns. This paper extend this work and present a much deeper analysis.

6.2 Dynamic Centrality Metrics

As the first step, we aggregate every contact at a single graph without loosing any temporal information. Let an aggregated weighted graph $G = (V, E)$ consists of all edges in G_t, $\forall t \in (1, 2, \cdots, n)$ such that $G = G_1 \cup G_2 \cup \cdots \cup G_n$ and $\alpha^t_{u,v}$ be the edge weights at time t of each $e \in E_t$. $\alpha^t_{u,v}$ is the contact duration between the two users u and v at time t. For instance, if $\exists\ (u, v, \alpha^1_{u,v} = 20) \in E_1$ and $(u, v, \alpha^2_{u,v} = 30) \in E_2$, there are two edges in E connecting u and v with the contact duration of 20 and 30 seconds at $t = 1$ and $t = 2$. Then, we focus on centrality metrics in G, which provides better approximations for $\sigma(\cdot)$, i.e. expected number of covered *consumers*.

Hereafter, we propose two types of centrality metrics: local and global. Local metrics consider the information available locally (i.e. one-hop away) to decide the influence of the user. In addition to its simplicity and distributed calculation, the privacy of the users is well preserved: The users only send aggregated values to the central entity. In contrast, global metrics consider the whole network topology to decide the centrality of the user, which is more complex and needs to be carried out in a central location.

6.2.1 Local metrics

One of the simplest centrality metric that implies the capability of neighbourhood coverage is the degree centrality $C_{LD}(u) = |N(u)|$ where $N(u)$ is the set of neighbours of u in the aggregated graph G. Degree centrality identifies popular nodes in the network and thus, has higher influence on content propagation.

Nevertheless, simple degree centrality does not guarantee that all counted encounters are useful for propagations of content due to the lack of consideration of temporal information. Further, the contacts that happen early are important in propagation than those that happen later. Hence, a centrality metric which captures temporal information could be more realistic to be considered in dynamic networks. To this end, we define the initial contact time as $I(u,v) = min\{t\} : \alpha^t_{u,v} > 0$ for all $t \leq \Delta$ and the total contact duration $D(u,v) = \sum_{t=1}^{\Delta} \alpha^t_{u,v}$ for an edge (u,v). We calculate the weight $w_{u,v} = I(u,v) + (1/D(u,v))$ for all $(u,v) \in E$. $w_{u,v}$ has the meaning of earliness and solidity of the contact (u,v). In practice, each mobile device can calculate w locally for all other devices it encounters for a given period. To this end, we define an improved dynamic degree centrality metric:

$$C_{LID}(u) = |N(u)| + \frac{|N(u)|}{\sum_{v \in N(u)} w_{u,v}}$$

$N(u)$ is the set of neighbours of u. C_{LID} describes how early and how independently the user makes other users into content *propagators*. We aim to use C_{LID} as a greedy choice property for CR problem.

6.2.2 Global metrics

Even though centralized systems have disadvantages in terms of privacy and scalability, we make use of the global information to perform more accurate heuristics. Here, we define two path-based centrality metrics for node ranking. We first define a naive simple metric $C_{GP}(u) = \sum_{v \in V} p(u,v)$ for all $t \leq \Delta$ where $p(u,v) = 1$ if there is a path between u and v and $p(u,v) = 0$

otherwise. This can be viewed as an extended degree for node u giving heuristics about the popularity and the availability of the node. Simplicity of the metric is the main advantage, which requires only information about existence of a path. This metric is motivated by the fact that nodes information beyond one-hop contacts, i.e., *k-vicinity knowledge*, has been shown to be a key ingredient to improve opportunistic network forwarding [30].

As simplicity does not provide accurate heuristics, we define an improved dynamic centrality metric by considering temporal information such as the contact duration $D(u, v)$ and the initial contact time $I(u, v)$. We construct a directed aggregated graph $G' = (V', E')$ by directing all edges in G for both directions with same weights. Next, we prune all unrealizable edges in the network, i.e. if a content is to be propagated via a node, its outgoing contact has to take place after at least its first incoming contact. At this point, we have an aggregated graph and at each node there is a guarantee that content will be propagated to other nodes if the content has arrived at the node. We calculate shortest-path $sp(u, v)$ for all node pairs $(u, v) \in V'$ in terms of edge weights $w_{u,v} = I(u, v) + (1/D(u, v))$. We define a path-based dynamic centrality metric C_{GIP}, similar to C_{LID}, such that it implies how early and how independently the user makes other content *propagators* as,

$$C_{GIP}(u) = \sum_{v \in V'} p(u,v) + \frac{\sum_{v \in V'} p(u,v)}{\sum_{v \in V'} sp(u,v)}$$

6.3 Community based Greedy Algorithm

This algorithm combines social sub-structural properties such as communities with previously defined dynamic centrality metrics.

Algorithm 2 COMMUNITY-GREEDY$(G, G', \Delta, \lambda, c)$

```
1. $D \leftarrow H(c) \leftarrow \emptyset$
2. for all $u \in V$ do
3.     Find centrality metric
       $C_{LD}(u), C_{LID}(u), C_{GP}(u), C_{GIP}(u)$
4. communities ← k-clique-algorithm$(G', 3)$
5. Let a community $com(u)$ be the $u$'s community
6. $D \leftarrow com(c)$
7. $H(c) \leftarrow c$
8. while $|H(c)| \leq \lambda(c)$ or $D \neq V$ do
9.     Let $u \in (V \setminus (D \cup H(c)))$
       maximizing $C_{LD}(u), C_{LID}(u), C_{GP}(u), C_{GIP}(u)$
10.    $H(c) \leftarrow u$
11.    $D \leftarrow D \cup com(u)$
12. return $H(c)$
```

For this we detect communities using *k-clique* community algorithm. Then, we distribute *helpers* among communities based on their ranking given by the proposed dynamic centrality metrics as in Algorithm 2. First, the *consumer* with highest centrality value is selected as a *helper* and rely on that *helper* to propagate the content within the community. Then, the next highest *consumer* from a different community is selected, i.e. line 9 of the Algorithm 2. If the threshold of replication $\lambda(c)$ is lower than the number of communities, initial content *propagators* will not be selected from the *creator*'s community, assuming that the *creator* is capable to propagate the content within its community. If the majority of the *consumers* do not belong to communities, the selection is purely based on the centrality value of *consumers*.

It is expected that the selected *helpers* will become quasi-static over the time due to the regular behavior of people [11]. Hence, the *helper* calculation will be carried out only for users who have changed their behavioral pattern. In practice, the number of users of the service increases incrementally and as a result *helpers* will only need to be calculated for newly added users. Therefore, we believe that the *helper* selection algorithm will scale with the increasing number of users.

7 Performance Evaluation

7.1 Datasets

Two real-world datasets and three synthetic datasets that represent different user environments are used in the evaluation. A summary of basic information of the datasets used are presented in Table 4.

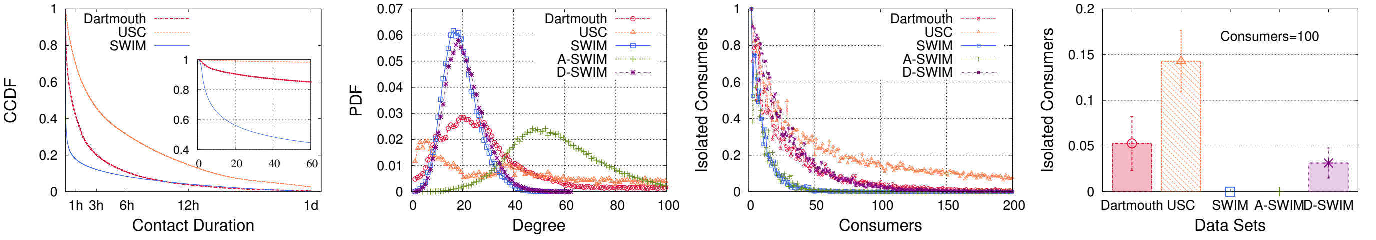
Dartmouth: The dataset [24] contains SNMP logs from the WiFi access points within the Dartmouth College campus during January to March 2004. When two users are connected to the same WiFi access point, we consider that the two users can exchange information as described in [4]. There are 1138 users whom in contact with approximately 12 other users per week.

USC: This dataset [19] contains connectivity patterns of users in a campus environment at the University of Southern California during a 8 week period between April to August 2005. Similar to Dartmouth, we create device-to-device connectivity patterns using the connected access point, which resulted in 61 contacts in average per week.

SWIM: This is a synthetic dataset that is generated using the SWIM simulator [28], which considers both human mobility behaviors and social interactions to generate contact patterns among groups of users. SWIM simulates 36 Bluetooth enabled iMotes, configured to log all visible mobile devices for 11 days. When two users are within the Bluetooth communication range of each other, we consider those two devices are in contact. SWIM dataset is further scaled up to generate 1500-nodes to understand the performance variation of the content replication in different environmental conditions as follows. *1) D-SWIM:* by keeping the density constant to represent large geographical area with a moderate number of users. *2) A-SWIM:* by keeping the area constant to represent an overpopulated small area.

7.2 Dynamics of the datasets

Fig. 5 shows different characteristics of the considered datasets and underlying contact graphs. Fig. 5a shows the aggregated duration of intermittent connectivity between two users per day. More than 50% of users in USC dataset have more than 3 hours of aggregated contact duration per day. This can be considered as a very high value that allows the transfer of any amount of data between two



(a) Contact duration distribution (b) Degree distribution (c) Isolated consumers (d) Isolated consumers

Fig. 5. (a) CCDF of aggregated contact duration. (b)-(d) Dynamics of datasets.

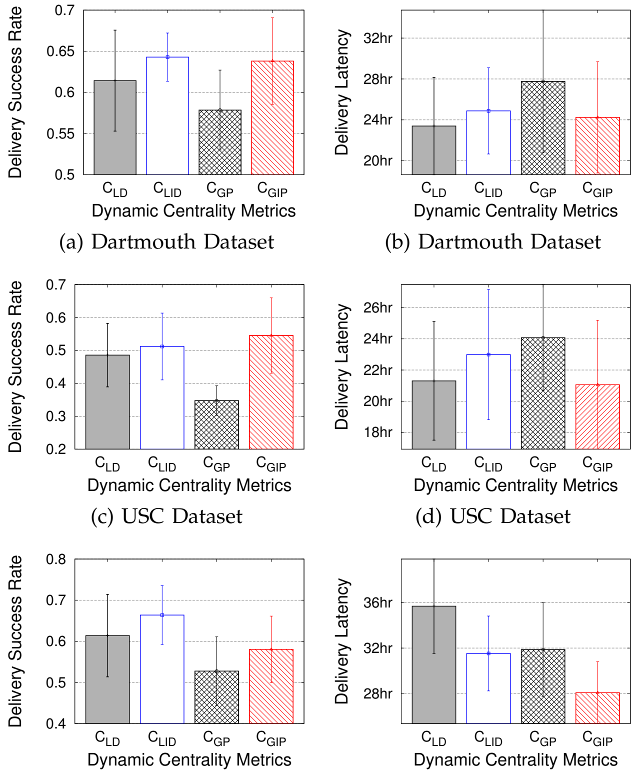
TABLE 4
Summary of mobility trace datasets

Dataset	Length (weeks)	Devices	Contacts (/device) /week)	Network type
Dartmouth	8	1138	12.456	WiFi
USC	8	1846	60.92	WiFi
SWIM	8	500	26.53	BT
D-SWIM	8	1500	29.85	BT
A-SWIM	8	1500	79.61	BT

users, if the allowed delivery latency is one day. Even in Dartmouth dataset more than 80% of users have more than 60 seconds contact duration per day. In contrast, SWIM has much lower aggregated contact duration.

We create a contact graph for all datasets where there exists an edge between two user nodes, if the two nodes are connected at least once. The degree distributions of the contact graphs are shown in Fig. 5b. SWIM users have highly skewed degree, where majority of the users have the similar number of contacts. D-SWIM also has similar degree distribution due to the fact that D-SWIM is generated keeping the SWIM density constant. Since A-SWIM is generated to show an overpopulated small area, its average degree is comparatively higher than the other datasets. In contrast, USC users have a distributed degree while the degree distribution of the Dartmouth lies in between those two extremes.

To evaluate the performance of proposed content dissemination strategies, we need to select *creator-consumer* groups. For this evaluation, *consumers* are selected randomly to evaluate the worst case scenario where the connectivity patterns of the users within the group have a minimum level of correlation. If there are *consumers* that never get in contact with any other friends of the selected group, it is not possible to opportunistically propagate content to those isolated *consumers*. Thus the amount of isolated *consumers* will have direct impact on the performance of the opportunistic dissemination. Fig. 5c illustrates that USC dataset contains a large number of isolated users compared to the other datasets. For the case of 100 *consumers* (Fig. 5d), nearly 15 of them are isolated in USC and only 5 of them are isolated in Dartmouth. In contrast, in all trials there are no isolated *consumers* in SWIM, it is less than 5 in D-SWIM. To this end, the five datasets are not similar and have properties that affect the performance of content replication in different aspects, representing a wide variety of social environments.



(a) Dartmouth Dataset (b) Dartmouth Dataset

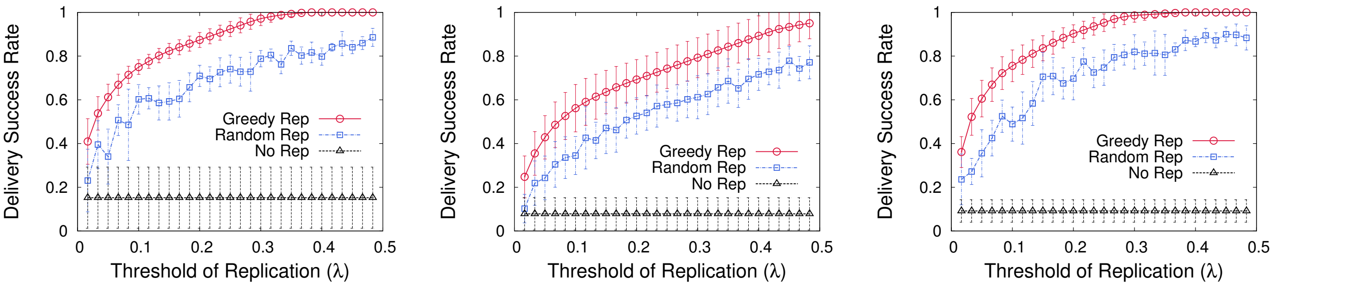
(c) USC Dataset (d) USC Dataset

(e) SWIM Dataset (f) SWIM Dataset

Fig. 6. Comparison of different dynamic centrality metrics. $\lambda = 10\%$

7.3 Simulation Setup

The datasets are divided into weeks as shown in Fig. 4. We select the set of *helpers* according to proposed algorithms during week k and evaluate the performance in terms of delivery success rate and delivery latency during week $k + 1$. We consider that the number of *consumers* for a *creator* to be 100 as it is the average size of a group of *friends* in the popular social networking service Facebook [44]. We select *consumers* randomly from the users in the dataset to evaluate the worst case scenario. Each user in a group is considered as a *creator* and we evaluate the performance for each *creator*. Therefore, for a given week, 100 simulations are carried out to obtain the average value for one particular performance metric. Further, all simulations are carried out varying the monitoring and evaluation periods through out the duration of the datasets. Each *creator* will generate a content of size 8.4MB, which is the median content size in YouTube [1] and the transfer rate among *consumers* are considered as uniform

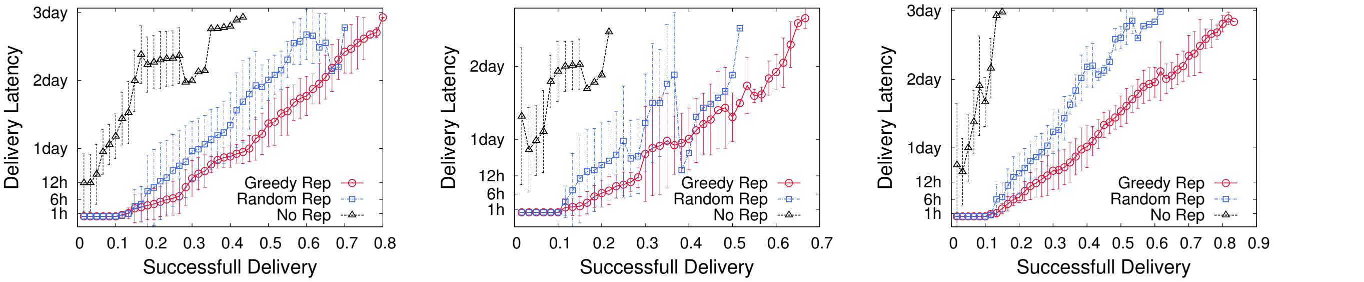


(a) Dartmouth Dataset

(b) USC Dataset

(c) SWIM-500 Dataset

Fig. 7. Delivery success rate against the threshold of replication (λ).



(a) Dartmouth Dataset

(b) USC Dataset

(c) SWIM-500 Dataset

Fig. 8. Delivery latency for specific delivery success rate when $\lambda = 0.1$.

and 2Mbps [25]. Hence, a *consumer* has to have aggregated contact duration $\alpha_T(u)$ of 33.6 seconds with the *creator* or any of the *helpers* or the *propagators* to completely download the content.

The content delivery deadline Δ of 3 days is considered to be the largest tolerable delay. The *Delivery Success Rate* is calculated as the ratio between number of successful deliveries and total number of *consumers*. The time lag between the content sharing by the *creator* and the content receiving time is considered as the *Delivery Latency*.

7.4 Evaluation of Dynamic Centrality Metrics

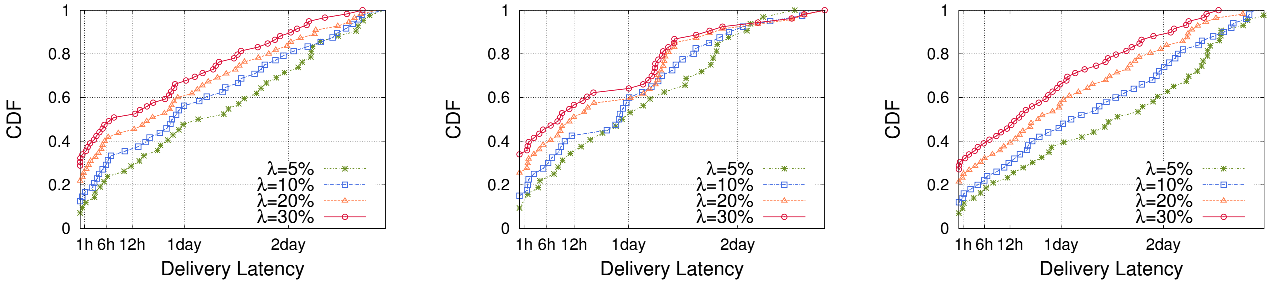
We compare the influence of different dynamic centrality metrics in *helper* selection that are defined in the Section 6.2. Fig. 6 shows the content delivery success rate and the mean content delivery latency, when the *helpers* are selected based on different centrality metrics according to the Algorithm 2. When we compared the two local centrality metrics, and showed that C_{LID} has a better delivery success rate with lower standard deviation than C_{LD} in all three datasets. Similarly, the improved global centrality metric C_{GIP} has better performance in terms of both delivery rate and latency compared to the naive C_{GP}. This is due to the fact that improved metrics, C_{LID} and C_{GIP}, consider the time dependency in connectivity patterns, which enhances the content propagation.

However, in some cases there is no significant difference between the performance of local and global metrics. All these general similarities are related to dynamics of the contact patterns among *consumers* in these environments. For instance, when the degree of the majority of the *consumers* are similar as shown in Fig. 5b for SWIM, the performance of a simple degree based local centrality metric becomes significant as depicted in Fig. 6e. In contrast, when the degree distribution is not skewed, the intelligent path based selection will perform better, similar to C_{GIP} performance for Dartmouth and USC. In particular, USC has the largest improvement of approximately 20% in coverage for C_{GIP} compared to C_{GP} because USC has the most distributed degree distribution. On the other hand, due to the large number of isolated consumers (Fig. 5c), USC does not have much gain in delivery latency. In SWIM, C_{GIP} has much lower delivery latency because it has the lowest number of isolated users. *Hence, the selection of the appropriate centrality metric to identify the most influential users is highly environment dependent and the appropriate centrality metric need to be identified by analysing the behavior of the users of the particular community.*

7.5 Evaluation of Delivery Performance

To evaluate the content delivery success rate and delivery latency of the proposed system independently from the dynamic centrality metrics, we use GREEDY-HELPER selection (Algorithm 1). We compare the delivery performance of the proposed system against the cases where (1) *No replication* is performed and (2) *Random replication*, which is the simplest way of selecting *helpers* without any knowledge about the contact patterns among *consumers*.

For all datasets in Fig. 7, it is evident that there is a significant gain in content delivery success rate and delivery latency compared to no replication. In fact, the greedy replication is always better than the random replication. Due to random selection of *consumers* in this evaluation, the *consumers* do not have high similarity in their mobility patterns. In real environments, *consumers* of the same content may also indicate that they belongs



(a) Dartmouth Dataset

(b) USC Dataset

(c) SWIM-500 Dataset

Fig. 9. The variation of CDF of delivery latency with λ values.

to the same community and therefore the performance can be expected to improve. Therefore, the results of the greedy replication can be considered as the worst case scenario. In Dartmouth (Fig. 7a), the delivery success rate is approximately 80% for 10% of content replication ($\lambda = 0.1$), while USC has a comparatively lower success rate of approximately 60%. After a certain level of replication, i.e. approximately $\lambda = 0.1$, the delivery rate shows linear improvement and further replication will not deliver content to any other *consumer* via opportunistic communication. This is because only isolated consumers will be selected for content replication. This threshold of replication can be considered as an effective upper bound for λ.

Fig. 8 shows the delivery time when using greedy replication, no replication, and random replication against the percentage of successful delivery. In Dartmouth (Fig. 8a), content replication delivers content to 40% of the *consumers* in less than 1 day, whereas the latency is almost 3 days if there is no initial content replication. All three datasets show similar behavior in terms of delivery latency. However, in general, the delivery latency for greedy replication is lower than the random replication.

Even though the proposed algorithms perform better than random *helper* selection, the results of random replication can not be ignored as it does produces comparable results in some cases without any prior knowledge of contact patterns. This behavior has also been reported in [13], [45]. For some groups of friends, we may not need an intelligent *helper* selection strategy, mainly because the connectivity graph is either too dense or too sparse. This can be addressed by considering the dynamics of the contact patterns to select the most suitable dynamic centrality metric.

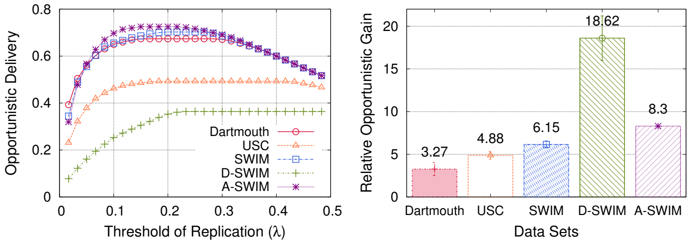
The cumulative distribution function of the delivery latency for successful deliveries is shown in Fig. 9. For all datasets, the probability of the delivery latency being less than 1 day is approximately 60% for $\lambda = 10\%$. In [43], it has been observed that 55% of Flickr content is uploaded after a lag of more than 1 day. Thus, we believe that the delivery latency resulting from the opportunistic communication is practical in such content dissemination applications. For applications/services that require a lower delay, it is possible to increase the threshold of content replication as shown in Fig. 9. The common pattern is that the delivery latency reduces with increasing λ. In Dartmouth and USC, there is a 20% improvement in reducing the delivery latency to less than 1 day when λ increases to 30%, and there is a 30% improvement in SWIM. However, the difference of latency for two consecutive λ values becomes smaller when λ increases. Again, similar to the delivery success rate, there is an effective upper bound for λ which does not increase the delivery latency significantly after that.

7.6 Opportunistic Delivery Gain

If the content is not received by the delivery deadline through low-cost networks, we assume that those *consumers* will download the content through other means of Internet connectivity such as via a cellular network. Thus, the portion of *Opportunistic Delivery* is calculated out of all content deliveries. Fig. 10a shows that the portion of opportunistic deliveries when using greedy replication. The portion of opportunistic deliveries increases with λ only for low values of λ. In Dartmouth, it is possible to deliver content for approximately 70% of the *consumers* via opportunistic communication with 10% of replication, compared to below 40% when there is no replication.

For the SWIM dataset, the results are closely related to the degree distribution of the datasets as shown in Fig. 5b. D-SWIM has the lowest performance because the simulation was extended by increasing the area and number of users while keeping the density of the network constant and equal to SWIM. Consequently a high level of replication is required to cover the same number of *consumers* as in SWIM. In contrast, when we increase the density in A-SWIM, it improves the opportunistic delivery percentage. Similarly, in USC, there is a large number of isolated users as shown in Fig. 5d, which decreases the overall density. Hence, the density of the contact graph has a considerable impact on the opportunistic delivery performance.

Fig. 10b summarizes the *Relative Opportunistic Gain* as the portion of opportunistic delivery with content replication and with no replication for all datasets. Even though D-SWIM has the lowest percentage of opportunistic delivery, it has the highest relative gain of 18.62 times because in D-SWIM, the percentage of opportunistic delivery when there is no replication is as low as 1.3%. Dartmouth dataset shows the lowest



(a)

(b)

Fig. 10. (a). Variation in opportunistic delivery with threshold of replication, (b). Achievable relative gain in opportunistic delivery for each dataset.

gain of 3.27 times, since it consists of well connected users compared to other environments. Thus, the results show that it is possible to significantly increase the one-hop opportunistic delivery with a low number of initial content replication, i.e. approximately less than 10% of the *consumers*. In addition, improvement in opportunistic transmission is proportional to the energy and communication cost which are additional incentives for mobile users.

8 Implementation

We extended our work described in [38], [39] to incorporate proposed system to show the practical feasibility. The integrated system referred to as **Yalut**[2] has been released on the Google Play Store. It consists of two main entities, (1) *Yalut mobile application* and (2) *Yalut cloud service*.

8.1 Yalut mobile application

The basic components of the mobile application and the Android app interface are shown in Fig. 11. The *Core Service* is the main component, which receives *Intents* from Android, such as the action of sharing a photo or accessing a link. It then orchestrates the appropriate flows to call the proper modules. When sharing a photo, *Core Service* calls *Connection Manager*, which manages peer-to-peer communication using modified BitTorrent communication protocols, namely Ttorrent [15] and libTorrent [16]. These were chosen to avoid license incompatibility and Android support issues.

Connection Manager first creates a torrent file for the content to be shared and then uploads the torrent file to the Yalut cloud service requesting a link to that particular torrent file. With the resulting link, *Core Service* requests *Social Network Manager* to send the "shared notification" through the selected social networking service, currently either Facebook, Google+, and Twitter.

After uploading the link to Yalut cloud service, the *Connection Manager* starts sending periodic messages to the tracker updating the contact information of the device. *Access Prediction Engine* monitors social networking feeds in the background and predicts the content that is likely to be consumed by the user based on the history of content access patterns. In the current implementation, the app periodically pre-fetches all content shared via the Yalut service. The content pre-fetching interval is set to 10 minutes by default and can also be set to any value through settings menu. Once a shared notification is identified by the *Access Prediction Engine* or by the user, the *Core Service* initiates the content downloading process first by fetching the related torrent file for that content from the Yalut cloud service. The torrent file contains the *announce-URL* of the tracker and the *Connection Manager* triggers the peer-to-peer content downloading process contacting the tracker. The *creator* or other online users who have the same torrent file will take part in forming the peer-to-peer network. The content sharing is set to perform only when devices are connected via WiFi networks to reduce the cost of usage. However, the user is again given the ability to change that to any network through the settings menu. We use the local peer discovery mechanism provided by libTorrent to give priority for friends who are connected to the same network when uploading content.

Besides that, *Context Info Collector* records context information such as connected networks and contact patterns that is used as input to the content replication algorithms described in Section 6. The collected data is uploaded to the Yalut cloud server once per day when connected to a WiFi networks to enable *helper* selection.

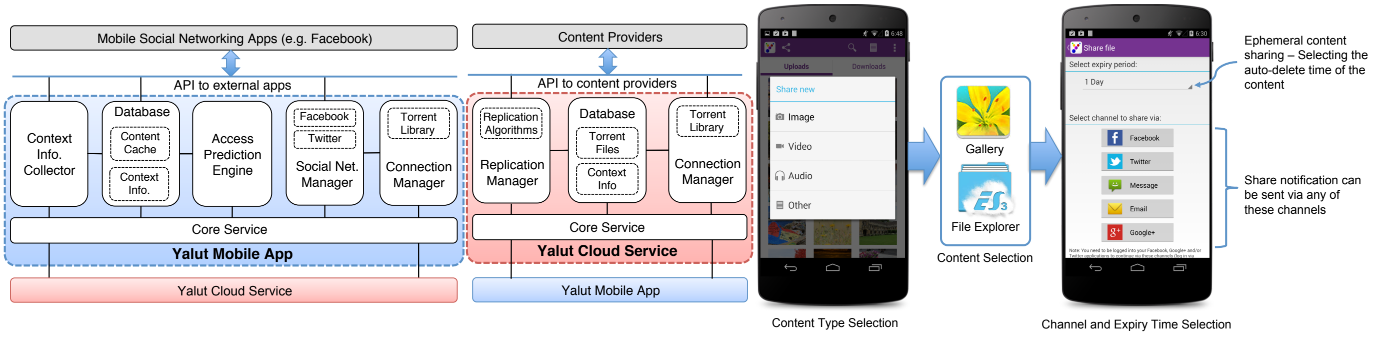
In real-life implementations, ultimately what matters is the user experience. Therefore, extra effort was taken to design the user interface of the app and the process of content sharing and downloading, from the user's perspective. Fig. 11c shows the flow of views that the user goes through when sharing content. When the sharing icon is invoked, the app asks to select the type of content that the user wishes to share. Then, the user is directed to either the photo, video or audio gallery, or to the file explorer based on the user input. After that, the app shows Yalut content sharing page, where the user can select the content expiry time and the service through which to send the shared notification. Once a service is selected, the user is directed to that service. For example, if Facebook is selected, the user is directed to the Facebook sharing page, in which the process will be similar to any usual content sharing in Facebook.

8.2 Yalut cloud service

Yalut server is the central entity that manages the communication between Yalut enabled devices and determines the *helpers* to be periodically used by *Replication Manager* (Fig. 11b). This is done using the context information collected by Yalut enabled devices as input to the replication algorithms proposed in Section 6. Replication Manager notifies all users when there is a change in the set of *helpers* of a particular user. The *Connection Manager* works as a torrent tracker and an indexer.

Thumbnail of the content or generic image that is stored in the Yalut server depending on the *creator's* preference, is displayed in the social networking feeds. Yalut keeps records of the owners of the shared content by storing

[2] http://www.yalut.com



(a) Yalut mobile application (b) Yalut cloud service (c) Yalut content sharing steps

Fig. 11. Components of Yalut cloud service and the workflow.

hashed social networking IDs, encrypted Yalut username and password. In addition, MD5-Hash of each content is also stored. This enables the prevention of sharing inappropriate and copyright violated content.

8.3 Preliminary survey results

The willingness of users to download and use Yalut type mobile app largely dependent on whether the user care for his/her data privacy. However, it is technically difficult to quantify the perception of privacy preservation of real users. Therefore, with the initial experimental trails of Yalut, we conducted a survey to get an idea about the social networking life of a sample of users. We had 32 female and 29 male respondents with diverse demographics. Almost everyone (98%) users use Facebook and 90% of them share their own photos with social media. However, 81% of respondents are concerned about use of personal data by third-parties. More importantly, if they could share content without *big-brother* like companies having access to data, 44% of them would be less concerned of their privacy, necessitating an application like Yalut for privacy-aware social networking. Further details of the survey results can be found at the Yalut Webpage.

9 Conclusion

We proposed a novel distributed content storage and dissemination architecture that uses intelligent content replication to addresses the issues of lack of trust and timeliness of delivery, and loss of privacy that has hindered the adoption and deployment of friend-to-friend content delivery. First, we showed that the content replication problem in opportunistic content dissemination architectures is NP-hard. Then, we developed a community based greedy algorithm for efficient content replication, taking advantage of routine behavioral patterns of social networking friends. Different dynamic centrality metrics were proposed as the greedy choice property to identify the most influential users within a community. Using both real world and synthetic datasets, we showed that our community based greedy algorithm results in delivery success rates of up to 80% with less than 10% replication. Furthermore, approximately 60% of the content can be delivered in less than one day. However, the results show that it is not possible to consistently identify most influential users within a community to optimise delivery as they are highly environment dependent.

Feasibility of the proposed system was demonstrated by implementing it on Android smart mobile devices. The system referred to as *Yalut*, which leverages of the existing social networking infrastructures for connecting and interacting users. It shows that is possible to use the proposed friend-to-friend content distribution architecture to provide improved user privacy, user control, and minimize their communications cost without compromising usability.

References

[1] A. Abhari and M. Soraya, "Workload generation for youtube," *Multimedia Tools and Applications*, vol. 46, no. 1, pp. 91–118, 2010.

[2] M. Barbera, J. Stefa, A. Viana, M. de Amorim, and M. Boc, "Vip delegation: Enabling vips to offload data in wireless social mobile networks," in *DCOSS'11*. IEEE, 2011, pp. 1–8.

[3] C. Boldrini, M. Conti, and A. Passarella, "Contentplace: social-aware data dissemination in opportunistic networks," in *Proceedings of the 11th international symposium on Modeling, analysis and simulation of wireless and mobile systems*. ACM, 2008, pp. 203–210.

[4] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott, "Impact of human mobility on opportunistic forwarding algorithms," *Mobile Computing, IEEE Transactions on*, vol. 6, pp. 606–620, 2007.

[5] G. Chen and F. Rahman, "Analyzing privacy designs of mobile social networking applications," in *IEEE/IFIP EUC'08*, vol. 2. IEEE, 2008, pp. 83–88.

[6] T. Chen, M. A. Kaafar, and R. Boreli, "The where and when of finding new friends: Analysis of a location-based social discovery network." in *ICWSM*, 2013.

[7] E. Cho, S. A. Myers, and J. Leskovec, "Friendship and mobility: user movement in location-based social networks," in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2011, pp. 1082–1090.

[8] Cisco, "Cisco visual networking index: Global mobile data traffic forecast update, 2011–2016," in *http://www.cisco.com*, feb' 2012.

[9] M. Cunche, M.-A. Kaafar, and R. Boreli, "I know who you will meet this evening! linking wireless devices using wi-fi probe requests," in *WoWMoM'12*, 2012, pp. 1–9.

[10] L. Cutillo, R. Molva, and T. Strufe, "Safebook: A privacy-preserving online social network leveraging on real-life trust," *Communications Magazine, IEEE*, vol. 47, no. 12, pp. 94–101, 2009.

[11] H. Falaki, R. Mahajan, S. Kandula, D. Lymberopoulos, R. Govindan, and D. Estrin, "Diversity in smartphone usage," in *MobiSys '10*, San Francisco, California, USA, 2010, pp. 179–194.

[12] R. Gracia-Tinedo, M. S. Artigas, and P. Garcia Lopez, "Analysis of data availability in f2f storage systems: When correlations matter," in *IEEE P2P'12*, sept. 2012, pp. 225 –236.

[13] B. Han, P. Hui, V. Kumar, M. Marathe, J. Shao, and A. Srinivasan, "Mobile data offloading through opportunistic communications and social participation," *Mobile Computing, IEEE Transactions on*, no. 99, pp. 1–1, 2011.

[14] http://joindiaspora.org.

[15] https://github.com/turn/ttorrent.

[16] http://www.libtorrent.org.

[17] P. Hui, J. Crowcroft, and E. Yoneki, "Bubble rap: Social-based forwarding in delay-tolerant networks," *Mobile Computing, IEEE Transactions on*, vol. 10, no. 11, pp. 1576–1589, 2011.

[18] S. Ioannidis, L. Massoulie, and A. Chaintreau, "Distributed caching over heterogeneous mobile networks," *SIGMETRICS Perform. Eval. Rev.*, vol. 38, no. 1, pp. 311–322, Jun. 2010.

[19] W. jen Hsu and A. Helmy, "CRAWDAD data set usc/mobilib (v. 2008-07-24)," Jul. 2008.

[20] V. Kann, "On the approximability of np-complete optimization problems," Ph.D. dissertation, Royal Institute of Technology Stockholm, 1992.

[21] L. Keller, A. Le, B. Cici, H. Seferoglu, C. Fragouli, and A. Markopoulou, "Microcast: Cooperative video streaming on smartphones," in *Proceedings of the 10th International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '12. New York, NY, USA: ACM, 2012, pp. 57–70.

[22] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2003, pp. 137–146.

[23] A. Khan, V. Subbaraju, A. Misra, and S. Seshan, "Mitigating the true cost of advertisement-supported free mobile applications," 2012.

[24] D. Kotz, T. Henderson, I. Abyzov, and J. Yeo, "CRAWDAD data set dartmouth campus," sep 2009.

[25] K. Lee, I. Rhee, J. Lee, S. Chong, and Y. Yi, "Mobile data offloading: how much can wifi deliver?" in *Proc. of the Co-NEXT '10*, Philadelphia, 2010, pp. 1–12.

[26] Y. Lee, Y. Ju, C. Min, S. Kang, I. Hwang, and J. Song, "Comon: cooperative ambience monitoring platform with continuity and benefit awareness," in *Proceedings of the 10th international conference on Mobile systems, applications, and services*. ACM, 2012, pp. 43–56.

[27] L. McNamara, C. Mascolo, and L. Capra, "Media sharing based on colocation prediction in urban transport," in *Proceedings of the 14th ACM international conference on Mobile computing and networking*. ACM, 2008, pp. 58–69.

[28] A. Mei and J. Stefa, "Swim: A simple model to generate small mobile worlds," in *INFOCOM'09, IEEE*, 2009, pp. 2106–2113.

[29] F. Nazir, J. Ma, and A. Seneviratne, "Time critical content delivery using predictable patterns in mobile social networks," *IEEE International Conference on Social Computing (SocialCom 2009) Workshop on Social Media Web*, 2009.

[30] T. Phe-Neau, M. D. de Amorim, and V. Conan, "The strength of vicinity annexation in opportunistic networking," in *IEEE NetSciCom*, 2013.

[31] A.-K. Pietilänen and C. Diot, "Dissemination in opportunistic social networks: the role of temporal communities," in *Proceedings of the thirteenth ACM international symposium on Mobile Ad Hoc Networking and Computing*. ACM, 2012, pp. 165–174.

[32] J. Pouwelse, P. Garbacki, J. Wang, A. Bakker, J. Yang, A. Iosup, D. Epema, M. Reinders, M. Van Steen, and H. Sips, "Tribler: a social-based peer-to-peer system," *Concurrency and Computation: Practice and Experience*, vol. 20, no. 2, pp. 127–138, 2008.

[33] J. Reich and A. Chaintreau, "The age of impatience: optimal replication schemes for opportunistic networks," in *Proceedings of the 5th international conference on Emerging networking experiments and technologies*. ACM, 2009, pp. 85–96.

[34] U. Sadiq, M. Kumar, and M. Wright, "Crisp: collusion-resistant incentive-compatible routing and forwarding in opportunistic networks," in *Proceedings of the 15th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems*. ACM, 2012, pp. 69–78.

[35] R. Sharma and A. Datta, "Supernova: Super-peers based architecture for decentralized online social networks," in *COMSNETS'12*. IEEE, 2012, pp. 1–10.

[36] R. Sharma, A. Datta, M. DeH'Amico, and P. Michiardi, "An empirical study of availability in friend-to-friend storage systems," in *IEEE P2P'11*. IEEE, 2011, pp. 348–351.

[37] M. Taghizadeh, K. Micinski, and S. Biswas, "Distributed cooperative caching in social wireless networks," *IEEE Transactions on Mobile Computing*, vol. doi: 10.1109/TMC.2012.66, 2013.

[38] K. Thilakarathna, X. Guan, and A. Seneviratne, "Demo: Yalut–user-centric social networking overlay," in *Proceedings of the 12th annual international conference on Mobile systems, applications, and services*. ACM, 2014, pp. 360–361.

[39] K. Thilakarathna, A. Karim, H. Petander, and A. Seneviratne, "MobiTribe: Enabling Device Centric Social Networking on Smart Mobile Devices," in *Proc. of IEEE SECON'13 Demostrations*, New Orleans, jun 2013, pp. 230–232.

[40] K. Thilakarathna, H. Petander, J. Mestre, and A. Seneviratne, "Enabling Mobile Distributed Social Networking on Smartphones," in *Proc. of ACM MSWiM'12*, oct 2012, pp. 357–366.

[41] ——, "MobiTribe: Cost Efficient Distributed User Generated Content Sharing on Smartphones," *IEEE Transactions on Mobile Computing*, vol. 13, no. 9, pp. 2058–2070, Sep 2014.

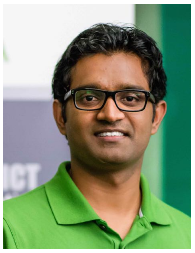
[42] K. Thilakarathna, A. C. Viana, A. Seneviratne, and H. Petander, "Mobile social networking through friend-to-friend opportunistic content dissemination," in *ACM MobiHoc'13*, Bangalore, India, 2013, pp. 263–266.

[43] I. Trestian, S. Ranjan, A. Kuzmanovic, and A. Nucci, "Taming user-generated-content in mobile networks via drop zones," in *INFOCOM'11*, 2011, pp. 2840–2848.

[44] J. Ugander, B. Karrer, L. Backstrom, and C. Marlow, "The anatomy of the facebook social graph," *Arxiv preprint arXiv:1111.4503*, 2011.

[45] J. Whitbeck, Y. Lopez, J. Leguay, V. Conan, and M. D. De Amorim, "Push-and-track: Saving infrastructure bandwidth through opportunistic forwarding," *Pervasive and Mobile Computing*, vol. 8, no. 5, pp. 682–697, 2012.

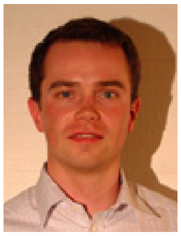
[46] N. Zhang, Y. Lee, M. Radhakrishnan, and R. K. Balan, "Gameon: P2p gaming on public transport," in *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '15. New York, NY, USA: ACM, 2015, pp. 105–119.



Kanchana Thilakarathna recently received his PhD is Electrical Engineering from the University of New South Wales. He is working as a Researcher at the Mobile Systems Research Group at NICTA. He has worked as a mobile network optimization Engineer for three years and as a research intern at INRIA-Saclay, France and NICTA, Australia. His current research interests are mobile content distribution and privacy preservation in mobile social networks.

Aline Carneiro Viana is a Senior Researcher at INRIA Saclay. Dr. Viana got her PhD in Computer Science from the University Pierre et Marie Curie in 2005. After holding a postdoctoral position at IRISA/INRIA Rennes, she joined INRIA Saclay in 2006. She was an Invited Researcher at the TKN Group of the Technical University of Berlin in 2010. Her research is primarily in data management and routing in wireless self-organized networks.

Aruna Seneviratne is the foundation Chair in Telecommunications and holds the Mahanakorn Chair of Telecommunications, and the leader of the Networks Reserach Group at NICTA. His current research interests are in mobile content distributions and preservation of privacy. He received his PhD in electrical engineering from the University of Bath , UK. He has held academic appointments at the University of Bradford, UK, Curtin University, and UTS.



Henrik Petander received his MSc in Electrical Engineering from Helsinki University of Technology in 2002 and his PhD in Computer Science from the same university in 2007. He has worked as a researcher at NICTA since 2005, and as a cofounder at Incoming, a mobile video company, since 2013. His research interests include mobile content delivery and networking in a heterogeneous mobile environment.