Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 826145, 14 pages
http://dx.doi.org/10.1155/2014/826145

Hindawi

Research Article

Analyzing the Impact of Storage Shortage on Data Availability in Decentralized Online Social Networks

Songling Fu,[1] Ligang He,[2,3] Xiangke Liao,[1] Kenli Li,[2] and Chenlin Huang[1]

[1] *School of Computer Science, National University of Defense Technology, Changsha 410073, China*
[2] *School of Information Science and Engineering, Hunan University, Changsha 410082, China*
[3] *Department of Computer Science, University of Warwick, Coventry CV4 7AL, UK*

Correspondence should be addressed to Ligang He; liganghe@gmail.com

Received 31 January 2014; Accepted 2 April 2014; Published 4 May 2014

Academic Editors: H.-C. Chao and Y. Pan

Maintaining data availability is one of the biggest challenges in decentralized online social networks (DOSNs). The existing work often assumes that the friends of a user can always contribute to the sufficient storage capacity to store all data. However, this assumption is not always true in today's online social networks (OSNs) due to the fact that nowadays the users often use the smart mobile devices to access the OSNs. The limitation of the storage capacity in mobile devices may jeopardize the data availability. Therefore, it is desired to know the relation between the storage capacity contributed by the OSN users and the level of data availability that the OSNs can achieve. This paper addresses this issue. In this paper, the data availability model over storage capacity is established. Further, a novel method is proposed to predict the data availability on the fly. Extensive simulation experiments have been conducted to evaluate the effectiveness of the data availability model and the on-the-fly prediction.

1. Introduction

In the last decade, online social networks (OSNs), such as Facebook [1], Twitter, and Sina Weibo [2], have gained extreme popularity with more than a billion users worldwide. OSNs allow a user to publish the data to all his friends in his friend circle.

Currently, the OSN platforms are typically centralized, where the users store their data in the centralized servers deployed by the OSN service providers. The service providers can utilize and analyze these data to know the users' private information, such as interest and personal affairs, and in the worst case may sell this information to the third party. Therefore, the current centralized online social networks (COSNs) have raised the serious concerns in privacy [3–6].

In order to address the data privacy issue, the decentralized online social networks (DOSNs) have been proposed recently [7–11]. Although the DOSN products [12] are not as popular and mature as the OSN products [1], DOSN is indeed under active research and development [13–17]. In DOSNs, in order to protect the data privacy the centralized servers are bypassed and the data published by a user are stored and disseminated only among the friend circle of the user [9, 10]. Although DOSNs can help protect the data privacy, maintaining data availability becomes a big challenge. This is because if a friend of the user is offline, the data stored in the friend cannot be accessed by other friends.

In order to achieve good data availability in DOSN, the data replication approach has been widely used. In this approach, a certain number of data replicas are created for each data item published by a user and these data replicas are stored in the user's friend circle. By doing so, if a friend is offline, the data in this offline friend node can be accessed through the replicated data stored in other friend nodes.

In the existing data replication work in DOSN, it is typically assumed that the friends of a user are always capable of contributing sufficient storage capacity to store all the published data [9, 14, 18]. This assumption is not ideal, especially in the current modern times. Nowadays, the users often use smart mobile devices, such as smart phones, to access the OSN services. The resources in the mobile devices are much more limited than the desktop computers used in

the "old fashioned" style of accessing OSNs. Moreover, the number of the friends in a friend circle is limited (typically less than 200) [19]. Therefore, it is desired to know what level of data availability can be achieved given the total storage capacity contributed by the friend circle. However, the existing work in DOSN has not yet conducted quantitative research in this aspect.

This paper aims to address the above issue and build a quantitative model to capture the relation between the total storage capacity contributed by the friends and the level of data availability in the DOSN.

The reason why we investigate the relation between the total storage capacity and data availability is because a data item is regarded as being available as long as it is stored in the online friend nodes in the DOSN, no matter which online friends the data replicas are stored in. The location of the data replicas does not directly affect the data availability but mainly imposes the impact in the following two aspects.

(i) Data accessing performance: due to, for example, the bandwidth and latency of the friends where the data are stored, other friends who are accessing the data may experience different performance.

(ii) The data maintenance overhead: when a friend goes offline, the data replicas on the friend have to be generated on other online friends. Various attributes of the friend, such as the storage capacity contributed by this friend, bandwidth, and latency, have impact. For example, if a friend offers the big storage capacity, then potentially more data have to be generated in other friends when this friend goes offline.

How to optimize data accessing performance and reduce data maintenance overhead is the work of the underlying data replication and placement strategies. This work is situated at the level of maintaining data availability. This is why this work mainly concerns the total storage size provided by the friends collectively. Following on from this work, we plan to work down the management levels in DOSN and develop the placement strategies for data replicas among the friends in DOSN.

In order to build the data availability model, we need to have deep understandings of the DOSN properties that are related to data availability. In this paper, we analyze these relevant properties and establish the probabilistic models for them. Further, the models for the individual properties are integrated to construct the data availability models. Further, a novel method is proposed to predict the level of data availability on the fly.

Using the data availability model developed in this paper, the DOSN designers can determine the average size of the storage pool that each friend should contribute for the published data, given the level of data availability that the DOSN desires to achieve. Moreover, in DOSN, the friends become online and offline dynamically; the data availability will drop when the number of online friends decreases. The on-the-fly prediction method can be used to conduct the real-time prediction for the level of data availability in the near future. The quantitative prediction results produced by the model can greatly help the data replication and storage policies make judicious decisions on the fly.

The rest of this paper is organized as follows. Section 2 discusses related work about analyses of OSN properties, the existing DOSN approaches, and data availability work. Section 3 states the problem which we try to address. Section 4 presents the data availability model over storage capacity. Section 5 presents the on-the-fly prediction model. Section 6 shows some case study. Section 7 conducts extensive experiments to verify our models and analyzes experimental results. Finally, we make conclusions.

2. Related Work

This section discusses the related work mainly in the following three aspects: (i) the existing work of analyzing the OSN properties, including both the characterizations of OSN networks and the analyses of user behaviors (Section 2.1), (ii) the existing research on DOSN, that is, the alternative approaches to decentralizing the OSNs (Section 2.2), and (iii) the existing studies on data availability in DOSN (Section 2.3). Moreover, this section also discusses the existing work in achieving data availability in grids and clouds (Section 2.4).

2.1. Analyses of the OSN Properties

2.1.1. Characterizations of OSN Networks. Some studies use the graphs to represent the OSN networks and investigate the graph structures of OSN, such as degree distribution, network diameter, and clustering property. They conduct the analyses through the crawled data gathered from popular OSN sites such as Facebook, Twitter, MySpace, Flickr, YouTube, LiveJournal, Cyworld, and orkut [13, 19–22]. It has been found that (i) OSNs manifest power-law, small-world, and scale-free properties; (ii) the social network is nearly fully connected; (iii) the neighborhoods of the users in the social graph contain the surprisingly dense structure, while the graph is sparse as a whole; (iv) most users have a moderate number of friends (less than 200). The findings about the number of friends will be used to design the simulation experiments in this paper.

2.1.2. Analyses of User Behaviours. The work in [23–27] studied the patterns of the user behaviors through the crawled or clickstream data. Jin et al. [23] conducted a comprehensive review about the user behavior in OSNs from several perspectives, including social connectivity and interaction among users, traffic activity, and the characteristics in mobile environments. Benevenuto et al. [24] collected the clickstream data over 12 days to study the characteristics of OSN sessions, including the accessing frequency, session durations, and total time spent on OSNs. Schneider et al. [25] focused on feature popularity, session characteristics, and the dynamics in the OSN sessions. Kwon and Wen [26] empirically examined how the individual characteristics affect the actual user acceptance of social network services. Yan et al. [27] studied the human behavior using the data obtained from the "Sina Microblog," which is one of the most

popular OSN sites in China. They found that the human activity patterns are heterogeneous and bursty and often follow the power-law distribution.

Since the existing research has revealed the dynamic characteristics about user behaviors, such as the distributions of online and offline durations, these will be used as the known parameters when we derive the data availability model and the on-the-fly prediction in this paper.

2.2. DOSN. To address the data privacy problem in COSNs, several decentralized approaches have been proposed [7–11]. Buchegger et al. [7] proposed a decentralized, peer-to-peer approach coupled with encryption. Yeung et al. [8] adopted a decentralized approach by using the URIs as the identifiers throughout, which can provide the same (or even higher) level of user interaction as with many of the current popular OSN sties. Tandukar and Vassileva [9] also proposed a decentralized OSN. With this approach, users can maintain the control over their data to protect their data privacy and forward the social data selectively to reduce the irrelevant data among the users. None of these approaches only stores the data published by a user in his friend circle.

There is another type of DOSNs [10, 11], known as friend-to-friend storage systems, which focus on providing the data storage services for all participants. Li and Dabek [10] argued that a node should choose its neighbors where the data are stored based on existing social relationships instead of randomly. Sharma et al. [11] find that the limitation of storing data only on friends has a marked impact on the data availability. They showed that the problem of obtaining maximal availability while minimizing redundancy is NP complete and proposed greedy data placement heuristics to improve the data availability. Our data availability model and the on-the-fly prediction can be integrated into these existing DOSNs; for example, the quantitative results produced by our models can be used to help make the data replication and/or data storage decisions.

2.3. Data Availability in DOSN. Because of the requirement of protecting data privacy, the data published by a user are only stored in his friend circle in the DOSN. Consequently, data availability is one of the biggest challenges in DOSNs. The existing work in improving data availability mainly focuses on designing smart data replication and data storage policies.

Shakimov et al. [28] propose three schemes for storing the data in DOSNs: the cloud-based scheme, the desktop-based scheme, and the hybrid scheme combining the above two. In the cloud-based scheme, the data will be stored in the cloud servers. In the desktop-based scheme, two mechanisms may be used: (i) the data replicas are encrypted when they are stored in potentially untrusted hosts; (ii) the users take advantage of the trust embedded in the social network to store the data replicas on trustworthy friends. The drawbacks of these mechanisms come from the complexity and overhead in the encryption key or trust management.

The approach proposed by Koll et al. [18] exchanges the recommendations among the socially related nodes in order to effectively distribute a user's data replicas among the eligible nodes carefully selected in the OSN.

In the approach developed by Olteanu and Pierre [14], the preferences are given to the nodes when it comes to selecting the nodes for storing the data (and their replicas) published by a user [14]. The online friends of the user have the highest priority. When all friends are offline, the data are then stored in the nodes which are not in the user's friend circle.

Buchegger et al. designed a two-tiered DOSN architecture (PeerSoN) [7]. One tier serves as a look-up service which is implemented by OpenDHT. The second tier consists of the peers and contains the user data. When a user is offline, all his data will be stored across the whole network.

Cutillo et al. [29] propose a P2P-based DOSN (Safebook), in which each node is accessible through the so-called shells. The profile data is mirrored and stored in a subset of a node's direct contacts, which form the so-called innermost shell. The data retrieval requires traversing the shells along a path of the nodes that are online and are friends with each other.

Tegeler et al. [30] propose an approach called Gemstone. Gemstone protects the user's privacy by encrypting all data using ABE and stores the user's data in the so-called data holding agents (DHAs). If a DHA itself is offline, the data have to be passed to the DHAs of this offline DHA.

All the above existing work about data availability focuses on how to store the data replicas so that they are still accessible when the users or certain friends of the users are offline. They all implicitly assume that the friends are always able to contribute the adequate storage capacities to store the replicated data.

2.4. Data Availability in Grids and Clouds. We also studied the existing work in achieving data availability in grids and clouds. Amjad et al. [31] surveyed the dynamic replication strategies for improving data availability in data grids. Kossmann et al. [32] proposed a modular cloud storage system. Zeng et al. [33] studied the cloud storage architecture and then pointed out the key techniques.

However, we found that the focuses and the considerations in achieving data availability in grids and clouds are quite different from those in DOSN. One of the biggest differences is that the data replication mechanisms in grids or clouds do not treat the total storage capacity as a limitation, although some studies considered the case where the storage capacity of individual nodes in a grid system is limited. Namely, these studies all explicitly or implicitly assume that the total storage space in grids or clouds is always sufficient to store the data replicas. This assumption is reasonable for grids and clouds because of the scale of such systems. However, it is not always true for DOSN due to the aforementioned facts that (1) smart mobile devices, whose storage capacity is limited, are often used in DOSN and (2) the number of friends in a friend circle is also limited.

3. Problem Statement

Figure 1 illustrates the data availability problem. In Figure 1, the user publishes the data at a series of time points along the

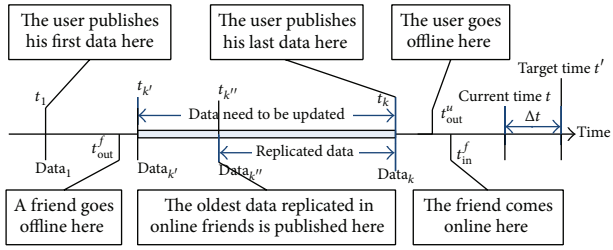


Figure 1: The illustration of the data availability problem.

time line. Assume t_1 is the first time point when he publishes the data, $Data_1$, after he comes online, and t_k is the last time point the user publishes the data, $Data_k$, before he goes offline at the time point t^u_{out}. Now let us consider one of the friends in the user's friend circle. Assume that the friend goes offline at time point t^f_{out} just before the user publishes $Data_{k'}$ (and after the user publishes $Data_{k'-1}$) and then comes online at time point t^f_{in} after the user goes offline. Therefore, $Data_{k'}$ to $Data_k$ are the data that the friend missed when he is offline and consequently need to be updated when he comes online. Since the user is already offline, the friend can only update the missed data from other online friends where the data replicas are stored. Note that if the friend comes online before the user goes offline, the friend can update all missed data from the user directly. Therefore, data availability is not a problem under this circumstance.

When a friend comes online, assume that the total amount of the data that the friend tries to update is D_{update}. Out of D_{update}, the amount of data that are stored in online friends of the user is D_{stored}. The level of data availability (denoted by DA) is defined as

$$\mathrm{DA} = \frac{D_{stored}}{D_{update}}. \tag{1}$$

The data replication frameworks typically work in the following way [10, 18, 34]. When the user publishes a data item, a certain number of data replicas are created and stored in the storage pools of the selected friends of the user. When a friend goes offline the data replicas which are stored in this friend will be recreated and stored on other online friends to maintain fixed number of data replicas for each data item. If the size of the storage pools is unlimited, the new data will just be added to the friend's storage pool. If the storage pool is limited and the pool is already full, the oldest data in the storage pool will be replaced with the new data. Therefore, the size of the storage pool will determine what period of data is stored in the pool, which affects the data availability of the DOSN. Consider Figure 1 again; for example, if the storage pool in the friends is limited and can only store the data published from t_k back to $t_{k''}$, then the data earlier than $t_{k''}$ are not available when the friend comes online at t^f_{in}.

One aim of this paper is to establish the data availability model to capture the relation between the level of data availability and the total size of the storage pools contributed by the friends. This is presented in Section 4.

Now consider a time point t' after the current time t. The other aim of this paper is to predict the level of data availability at t' on the fly, which is presented in Section 5. This prediction is very useful for the data replication or storage policies to make judicious decisions dynamically.

The notations that are used in the derivations of the data availability models are introduced as Table 1.

4. The Data Availability Model over Storage Capacity

As discussed in Section 3, the total size of the storage pool contributed by a user's friends (denoted by SS) can determine the period of the published data stored in the storage pool. t_{tl} denotes the publishing time of the oldest data stored in the storage pool (i.e., $t_{k''}$ in Figure 1), and t^u_{out} denotes the time when the user goes offline. Then $[t_{tl}, t^u_{out}]$ is the period of the published data stored in the storage pool. This section first determines t_{tl} (Section 4.1) and then presents the method of establishing the relation between SS and the DA of the data published by the user (Section 4.2).

4.1. Calculating t_{tl}. In order to determine t_{tl}, the size of the data published by the user has to be calculated first. $X(t_{pu})$ denotes the number of times that the user publishes the data in the time duration t_{pu}. $X(t_{pu})$ is a discrete random variable. $P_{pu}(x(t_{pu}))$ denotes the probability density function (pdf) of $X(t_{pu})$. a denotes the average size of the data published by the user each time. $S(t_{pu})$ denotes the total size of the data published by the user in t_{pu}. Clearly, $S(t_{pu}) = aX(t_{pu})$. Therefore, the pdf of $S(t_{pu})$, denoted by $S_{pu}(s(t_{pu}))$, can be determined by (2) and the expectation of $s(t_{pu})$ can be calculated by (3) as follows:

$$S_{pu}\left(s\left(t_{pu}\right)\right) = a \cdot P_{pu}\left(x\left(t_{pu}\right)\right), \tag{2}$$

$$E\left[S\left(t_{pu}\right)\right] = a \cdot E\left[X\left(t_{pu}\right)\right] = a \cdot \sum_{x=1}^{+\infty} x \cdot P_{pu}\left(x\left(t_{pu}\right)\right). \tag{3}$$

The publishing time of the oldest data stored in the storage pool, t_{tl}, can be calculated using (4) given SS, where k is the replication degree in the OSN, that is, the number of replicas created for each data item. Consider

$$E\left[S\left(t^u_{out} - t_{tl}\right)\right] \cdot k = \mathrm{SS}. \tag{4}$$

4.2. Establishing the Relation between DA and SS. When a friend comes online at t^f_{in} (as in Figure 1) and his last logout time (denoted by t^f_{out}) is no earlier than t_{tl}, the friend can update all the data missed during his offline duration from other online friends. Namely, DA for a friend coming online at t^f_{in}, denoted by $\mathrm{DA}(t^f_{in}, t^f_{out})$, is 100% in this case. When t^f_{out} is earlier than t_{tl}, the data published in $[t^f_{out}, t_{tl}]$ are not available to the friend. Therefore, DA in this case equals the

Table 1: The notations that are used in the derivation.

Notations	Descriptions
v_t	The user
N	The number of the user's friends
t	Current time point
t'	Target time point in near future, $t' = t + \Delta t$, where Δt is a time duration after t. We want to predict the state of the DOSN at the time point t'
t^u_{out}	The time point at which the user v_t goes offline
V_{on}	The set of all online users in the friend circle of the user v_t at current time t
N_{on}	The number of online users in set V_{on}
V_{off}	The set of all offline users in the friend circle of the user v_t at current time t
N_{off}	The number of offline users in set V_{on}
$t^{\text{on}}_{\text{in\_}i}$	The latest login time of the online user v_i in V_{on} before current time t
$t^{\text{on}}_{\text{out\_}i}$	The first logout time of the online user v_i in V_{on} after current time t
$t^{\text{off}}_{\text{out\_}j}$	The latest logout time of the offline user v_j in V_{off} before current time t
$t^{\text{off}}_{\text{in\_}j}$	The first login time of the offline user v_j in V_{off} after current time t
E_{login} E_{logout}	The login and logout events, respectively. When any of these two events occurs, the state of a user changes from *OFFLINE* to *ONLINE* or from *ONLINE* to *OFFLINE*
t_{on} $f_{\text{on}}(t_{\text{on}})$ $F_{\text{on}}(t_{\text{on}})$	The time duration of a user being online continuously (i.e., the time duration from an E_{login} event to the following E_{logout} event), which is a random variable and whose probability density function and probability distribution function are denoted by $f_{\text{on}}(t_{\text{on}})$ and $F_{\text{on}}(t_{\text{on}})$, respectively
t_{off} $f_{\text{off}}(t_{\text{off}})$ $F_{\text{off}}(t_{\text{off}})$	The time duration of a user being offline, which is also a random variable and whose probability density function and probability distribution function are denoted by $f_{\text{off}}(t_{\text{off}})$ and $F_{\text{off}}(t_{\text{off}})$, respectively
x $P_{\text{pu}}(x,t)$	The number of times that the user publishes the data, which is a discrete random variable and whose probability density function in a duration t is denoted by $P_{\text{pu}}(x,t)$
a	The statistical average size of the data published by the user each time. a is a constant
k	The replication degree, that is, the number of replicas created for each data item
t_{tl}	The publishing time of the oldest data stored in the storage pool
SS	The total storage capacity contributed by all online friends
S	The maximum storage capacity that each friend is able to contribute

proportion of the data that are published in $[t_{tl}, t^u_{\text{out}}]$ to those in $[t^f_{\text{out}}, t^u_{\text{out}}]$. In summary, $\text{DA}(t^f_{\text{in}}, t^f_{\text{out}})$ can be calculated using

$$\text{DA}\left(t^f_{\text{in}}, t^f_{\text{out}}\right) = \begin{cases} 100\% & t^f_{\text{out}} \geq t_{tl} \\ \dfrac{E\left[S\left(t^u_{\text{out}} - t_{tl}\right)\right]}{E\left[S\left(t^u_{\text{out}} - t^f_{\text{out}}\right)\right]} \cdot 100\% & t^f_{\text{out}} < t_{tl}. \end{cases} \quad (5)$$

t_{off} denotes the time duration of a friend being offline continuously. $f_{\text{off}}(t_{\text{off}})$ denotes the pdf of t_{off}. The probability that a friend went offline at t^f_{out} and then comes online at t^f_{in} is $f_{\text{off}}(t^f_{\text{in}} - t^f_{\text{out}})dt^f_{\text{out}}$ and the corresponding $\text{DA}(t^f_{\text{in}}, t^f_{\text{out}})$ is obtained by (5). Then, DA at time point t^f_{in} can be expressed by

$$\int_{t^u_{\text{out}}}^{0} f_{\text{off}}\left(t^f_{\text{in}} - t^f_{\text{out}}\right) \cdot \text{DA}\left(t^f_{\text{in}}, t^f_{\text{out}}\right) dt^f_{\text{out}}. \quad (6)$$

$\text{DA}_{[t^u_{\text{out}},\, h]}$ denotes the expectation of DA over the time duration between t^u_{out} and t^f_{in}, where h is the duration between the user's two consecutive logins (the work in [25, 35, 36] has presented the method to obtain the value of h). $\text{DA}_{[t^u_{\text{out}},\, h]}$ can be calculated by (7), where $f_{\text{at}}(t^f_{\text{in}})$ is the probability density function that a friend comes online at time t^f_{in}:

$$\text{DA}_{[t^u_{\text{out}},\, h]} = \int_{t^u_{\text{out}}}^{h} f_{\text{at}}\left(t^f_{\text{in}}\right) \cdot \int_{t^u_{\text{out}}}^{0} f_{\text{off}}\left(t^f_{\text{in}} - t^f_{\text{out}}\right) \text{DA}\left(t^f_{\text{in}}, t^f_{\text{out}}\right) dt^f_{\text{out}} dt^f_{\text{in}}. \quad (7)$$

$\text{DA}_{[0, t^u_{\text{out}}]}$ denotes the expectation of DA over the time duration between 0 and t^u_{out}. Since the user is online between 0 and t^u_{out}, DA is 100% over the time duration between 0 and t^u_{out}; that is, (8) holds:

$$\text{DA}_{[0, t^u_{\text{out}}]} = 100\%. \quad (8)$$

t_{on} denotes the time duration of a friend being online continuously. $f_{\text{on}}(t_{\text{on}})$ denotes the pdf of t_{on}. DA of the data published by the user under the given value of h, denoted by $\text{DA}(h)$, can be calculated by combining (7) and (8) as follows:

$$\text{DA}(h) = \int_0^h f_{\text{on}}\left(t^u_{\text{out}}\right) \cdot \left(\frac{t^u_{\text{out}}}{h} \cdot \text{DA}_{[0, t^u_{\text{out}}]} + \frac{h - t^u_{\text{out}}}{h} \cdot \text{DA}_{[t^u_{\text{out}},\, h]}\right) dt^u_{\text{out}}. \quad (9)$$

$h = t_{\text{on}} + t_{\text{off}}$ is also a random variable. $f_H(h)$ denotes the probability density function of h, which can be derived from the probability density functions of t_{on} and t_{off} and has also been studied in the literature [26, 37].

Therefore, DA of the data published by the user can be finally calculated using

$$\text{DA} = \int_0^{\infty} \text{DA}(h) \cdot f_H(h)\, dh. \quad (10)$$

As can be seen from (9), DA is a function over $\mathrm{DA}_{[t^u_{\mathrm{out}},\, H]}$, which is in turn a function over $\mathrm{DA}(t^f_{\mathrm{in}}, t^f_{\mathrm{out}})$ (shown in (7)). $\mathrm{DA}(t^f_{\mathrm{in}}, t^f_{\mathrm{out}})$ is the function over t_{tl} (shown in (5)). As shown in (4), t_{tl} can be calculated from SS. Therefore, we have now established the function of DA over SS.

5. Predicting the Data Availability on the Fly

Using the method presented in Section 4, we can calculate SS required to achieve the desired DA of the data published by the user. Note that SS is the total size of the storage pool contributed by all online friends of the user. The friends log in and out dynamically and therefore the number of online friends varies over time. When the number of online friends decreases, the size of the individual storage pool contributed by each online friend has to be increased in order to maintain the desired DA. The existing work in the literature often assumes that the friends of a user are always capable of contributing sufficient storage capacity for the replicated data published by the user. Consequently, there is little work yet in the literature investigating the impact of the friends' dynamic behaviors (i.e., dynamic login and logout) on DA. However, as we have discussed in the introduction section, it is not always acceptable to assume that the friends are willing and able to contribute unlimited storage capacity in the nowadays OSNs. In this paper, we assume that the maximum storage capacity that each friend is able to contribute is S. When the required SS exceeds the total storage capacity contributed by all online friends, the DA will drop. Due to the friends' dynamic behaviors, it is very useful to be able to predict the DA on the fly. This section addresses this issue. Consider Figure 1 again. Assume the current time is t. The problem of the on-the-fly prediction of DA is to predict the DA at a future time point $t'(t' > t)$.

According to the discussions above, the key of predicting DA is to predict the number of online friends. At the current time t, we know how many friends are online or offline. We can predict the number of friends who are online at a future time t', if we can predict the following two parameters: (i) how many of the friends who are online at time t do not change their states from online to offline before or at t', and (ii) how many of the friends who are offline at time t change their states to online before or at t'. The methods of predicting the above two parameters are presented in Sections 5.1 and 5.2, respectively. Section 5.3 combines the results obtained in Sections 5.1 and 5.2 to predict the number of online friends and further predict the DA at time t'.

5.1. Predicting the Number of the Friends Who Are Online at t and Do Not Change to Offline before or at t'. Given an online friend v_i at time t, we can know the time point at which the friend logged in (i.e., became online), which is denoted by $t^{\mathrm{on}}_{\mathrm{in}\_i}$. The probability that friend v_i does not change to offline before t' equals the probability that v_i will only log out after t' (i.e., v_i's logout time, denoted by $t^{\mathrm{on}}_{\mathrm{out}\_i}$, is greater than t'). The probability, denoted by $p^{\mathrm{on}}_{\mathrm{out}\_i}(t^{\mathrm{on}}_{\mathrm{out}\_i} > t')$, in turn equals the probability that v_i's online duration is greater than $(t' - t^{\mathrm{on}}_{\mathrm{in}\_i})$ under the condition that v_i's online duration is no less than $(t - t^{\mathrm{on}}_{\mathrm{in}\_i})$, which can be computed using the conditional probability shown in (11). The condition of $(t_{\mathrm{on}} \geq t - t^{\mathrm{on}}_{\mathrm{in}\_i})$ in (11) reflects the fact that v_i has been staying online for the duration of $(t - t^{\mathrm{on}}_{\mathrm{in}\_i})$:

$$\begin{aligned} & p^{\mathrm{on}}_{\mathrm{out}\_i}\left(t^{\mathrm{on}}_{\mathrm{out}\_i} > t'\right) \\ &= p_{\mathrm{on}}\left(\left(t_{\mathrm{on}} > t' - t^{\mathrm{on}}_{\mathrm{in}\_i}\right) \mid \left(t_{\mathrm{on}} \geq t - t^{\mathrm{on}}_{\mathrm{in}\_i}\right)\right) \\ &= \frac{p_{\mathrm{on}}\left(t > t' - t^{\mathrm{on}}_{\mathrm{in}\_i}\right)}{p_{\mathrm{on}}\left(t > t - t^{\mathrm{on}}_{\mathrm{in}\_i}\right)} \\ &= \frac{1 - F_{\mathrm{on}}\left(t' - t^{\mathrm{on}}_{\mathrm{in}_i}\right)}{1 - F_{\mathrm{on}}\left(t - t^{\mathrm{on}}_{\mathrm{in}_i}\right)}. \end{aligned} \tag{11}$$

V_{on} and N_{on} denote the set and the number of all online friends at time t, respectively. Then the number of the friends in V_{on} who are still online at time t' can be predicted using

$$\sum_{i=1}^{N_{\mathrm{on}}} p^{\mathrm{on}}_{\mathrm{out}\_i}\left(t^{\mathrm{on}}_{\mathrm{out}\_i} > t'\right). \tag{12}$$

5.2. Predicting the Number of the Friends Who Are Offline at t and Change the States to Online before or at t'. The method of predicting the number of the friends who are offline at t and change the states to online before or at t' is similar to that presented in Section 5.1:

$$\begin{aligned} & p^{\mathrm{off}}_{\mathrm{in}\_j}\left(t^{\mathrm{off}}_{\mathrm{in}\_j} \leq t'\right) \\ &= p_{\mathrm{off}}\left(\left(t_{\mathrm{off}} \leq t' - t^{\mathrm{off}}_{\mathrm{out}\_j}\right) \mid \left(t_{\mathrm{off}} \geq t - t^{\mathrm{off}}_{\mathrm{out}\_j}\right)\right) \\ &= \frac{p_{\mathrm{off}}\left(t - t^{\mathrm{off}}_{\mathrm{out}\_j} \leq t_{\mathrm{off}} \leq t' - t^{\mathrm{off}}_{\mathrm{out}\_j}\right)}{p_{\mathrm{off}}\left(t_{\mathrm{off}} \geq t - t^{\mathrm{off}}_{\mathrm{out}\_j}\right)} \\ &= \frac{F_{\mathrm{off}}\left(t' - t^{\mathrm{off}}_{\mathrm{out}\_j}\right) - F_{\mathrm{off}}\left(t - t^{\mathrm{off}}_{\mathrm{out}\_j}\right)}{1 - F_{\mathrm{off}}\left(t - t^{\mathrm{off}}_{\mathrm{out}\_j}\right)}. \end{aligned} \tag{13}$$

Given an offline friend v_j at time t, we can know the time when v_j logged off, denoted by $t^{\mathrm{off}}_{\mathrm{out}\_j}$. The probability that v_j changes the state to online before or at t' equals the probability that v_j's login time, $t^{\mathrm{off}}_{\mathrm{in}\_j}$, is no later than t'. The probability, denoted by $p^{\mathrm{off}}_{\mathrm{in}\_j}(t^{\mathrm{off}}_{\mathrm{in}\_j} \leq t')$, in turn equals the probability that v_j's offline duration is smaller than $(t' - t^{\mathrm{off}}_{\mathrm{out}\_j})$ under the condition that v_j's offline duration is no less than $(t - t^{\mathrm{off}}_{\mathrm{out}\_j})$, which can be calculated using (13).

V_{off} and N_{off} denote the set and the number of all offline friends at time t, respectively. Then the number of the friends in V_{off} who change the states to online before or at time t' can be predicted using

$$\sum_{j=1}^{N_{\mathrm{off}}} p^{\mathrm{off}}_{\mathrm{in}\_j}\left(t^{\mathrm{off}}_{\mathrm{in}\_j} \leq t'\right). \tag{14}$$

5.3. Predicting the Number of Online Friends and the DA at t'. $N_{\text{on}}(t')$ denotes the number of online friends at t'. $N_{\text{on}}(t')$ can be calculated by (15) by combining (12) and (14) as follows:

$$\begin{aligned} N_{\text{on}}\left(t'\right) &= \sum_{i=1}^{N_{\text{on}}} p_{\text{out\_}i}^{\text{on}}\left(t_{\text{out\_}i}^{\text{on}} > t'\right) + \sum_{j=1}^{N_{\text{off}}} p_{\text{in\_}j}^{\text{off}}\left(t_{\text{in\_}j}^{\text{off}} \le t'\right) \\ &= \sum_{i=1}^{N_{\text{on}}}\left(\frac{1 - F_{\text{on}}\left(t' - t_{\text{in\_}i}^{\text{on}}\right)}{1 - F_{\text{on}}\left(t - t_{\text{in\_}i}^{\text{on}}\right)}\right) \\ &\quad + \sum_{j=1}^{N_{\text{off}}}\left(\frac{F_{\text{off}}\left(t' - t_{\text{out\_}j}^{\text{off}}\right) - F_{\text{off}}\left(t - t_{\text{out\_}j}^{\text{off}}\right)}{1 - F_{\text{off}}\left(t - t_{\text{out\_}j}^{\text{off}}\right)}\right). \end{aligned} \tag{15}$$

S is the maximum storage capacity that each friend is able to contribute. Then the total storage capacity contributed by all online friends at time t' is $(S \cdot N_{\text{on}}(t'))$. Using the method presented in Section 4, the DA at t' can be determined.

6. Case Study

When we derive the DA model over storage capacity and the on-the-fly prediction of DA in Sections 4 and 5, we used the generic form of the probability distribution for online and offline durations (i.e., $f_{\text{on}}(t_{\text{on}})$ and $f_{\text{off}}(t_{\text{off}})$) as well as for the data publishing pattern, that is, the number of times that the user publishes the data in a given time duration (i.e., $P_{\text{pu}}(x,t)$). However, it has been shown that the online and offline durations may follow the power-law distribution or the exponential distribution [35, 37, 38] and that the data publishing pattern may follow the Poisson process [37]. In this section, we conduct a few case studies by substituting the generic form of the probability distribution for the power-law, the exponential, and the Poisson distribution. In fact, any probability distributions can be used in the proposed models. Even if the mathematical derivations may not be carried out with some probability distributions, the *Mathematica* software [39] can be used to calculate the model results.

6.1. Poisson Distribution. The data publishing pattern may follow the Poisson process [35]. If $X(t_{\text{pu}})$ follows the Poisson distribution with the parameter λ_{pu}, then we have (16). Consequently, $E[X(t_{\text{pu}})]$ can be calculated using (17), as follows:

$$P_{\text{pu}}\left(x\left(t_{\text{pu}}\right)\right) = e^{-\lambda_{\text{pu}} t_{\text{pu}}} \frac{\left(\lambda_{\text{pu}} t_{\text{pu}}\right)^x}{x!}, \tag{16}$$

$$E\left[X\left(t_{\text{pu}}\right)\right] = \lambda_{\text{pu}} t_{\text{pu}}. \tag{17}$$

Further, (3) can be transformed to

$$E\left[S\left(t_{\text{pu}}\right)\right] = a \cdot E\left[X\left(t_{\text{pu}}\right)\right] = a\lambda_{\text{pu}} t_{\text{pu}}. \tag{18}$$

With (18), (4) becomes

$$ak\lambda_{\text{pu}}\left(t_{\text{out}}^{u} - t_{tl}\right) = \text{SS}. \tag{19}$$

Therefore, given the storage capacity SS, the replication degree k, and the logout time of the user t_{out}^{u}, the publishing time of the oldest data stored in the storage pool, t_{tl}, can be calculated using

$$t_{tl} = t_{\text{out}}^{u} - \frac{\text{SS}}{ak\lambda_{\text{pu}}}. \tag{20}$$

Moreover, with (18), (5) then becomes

$$\text{DA}\left(t_{\text{in}}^{f}, t_{\text{out}}^{f}\right) = \begin{cases} 100\% & t_{\text{out}}^{f} \ge t_{tl} \\ \dfrac{t_{\text{out}}^{u} - t_{tl}}{t_{\text{out}}^{u} - t_{\text{out}}^{f}} \cdot 100\% & t_{\text{out}}^{f} < t_{tl}. \end{cases} \tag{21}$$

6.2. Power-Law Distribution. If the offline duration, t_{off}, follows the power-law distribution with parameter λ_{off}, then we have (22), where $c = (\lambda_{\text{off}} - 1){t_{\text{min}}}^{\lambda_{\text{off}}-1}$ given the minimal duration t_{min} [40]:

$$f_{\text{off}}\left(t_{\text{off}}\right) = c \cdot {t_{\text{off}}}^{-\lambda_{\text{off}}}. \tag{22}$$

We now show how to use the power-law distribution to derive the on-the-fly prediction for the number of online friends, which is obtained in Section 5 through (11), (13), and (15).

Equation (11) can be further derived with the power-law distribution to obtain

$$\begin{aligned} p_{\text{out\_}i}^{\text{on}}\left(t_{\text{out\_}i}^{\text{on}} > t'\right)_{\text{pl}} &= \frac{1 - F_{\text{on}}\left(t' - t_{\text{in\_}i}^{\text{on}}\right)}{1 - F_{\text{on}}\left(t - t_{\text{in\_}i}^{\text{on}}\right)} \\ &= \frac{1 - \int_{t_{\text{min}}}^{t' - t_{\text{in\_}i}^{\text{on}}} c{t_{\text{on}}}^{-\lambda_{\text{on}}} dt_{\text{on}}}{1 - \int_{t_{\text{min}}}^{t - t_{\text{in\_}i}^{\text{on}}} c{t_{\text{on}}}^{-\lambda_{\text{on}}} dt_{\text{on}}} \\ &= \left(\frac{t' - t_{\text{in}_i}^{\text{on}}}{t - t_{\text{in}_i}^{\text{on}}}\right)^{1-\lambda_{\text{on}}}. \end{aligned} \tag{23}$$

Equation (13) can be further derived to obtain

$$\begin{aligned} &p_{\text{in\_}j}^{\text{off}}\left(t_{\text{in\_}j}^{\text{off}} \le t'\right)_{\text{pl}} \\ &= \frac{F_{\text{off}}\left(t' - t_{\text{out\_}j}^{\text{off}}\right) - F_{\text{off}}\left(t - t_{\text{out\_}j}^{\text{off}}\right)}{1 - F_{\text{off}}\left(t - t_{\text{out\_}j}^{\text{off}}\right)} \\ &= \frac{\int_{t - t_{\text{out\_}j}^{\text{off}}}^{t' - t_{\text{out\_}j}^{\text{off}}} c{t_{\text{off}}}^{-\lambda_{\text{off}}} dt_{\text{off}}}{1 - \int_{t_{\text{min}}}^{t - t_{\text{out\_}j}^{\text{off}}} c{t_{\text{off}}}^{-\lambda_{\text{off}}} dt_{\text{off}}} \\ &= \frac{{t_{\text{min}}}^{\lambda_{\text{off}}-1}\left(\left(t - t_{\text{out\_}j}^{\text{off}}\right)^{1-\lambda_{\text{off}}} - \left(t' - t_{\text{out\_}j}^{\text{off}}\right)^{1-\lambda_{\text{off}}}\right)}{1 - {t_{\text{min}}}^{\lambda_{\text{off}}-1}\left({t_{\text{min}}}^{1-\lambda_{\text{off}}} - \left(t - t_{\text{out\_}j}^{\text{off}}\right)^{1-\lambda_{\text{off}}}\right)} \\ &= 1 - \left(\frac{t' - t_{\text{out}_j}^{\text{off}}}{t - t_{\text{out}_j}^{\text{off}}}\right)^{1-\lambda_{\text{off}}}. \end{aligned} \tag{24}$$

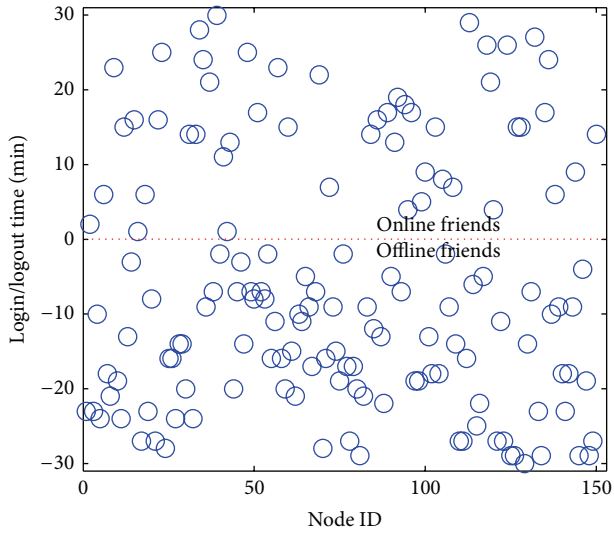


Figure 2: The states of all friends at current time point.

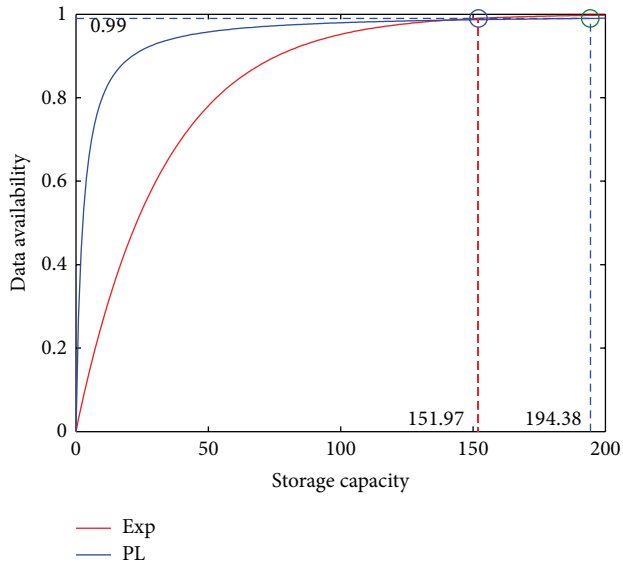


Figure 3: The impact of SS on DA.

Equation (15) can be further derived to

$$\begin{aligned} N_{\text{on}}(t')_{\text{pl}} &= \sum_{i=1}^{N_{\text{on}}} p_{\text{out\_}i}^{\text{on}}\left(t_{\text{out\_}i}^{\text{on}} > t'\right)_{\text{pl}} \\ &\quad + \sum_{j=1}^{N_{\text{off}}} p_{\text{in\_}j}^{\text{off}}\left(t_{\text{in\_}j}^{\text{off}} \le t'\right)_{\text{pl}} \\ &= \sum_{i=1}^{N_{\text{on}}}\left(\frac{t' - t_{\text{in\_}i}^{\text{on}}}{t - t_{\text{in\_}i}^{\text{on}}}\right)^{1-\lambda_{\text{on}}} \\ &\quad + \sum_{j=1}^{N_{\text{off}}}\left(1 - \left(\frac{t' - t_{\text{out}_j}^{\text{off}}}{t - t_{\text{out}_j}^{\text{off}}}\right)^{1-\lambda_{\text{off}}}\right). \end{aligned} \tag{25}$$

6.3. Exponential Distribution. If a random variable t follows the exponential distribution with parameter λ, then its probability density function and probability distribution function can be expressed as in

$$\begin{aligned} f(t) &= \lambda e^{-\lambda t}, \\ F(t) &= 1 - e^{-\lambda t}. \end{aligned} \tag{26}$$

We now show how to use the exponential distribution to derive the on-the-fly prediction for the number of online friends.

With the exponential distribution, (11) can be derived to obtain

$$\begin{aligned} &p_{\text{out}_i}^{\text{on}}\left(t_{\text{out}_i}^{\text{on}} > t'\right)_{\text{exp}} \\ &\quad = \frac{1 - F_{\text{on}}\left(t' - t_{\text{in}_i}^{\text{on}}\right)}{1 - F_{\text{on}}\left(t - t_{\text{in}_i}^{\text{on}}\right)} \\ &\quad = \frac{1 - \left(1 - e^{-\lambda_{\text{on}}(t' - t_{\text{in}_i}^{\text{on}})}\right)}{1 - \left(1 - e^{-\lambda_{\text{on}}(t - t_{\text{in}_i}^{\text{on}})}\right)} \\ &\quad = e^{-\lambda_{\text{on}}(t' - t)}. \end{aligned} \tag{27}$$

Also, (13) can be transformed to

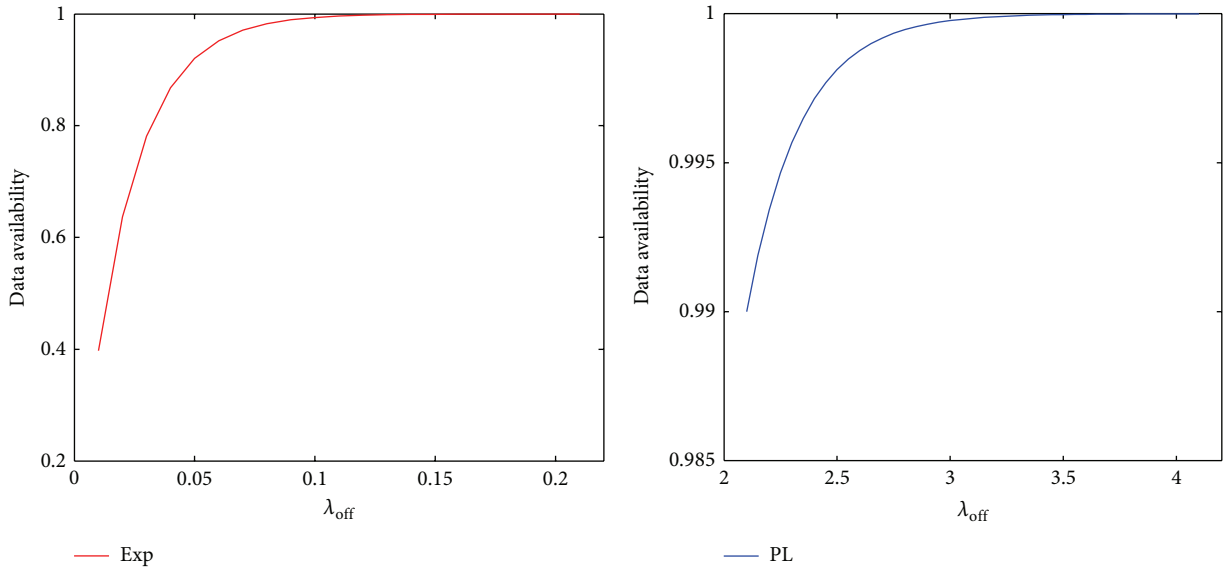
$$\begin{aligned} &p_{\text{in\_}j}^{\text{off}}\left(t_{\text{in\_}j}^{\text{off}} \le t'\right)_{\text{exp}} \\ &\quad = \frac{F_{\text{off}}\left(t' - t_{\text{out\_}j}^{\text{off}}\right) - F_{\text{off}}\left(t - t_{\text{out\_}j}^{\text{off}}\right)}{1 - F_{\text{off}}\left(t - t_{\text{out\_}j}^{\text{off}}\right)} \\ &\quad = \frac{e^{-\lambda_{\text{off}}(t - t_{\text{out}_j}^{\text{off}})} - e^{-\lambda_{\text{off}}(t' - t_{\text{out}_j}^{\text{off}})}}{e^{-\lambda_{\text{off}}(t - t_{\text{out}_j}^{\text{off}})}} \\ &\quad = 1 - e^{-\lambda_{\text{off}}\cdot(t' - t)}. \end{aligned} \tag{28}$$

Further, (15) then becomes

$$\begin{aligned} N_{\text{on}}(t')_{\text{exp}} &= \sum_{i=1}^{N_{\text{on}}} p_{\text{out\_}i}^{\text{on}}\left(t_{\text{out\_}i}^{\text{on}} > t'\right) \\ &\quad + \sum_{j=1}^{N_{\text{off}}} p_{\text{in\_}j}^{\text{off}}\left(t_{\text{in\_}j}^{\text{off}} \le t'\right) \\ &= N_{\text{on}} \cdot \left(e^{-\lambda_{\text{on}}\cdot(t' - t)}\right) + N_{\text{off}} \cdot \left(1 - e^{-\lambda_{\text{off}}\cdot(t' - t)}\right). \end{aligned} \tag{29}$$

7. Evaluation

A discrete simulator has been developed in this work to simulate an OSN. There are N users in the simulated OSN. Some users act as the friends of another user and update the data published by the user. The online and offline durations of the users in the simulated OSN follow the power-law



(a) Exponential distribution (b) Power-law distribution

Figure 4: The impact of the offline durations on DA.

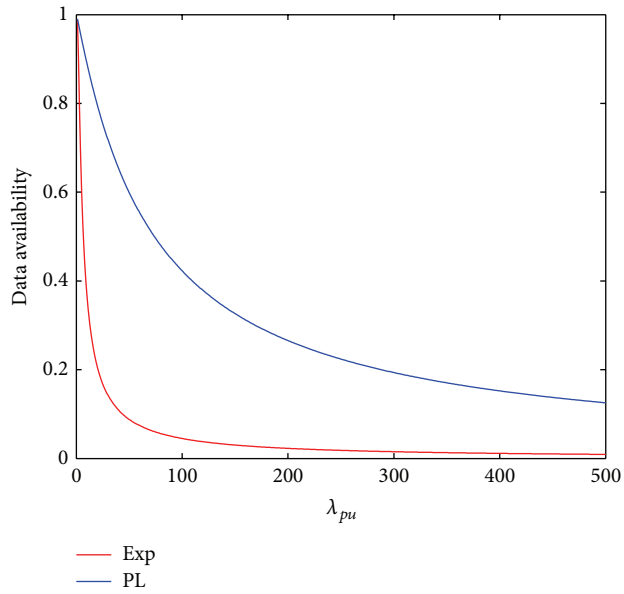


Figure 5: The impact of the data publishing rate on DA.

distribution (PL) or the exponential distribution (Exp), as observed in the literature [37]. The user publishes the data following the Poisson process and k copies of replicas are created for each data item and stored in the online friends.

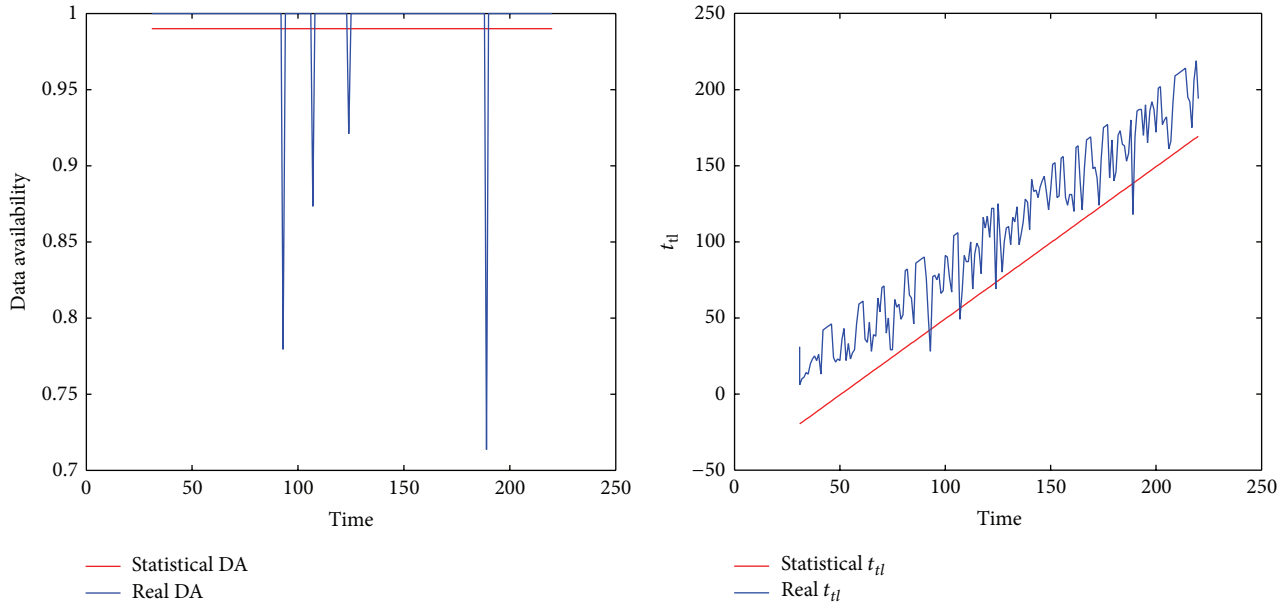
In order to evaluate the DA model over storage capacity, the DA is predicted given the size of storage capacity and the values of other OSN parameters. Then the simulated OSN is run using those parameters values. Each friend contributes the same storage capacity and the storage capacity is allowed to be adjusted so that the total storage capacity of all online friends always equals the storage capacity used to predict the DA. During the running, when a friend comes online at a time point, the DA of the published data for the friend is recorded. The average of all recorded DA is regarded as the actual DA, which is compared against the predicted DA to measure the accuracy of the prediction.

In order to evaluate the on-the-fly prediction, the experimental scenario is designed as follows. A user and his friends log in and out following the specified distribution during the time interval $[0, l]$. The current time is set to be mth min ($m < l$ and the user is offline at time m). The online or offline states of all friends at time m as well as the latest login or logout time before time m are collected. The collected data, combining with the specified distributions, are used to predict the number of online friends and DA at the future time points (i.e., the time points later than m). The predicted data are then compared against the data obtained from the actual running. For example, the number of the friends of a user is set to be 150. Figure 2 shows the online/offline state of each friend when the current time is set to be 31st min. A point above the red line (i.e., when $y = 0$) represents the latest login time of a friend who is online at 31st min, while a point below the red line shows the latest logout time of a friend who is offline at 31st min.

In the rest of this section, the DA model over storage capacity is evaluated in Section 7.1 with regard to the following aspects: (i) the impact of storage capacity on DA, (ii) the impact of the DOSN parameters, including online/offline duration and the rate of user publishing data, on DA, and (iii) the accuracy of the relation established between DA and SS.

In Section 7.2, the on-the-fly prediction is evaluated with regard to the following aspects: (i) the accuracy of predicting the number of online friends on the fly and (ii) the accuracy of the DA predicted on the fly.

Unless stated otherwise, the experimental parameters used in the performance evaluations take the values shown



(a) From the perspective of DA

(b) From the perspective of t_{tl}

Figure 6: The accuracy of the DA model.

Table 2: The default parameters in performance evaluations.

Notations	Default value	Descriptions
N	150	The number of the user's friends
a	1	The average size of published data
$\lambda_{\text{on}}^{\text{exp}}$	1/3	The parameter of the online time duration which follows exponential distribution
$\lambda_{\text{off}}^{\text{exp}}$	1/11	The parameter of the offline time duration which follows exponential distribution
$\lambda_{\text{on}}^{\text{pl}}$	2.5	The parameter of the online time duration which follows power-law distribution
$\lambda_{\text{off}}^{\text{pl}}$	2.1	The parameter of the offline time duration which follows power-law distribution
$\lambda_{\text{pu}}^{\text{ps}}$	1	The parameter of the number of times the user publishes data which follows Poisson distribution

in Table 2. These values are chosen based on those used in the literature [19, 35, 37].

7.1. Evaluating the DA Model over Storage Capacity

7.1.1. Impact of Storage Capacity on DA. Figure 3 shows the impact of the total storage capacity (i.e., SS in Section 4) on the DA calculated from the DA model presented in Section 4. As shown in Figure 3, the DA increases as SS increases. Under both exponential distribution and power-law distribution of the friends' online duration, data availability tails off after SS increases more than a certain value. These results suggest that it is unnecessary to ask the friends to contribute unlimited storage capacity, as often assumed in the work in the literature [14, 18].

From this figure, we can also determine SS that is required to achieve a certain DA. For example, DA reaches 99% under PL or Exp when SS is 194.38 and 151.97, respectively.

7.1.2. Impact of On/Offline Durations on DA. As can be seen from the derivation of the DA model presented in Section 4, the online/offline durations have impact on DA. We conducted the experiments to evaluate their impact. Since the online and offline durations have the similar impact, only the results for offline durations are presented in this subsection. Given the distribution of the offline duration, the average duration is controlled by λ_{off}. The inverse of λ_{off} is the length of the duration.

Figure 4 shows the impact of λ_{off} on DA. In the experiments in Figure 4, SS is set to be 194.38 and 151.97 under PL and Exp (as shown in Figure 3), respectively, so that DA is 99% under the default value of λ_{off} (as in Table 2). We then change the value of λ_{off} and plot the corresponding DA. It can be observed that DA increases as λ_{off} increases under both Exp and PL. These results can be explained as follows. When λ_{off} increases, the average length of the friends' offline durations decreases. Given the certain SS, the period of the stored data (i.e., $[t_{tl}, t]$) is fixed. Therefore, the shorter offline durations of the friends result in higher probability that the times of the data that the friends try to update fall into $[t_{tl}, t]$. Consequently, DA is higher.

7.1.3. Impact of the Data Publishing Rate on DA. From the DA model, we can also know that the pattern with which the user publishes data has the impact on DA. It is shown in the literature that the number of times that the user publishes the data in a duration follows the Poisson distribution. Then, the parameter of the Poisson distribution, λ_{pu}, reflects the data

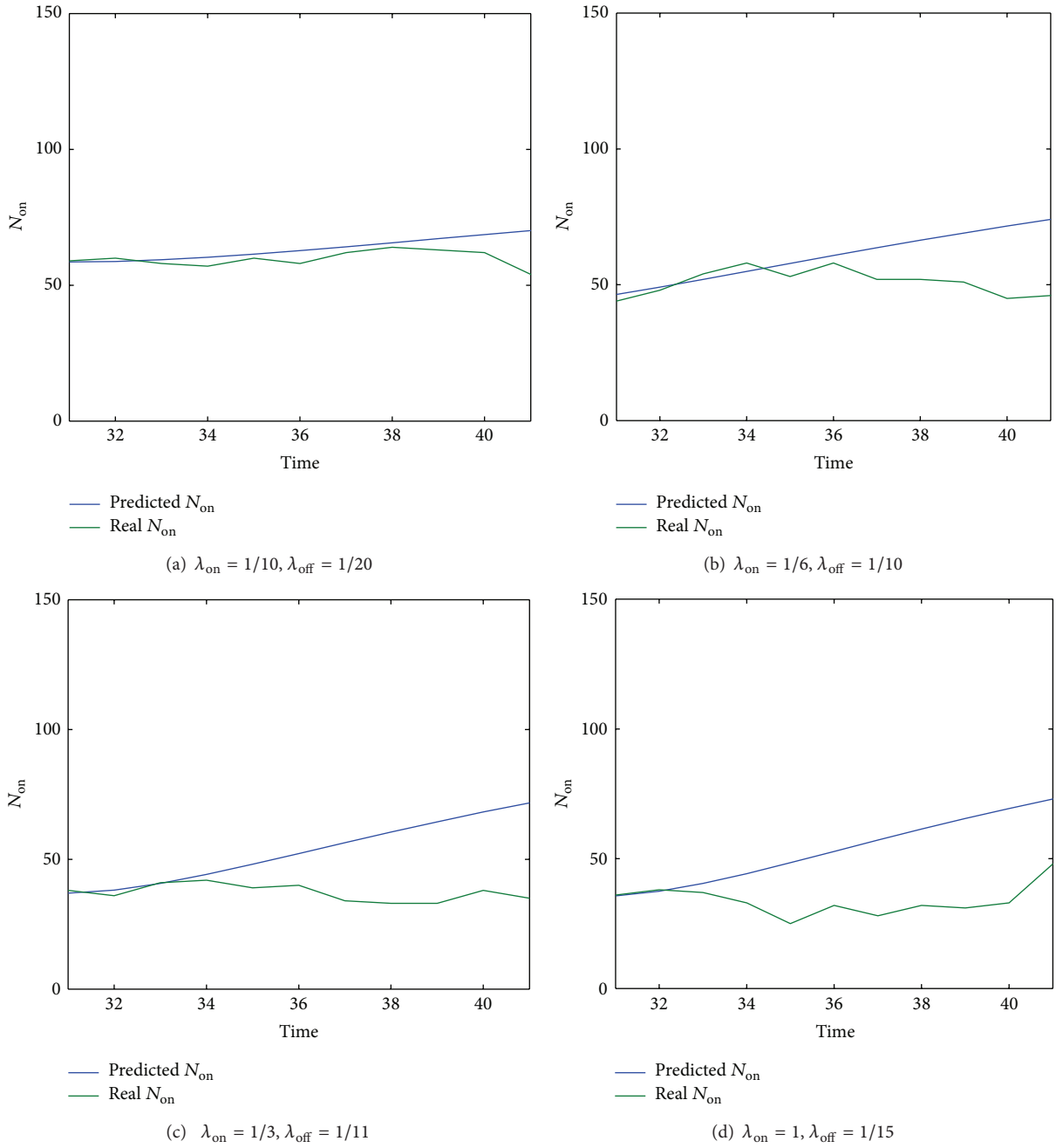


FIGURE 7: The accuracy of prediction model over time.

publishing rate. The higher the λ_{pu}, the higher the data publishing rate.

Figure 5 demonstrates the impact of λ_{pu} on DA. The setting of SS is the same as that in Figure 4. The figure shows that DA decreases as λ_{pu} increases. This is because when the data are published at a higher rate, $[t_{tl}, t]$ is shorter given a fixed SS. Consequently, DA is lower.

7.1.4. Accuracy of the DA Model. The DA model over storage capacity proposed in Section 4 can calculate the DA given an SS. We conducted the experiments to study how accurate the calculated DA is, compared with the DA obtained from the actual running. The results are presented in Figure 6. The results under Exp and PL show similar pattern. Therefore, only the results under Exp are presented.

In Figure 6, the setting of SS is the same as that in Figure 4 (i.e., SS = 151.97). The DA calculated by the DA model is 99%, which is the red line in Figure 6(a). We run the simulated OSN with this SS and plot the actual DA over time, which is the blue line in Figure 6(a). It can be seen that the DA is fairly close to the calculated DA in most cases. These results suggest that the DA model is effective. In order to reveal the fundamental reason for this, we also compared t_{tl} obtained

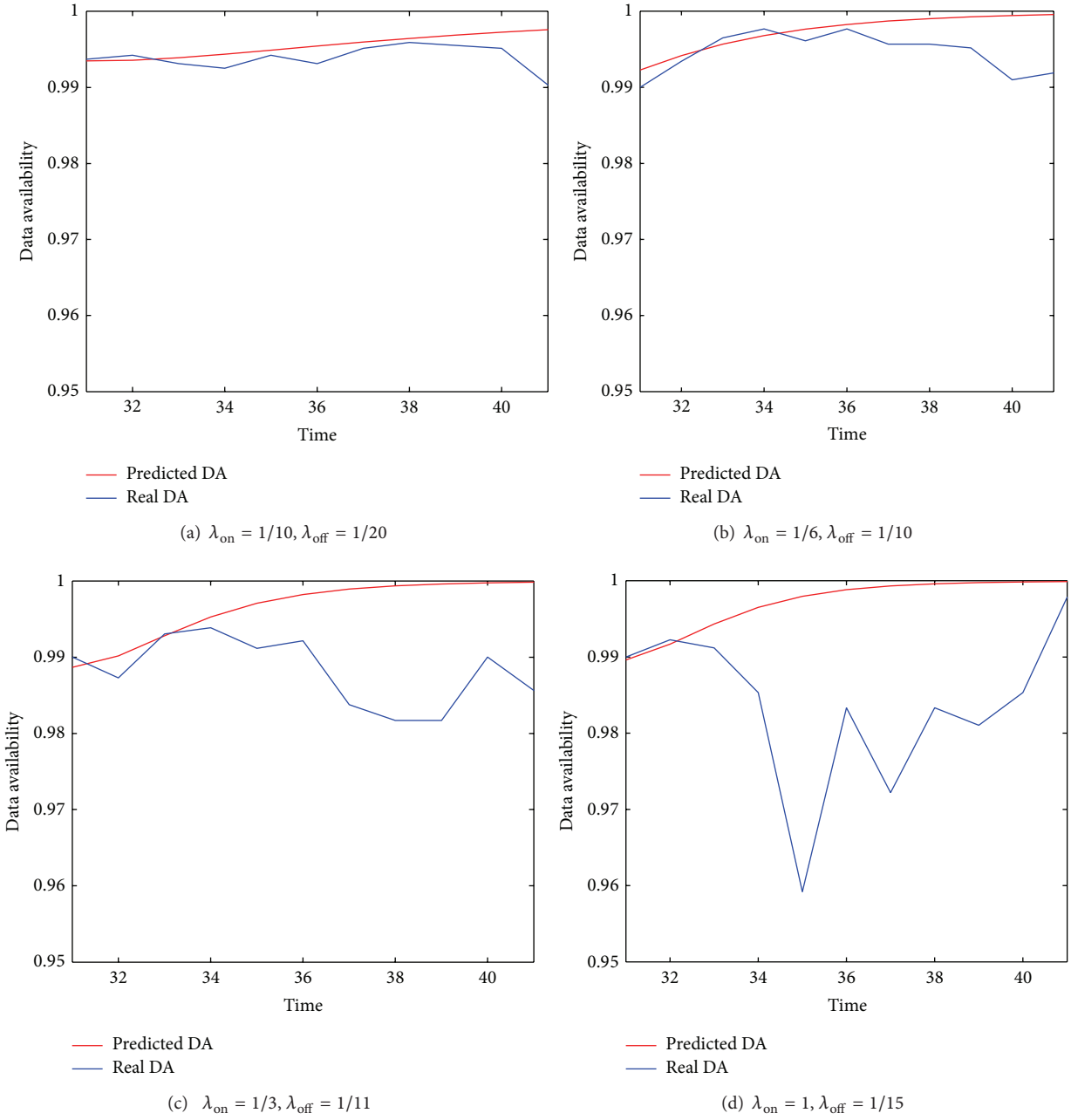


Figure 8: The accuracy of the on-the-fly prediction of DA.

in the DA model (the red line in Figure 6(b)) with the time of the oldest data that a friend tried to update when he came online at a time point (plotted in blue in Figure 6(b)). If the time of the oldest data is not earlier than the calculated t_{tl}, the DA model is effective. As can be seen from Figure 6(b), the blue line is higher (i.e., the corresponding time is later) than the red line in most cases. This gives the fundamental reason why the DA model is effective; that is, with the SS obtained by the DA model, the online friends can in most cases store the data that a friend tries to update when he comes online.

7.2. Evaluating the on-the-Fly Prediction of DA

7.2.1. Accuracy of the Predicted Number of Online Friends and the Impact of Online and Offline Durations. As shown in Section 5, the predicted number of online friends (i.e., N_{on}) determines the value of the on-the-fly DA. Therefore, we conducted the experiments to evaluate the accuracy of predicting N_{on}. The experimental scenario has been presented in the third paragraph of Section 7. The experimental results are shown in Figure 7. In Figure 7, the current time point is set to be 31st min and the on-the-fly prediction predicts N_{on} from

31st min onwards, which is plotted in blue. The actual N_{on} from 31st min onwards is plotted in green. Figures 7(a), 7(b), 7(c), and 7(d) show the results under different λ_{on} and λ_{off} (i.e., online and offline durations).

It can be seen from Figure 7(a) that, compared with its actual values, the prediction of N_{on} is fairly accurate in the first 10 minutes, which shows the effectiveness and applicability of the proposed prediction method since the prediction can be conducted on the fly as the time elapses. By comparing Figures 7(a), 7(b), 7(c), and 7(d), we can see that the length of the accurate prediction decreases as the settings of λ_{on} and λ_{off} change from Figures 7(a)–7(d). These results indicate that the online and offline durations have impact on the prediction accuracy. After carefully analyzing the changing trend of λ_{on} and λ_{off}, it appears that the minimum value between the online and the offline durations (i.e., $\min(1/\lambda_{on}, 1/\lambda_{off})$) determines the length of accurate prediction. The less the value of $\min(1/\lambda_{on}, 1/\lambda_{off})$, the shorter the length of the accurate prediction. The reason for this is because when $\min(1/\lambda_{on}, 1/\lambda_{off})$ is smaller, the friends are more dynamic and, consequently, it is more difficult to obtain the accurate prediction in the future.

7.2.2. Accuracy of the Predicted DA. Finally, Figure 8 presents the experiments results that show the accuracy of the on-the-fly prediction of DA. The experimental settings in Figure 8 are the same as those in Figure 7. It can be seen from Figure 8 that the trends shown in Figure 8 are consistent with those in Figure 7. This once again shows the effectiveness of the on-the-fly prediction.

8. Conclusions

This paper proposes a data availability model over storage capacity for DOSNs. Further, a novel method is proposed to predict the data availability on the fly. Extensive simulation experiments have been conducted. The results show that the proposed data availability method is able to capture the relation between data availability and storage capacity effectively, and that the on-the-fly prediction method can predict the level of data availability accurately.

This work is situated at the level of maintaining the data availability. How to optimize the data accessing performance and reduce the data maintenance overhead is the work of the underlying data replication and placement strategies. In the future, we plan to work down the management level in DOSN and develop the strategies of placing data replicas among friends in DOSN. When designing the placement strategies, the attributes of individual friends, such as the bandwidth and latency associated with a friend and the storage capacity contributed by a friend, will be taken into account.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The authors would like to thank the users and the developer community for their help with this work. The work reported in this paper was supported by China HGJ Project (no. 2013ZX01040-002) and China Open Fund Project (no. KJ-13-105).

References

[1] "Facebook," https://www.facebook.com/.

[2] Sina Microblog, http://weibo.com/.

[3] B. Krishnamurthy and C. E. Wills, "Characterizing privacy in online social networks," in *Proceedings of the 1st Workshop on Online Social Networks (WOSN '08)*, pp. 37–42, ACM, August 2008.

[4] R. Narendula, T. G. Papaioannou, and K. Aberer, "Privacy-aware and highly-available OSN profiles," in *Proceedings of the 19th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE '10)*, pp. 211–216, IEEE, June 2010.

[5] B. Zhou, J. Pei, and W. S. Luk, "A brief survey on anonymization techniques for privacy preserving publishing of social network data.," *ACM SIGKDD Explorations Newsletter*, vol. 10, no. 2, pp. 12–22, 2008.

[6] R. Rabade, N. Mishra, and S. Sharma, "Survey of influential user identification techniques in online social networks," in *Recent Advances in Intelligent Informatics*, pp. 359–370, Springer, 2014.

[7] S. Buchegger, D. Schiöberg, L.-H. Vu, and A. Datta, "Peerson: P2P social networking—early experiences and insights," in *Proceedings of the 2nd ACM EuroSys Workshop on Social Network Systems (SNS '09)*, pp. 46–52, ACM, March 2009.

[8] C. A. Yeung, I. Liccardi, K. Lu et al., "Decentralization: the future of online social networking," in *Proceedings of the W3C Workshop on the Future of Social Networking Position Papers*, 2009.

[9] U. Tandukar and J. Vassileva, "Selective propagation of social data in decentralized online social network," in *Advances in User Modeling*, pp. 213–224, Springer, Berlin, Germany, 2012.

[10] J. Li and F. Dabek, "F2F: reliable storage in open networks," in *Proceedings of the the 5th International Workshop on Peer-to-Peer Systems (IPTPS '06)*, 2006.

[11] R. Sharma, A. Datta, M. Deh'Amico, and P. Michiardi, "An empirical study of availability in friend-to-friend storage systems," in *Proceedings of the 11th IEEE International Conference on Peer-to-Peer Computing (P2P '11)*, pp. 348–351, September 2011.

[12] Diaspora, https://joindiaspora.com/.

[13] R. E. Wilson, S. D. Gosling, and L. T. Graham, "A review of facebook research in the social sciences," *Perspectives on Psychological Science*, vol. 7, no. 3, pp. 203–220, 2012.

[14] A. Olteanu and G. Pierre, "Towards robust and scalable peer-to-peer social networks," in *Proceedings of the 5th Workshop on Social Network Systems (WOSN '12)*, ACM, 2012.

[15] Z. Bu, Z. Xia, J. Wang et al., "A last updating evolution model for online social networks," *Physica A: Statistical Mechanics and Its Applications*, vol. 392, no. 9, pp. 2240–2247, 2013.

[16] S. Buchegger and A. Datta, "A case for P2P infrastructure for social networks—opportunities and challenges," in *Proceedings of the 6th International Conference on Wireless On-demand Network Systems and Services (WONS '09)*, pp. 161–168, February 2009.

[17] N. Li and G. Chen, "Analysis of a location-based social network," in *Proceedings of the Computational Science and Engineering (CSE '09)*, pp. 263–270, August 2009.

[18] D. Koll, J. Li, and X. Fu, "With a Little help from my friends: replica placement in decentralized online social networks," Tech. Rep. IFI-TB-2013-01, Institute of Computer Science, University of Goettingen, Göttingen, Germany, 2013.

[19] J. Ugander, B. Karrer, L. Backstrom, and C. Marlow, "The anatomy of the facebook social graph," http://arxiv.org/abs/1111.4503.

[20] M. McGlohon, L. Akoglu, and C. Faloutsos, "Statistical properties of social networks," in *Social Network Data Analytics*, pp. 17–42, Springer, New York, NY, USA, 2011.

[21] Y.-Y. Ahn, S. Han, H. Kwak, S. Moon, and H. Jeong, "Analysis of topological characteristics of huge online social networking services," in *Proceedings of the 16th International World Wide Web Conference (WWW '07)*, pp. 835–844, ACM, May 2007.

[22] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee, "Measurement and analysis of online social networks," in *Proceedings of the 7th ACM SIGCOMM Internet Measurement Conference (IMC '07)*, pp. 29–42, ACM, October 2007.

[23] L. Jin, Y. Chen, T. Wang et al., "Understanding user behavior in online social networks: a survey," *IEEE Communications Magazine*, vol. 51, no. 9, pp. 144–150, 2013.

[24] F. Benevenuto, T. Rodrigues, M. Cha et al., "Characterizing user behavior in online social networks.," in *Proceedings of the 9th ACM SIGCOMM Conference on Internet Measurement Conference*, pp. 49–62, ACM, 2009.

[25] F. Schneider, A. Feldmann, B. Krishnamurthy et al., "Understanding online social network usage from a network perspective," in *Proceedings of the 9th ACM SIGCOMM Conference on Internet Measurement Conference*, pp. 35–48, ACM, 2009.

[26] O. Kwon and Y. Wen, "An empirical study of the factors affecting social network service use," *Computers in Human Behavior*, vol. 26, no. 2, pp. 254–263, 2010.

[27] Q. Yan, L. Wu, C. Liu, and X. Li, "Information propagation in online social network based on human dynamics," *Abstract and Applied Analysis*, vol. 2013, Article ID 953406, 6 pages, 2013.

[28] A. Shakimov, A. Varshavsky, L. P. Cox et al., "Privacy, cost, and availability tradeoffs in decentralized OSNs," in *Proceedings of the 2nd ACM Workshop on Online Social Networks*, pp. 13–18, ACM, 2009.

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

[30] F. Tegeler, D. Koll, and X. Fu, "Gemstone: empowering decentralized social networking with high data availability," in *Proceedings of the 54th Annual IEEE Global Telecommunications Conference (GLOBECOM '11)*, pp. 1–6, IEEE, December 2011.

[31] T. Amjad, M. Sher, and A. Daud, "A survey of dynamic replication strategies for improving data availability in data grids," *Future Generation Computer Systems*, vol. 28, no. 2, pp. 337–349, 2012.

[32] D. Kossmann, T. Kraska, S. Loesing et al., "Cloudy: a modular cloud storage system," *Proceedings of the VLDB Endowment*, vol. 3, no. 1-2, pp. 1533–1536, 2010.

[33] W. Zeng, Y. Zhao, K. Ou, and W. Song, "Research on cloud storage architecture and key technologies," in *Proceedings of the 2nd International Conference on Interaction Sciences: Information Technology, Culture and Human (ICIS '09)*, pp. 1044–1048, ACM, November 2009.

[34] K. Rzadca, A. Datta, and S. Buchegger, "Replica placement in p2p storage: Complexity and game theoretic analyses," in *Proceedings of the 30th IEEE International Conference on Distributed Computing Systems (ICDCS '10)*, pp. 599–609, IEEE, June 2010.

[35] T. Zhou, X. P. Han, and B. H. Wang, "Towards the understanding of human dynamics," in *Science Matters: Humanities as Complex Systems*, pp. 207–233, 2008.

[36] F. T. O'Donovan, C. Fournelle, S. Gaffigan et al., "Characterizing user behavior and information propagation on a social multimedia network," in *Proceedings of the International IEEE Workshop on Social Multimedia Research (SMMR '13)*, San Jose, Calif, USA, July 2013.

[37] A.-L. Barabási, "The origin of bursts and heavy tails in human dynamics," *Nature*, vol. 435, no. 7039, pp. 207–211, 2005.

[38] D. Stutzbach and R. Rejaie, "Understanding churn in peer-to-peer networks," in *Proceedings of the 6th ACM SIGCOMM on Internet Measurement Conference (IMC '06)*, pp. 189–202, ACM, October 2006.

[39] Mathematica, http://www.wolfram.com/.

[40] M. E. J. Newman, "Power laws, Pareto distributions and Zipf's law," *Contemporary Physics*, vol. 46, no. 5, pp. 323–351, 2005.