Accelerating Peer-to-Peer File Sharing with Social Relations

Haiyang Wang, *Member, IEEE*, Feng Wang, *Member, IEEE*, Jiangchuan Liu, *Senior Member, IEEE*, Chuang Lin, *Senior Member, IEEE*, Ke Xu, *Senior Member, IEEE*, and Chonggang Wang, *Senior Member, IEEE*

***Abstract*—Peer-to-peer file sharing systems, most notably BitTorrent (BT), have achieved tremendous success among Internet users. Recent studies suggest that long-term relationships among BT peers could be explored for peer cooperation, so as to achieve better sharing efficiency. However, whether such long-term relationships exist remain unknown. From an 80-day trace of $100,000$ real world swarms, we find that less than 5% peers can meet each other again throughout the whole period, which largely invalidates the fundamental assumption of these peer cooperation protocols.**

Yet the recent emergence of online social network applications sheds new light on this problem. In particular, a number of BT swarms are now triggered by Twitter, reflecting a new trend for initializing sharing among communities. In this paper, we for the first time examine the challenges and potentials of accelerating peer-to-peer file sharing with Twitter social networks. We show that the peers in such swarms have stronger temporal locality, thus offering great opportunity for improving their degree of sharing. Based on the Hadamard Transform of peers' online behaviors, we develop a *social index* to quickly locate peers of common patterns. We further demonstrate a practical cooperation protocol that identifies and utilizes the social relations with the index. Our PlanetLab experiments indicate that the incorporation of social relations remarkably accelerates the downloading time. The improvement remains noticeable even in a hybrid system with a small set of socially active peers only.

***Index Terms*—Peer-to-peer, social relationship, measurement.**

I. Introduction

PEER-TO-PEER file sharing systems, in particular, BitTorrent (BT), have achieved tremendous success among Internet users. To ensure that the system grows organically, the existing BT protocol relies on a Tit-for-Tat mechanism to penalize free-riders [1]. This incentive mechanism deals well with certain peers' selfish behaviors, and has indeed become a key factor toward the prevailing popularity of BT. Unfortunately, it also hinders decent peers or peers of close relations from more efficient collaboration. Recent studies suggest that long-term relationships among certain BT peers could be explored to achieve better sharing efficiency [2][3]. However, whether such long-term relationships do exist and how they could be effectively identified remain unknown. We have collected trace-data from more than $100,000$ real world swarms spanning over 80 days. We find that peers' online patterns in conventional BT swarms are highly diverse: less than 5% peers can meet each other again in our entire measurement duration[1]. As such, the peers hardly have a chance to help each other, implying the cooperation protocols that blindly assume the existence of long-term relations may not work well.

The recent emergence of online social network applications, for example, Facebook [4] and Twitter [5], sheds new light into this problem. Such applications have been quickly changing the Internet users' behaviors as well as the content sharing landscape. In particular, we have noticed that a number of BT swarms are now triggered by Twitter, reflecting a new trend for initializing sharing among communities. In our 80-day trace, we found that there are $2,106$ Twitter-triggered swarms among the $100,000$ real world swarms, and its percentage steadily grows in our more recent data (as we finished this paper, it reached above 7%).

In this paper, we for the first time examine the challenges and potentials of accelerating peer-to-peer file sharing with social networks, particularly *Twitter-trigger BT swarms*, whose downloads are initialized/shared in Twitter communities. We show that, in these swarms, the peers' online periods are much better overlapped. In particular, more than 35% peers can meet each other again, thus being able to perform data exchange. A closer look into individual peers suggests that a number of peers indeed exhibit very similar online patterns. This temporal locality partly reflects their common social interests, and offers a great opportunity for improving their degree of sharing.

Given the sheer population of Internet peers, identifying these peer pairs of common patterns is resource-intensive, not to mention the requirement of real-time online computation. We address this challenge through qualifying whether a peer is *socially active* in BT networks. Intuitively, assuming a swarm is a party, a socially active peer is a person who regularly attends many parties to meet his/her friends. Through a Hadamard Transform [6] of peers' online behaviors, we develop *social index*, a simple hint to locate these active peers.

Manuscript received February 29, 2012; revised July 14, 2012, 2012.

H. Wang is with the Department of Computer Science, University of Minnesota Duluth, MN, USA (e-mail: hwang@d.umn.com).

F. Wang is with the Department of Computer and Information Science, the University of Mississippi, MS, USA (e-mail: fwang@cs.olemiss.edu).

J. Liu is with the School of Computing Science, Simon Fraser University, British Columbia, Canada (e-mail: jcliu@cs.sfu.ca).

C. Lin and K. Xu are with the Department of Computing Science, Tsinghua University, Beijing, China (e-mail: chlin@tsinghua.edu.cn, xuke@csnet1.cs.tsinghua.edu.cn).

C. Wang is with the Inter Digital, Inc., USA (e-mail: cgwang@ieee.org).

This paper is a significant extension to an earlier short paper (five pages) in INFOCOM 2012 mini-conference.

Digital Object Identifier 10.1109/JSAC.2013.SUP.0513006

[1]Note that the peers are not necessarily online at the same time even they are downloading identical contents in the same swarms.

Its effectiveness has been validated through our trace data. We further demonstrate a practical cooperation protocol that identifies and utilizes the social relations through the index. Preliminary PlanetLab experiments indicate that the incorporation of social relations remarkably accelerates the downloading time for BT peers. The performance improvement remains noticeable even in a hybrid system with a small set of socially active peers only.

The rest of this paper is organized as follows: In Section 2, we present the related works. Based on the measurement of real world Twitter swarms in Section 3, we examine four key issues on accelerating peer-to-peer file sharing with social relations in Sections 4, 5, 6 and 7, respectively. Some piratical issues are further discussed in Section 8 and the paper is concluded in Section 9.

II. Related Works

There have been numerous studies on the implementation, analysis, and optimization on peer-to-peer file sharing, particularly on BT [7]. To deal with certain peers' selfish behaviors, BT introduces the Tit-for-Tat incentive mechanism, which largely prevents a peer from free riding [8]. The effectiveness of Tit-for-Tat has been evaluated through both theoretical analysis and practical experiments [9] [10]. Fan et al. [11] further proposed strategies for assigning rates to connections, which ensures fairness if universally adopted. Neely et al. [12] explored the utility optimization for dynamic networks with Tit-for-Tat. Unfortunately, recent studies have also identified potential problems in this incentive mechanism, e.g., data pollution [13] and weak robustness [14].

More importantly, it is known that Tit-for-Tat hinders decent peers or peers of close relationship from more efficient collaboration. To address this problem, private torrents that do not solely rely on Tit-for-Tat have been introduced for closed communities [3] [15]. For public torrents, long-term relationships among peers have been explored to improve the content availability [16] and the sharing efficiency [17]. However, whether such long-term relationships do exist and how they could be effectively identified and then properly utilized remains unknown, which will be addressed in this paper. We also extend the earlier works on community-based BT [2] [18] by explicitly incorporating the Twitter communities, a real-world social network that have gained great popularity among BT users.

III. Background and Measurement Scheme

A. BitTorrent and Social Applications

Despite its name, peer-to-peer file sharing is often a solitary pursuit, where the peers swap bits of contents, but each of them remains anonymous to one another. Yet, an increasing number of users as well as the BT itself is trying to make downloading more socialized by incorporating social relationships [19]. Twitter, one of the most popular social applications on the Internet, has therefore attracted significant attention from BT users and developers[2]. A new feature in the latest version of the uTorrent [20] client called "Torrent Tweets" allows users to talk about a given download from the application and see what everyone else is saying on Twitter. These social functionalities, have already started changing the way of Internet torrents sharing, similar to the increasing video sharing on Facebook [4].

We have found that more than $10,0000$ groups on Twitter site are built for torrent sharing. It is well known that Twitter emphasizes the up-to-date sharing of instant information among friends. Once a *tweet*(Twitter user) updates a message/torrent link, his/her followers will be able to see it at the same time through updating notifications to their PCs or smartphones. We believe that this feature will potentially change the sharing behavior of peers and is thus worth investigation.

B. Measurement Scheme

To this end, we first collected over $100,000$ torrents from *www.btmon.com*, one of the most popular BT torrent sites. We further crawled the Twitter pages from a cluster of servers in Simon Fraser university to check whether these torrents are also shared among Twitter communities. We found that about 2% ($2,106$ out of $100,000$) of torrents in our dataset are shared on Twitter by Feb 2010, and this ratio has steadily increased to 7% when we finished this paper . For ease of exposition, we call these swarms *Twitter swarms* and others *Normal swarms*.

To learn the peers' online behaviors in these swarms, we passively monitored the BT traffic from the out-going switch of a local ISP from Oct 2009 to Jan 2010 (for over 80 days)[3]. We generated a tracker list based on the collected torrent files (resulting in 683 active trackers in total). According to this tracker list, we obtained the updating message between these trackers and the peers that are located in this ISP. Based on the torrent information in these messages, we found 334 torrents ($10,120$ peers) belonging to our Twitter swarm dataset and $2,271$ torrents ($33,240$ peers) belonging to our normal swarm dataset. In other words, the peers in this ISP have participated in $2,605$ torrents (out of $100,000$) over the 80-day duration.

Considering that many peers may not belong to the ISP that we have measured, we also actively probe the peer information from PlanetLab nodes [21] to obtain more detailed peer information in the swarms. We ran a modified BT client on over 250 PlanetLab nodes, which actively joined the torrents and recorded the observed peer information, as in [22]. As such, we successfully detected the IP addresses of over 95% peers for most of the swarms[4]. We use this result to infer the common interest among BT peers.

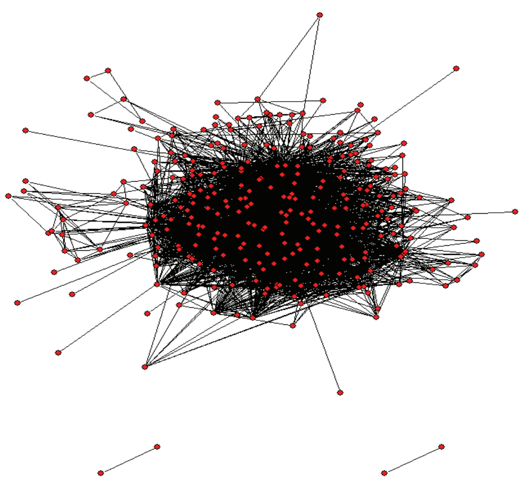
IV. Common Interests Among Peers

We start from examining the peers of commonly interested files. We model the peer relationship across different swarms in a $n \times n$ matrix, Q, where n is the number of peers. Each component of Q, $Q^{i,j}$, is a binary value which indicates

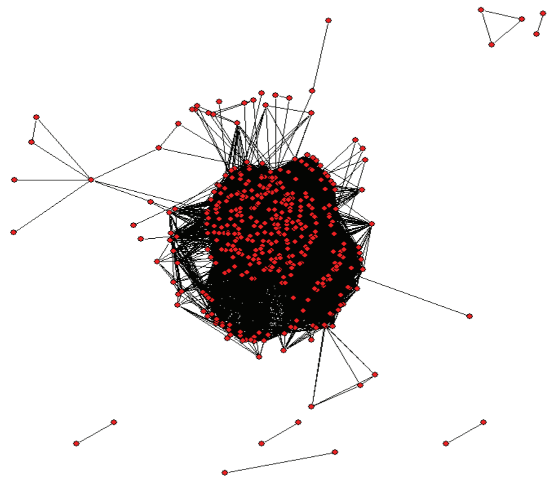
[2]It is worth noting that Twitter itself also highly depends on the BT to manage its thousands of data servers. The BitTorrent-powered system in Twitter's new setup has made the Twitter server deployment 75 times faster than before [5].

[3]This is one of the largest ISPs in China which provides both cable and DSL access service for the users.

[4]This ratio is calculated by comparing the number of detected peers against the total number of peers advertised by the tracker of each torrent.



(a) Peer's common interest in normal swarms



(b) Peer's common interest in Twitter swarms

Fig. 1: Sampled graph visualization of: (a) Q_{normal}; (b) $Q_{twitter}$

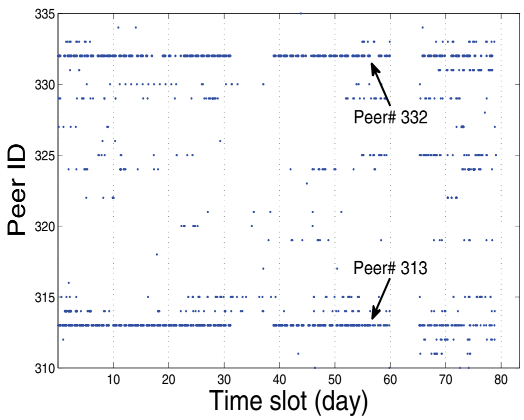


Fig. 2: An illustration to show the existence of highly overlapped peers showing that the peers in Twitter swarms are more tightly related with their common interests.

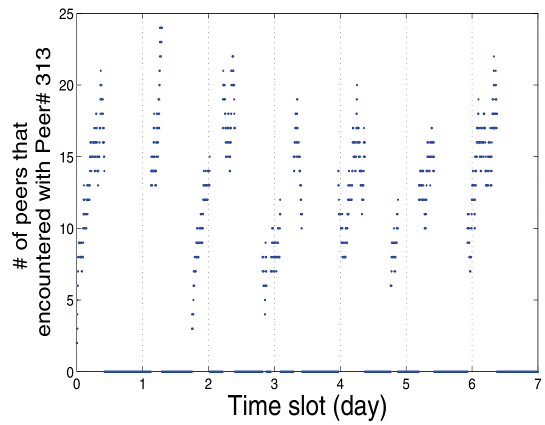


Fig. 3: # of peers that encountered with peer# 313

whether peers i and j share at least one common torrent (1-yes, 0-no). We use Q_{normal} and $Q_{twitter}$ to record the peer relationships in normal swarms and in Twitter swarms, respectively.

A sample graph visualization of Q_{normal} and $Q_{twitter}$ is presented in Figure 1a and Figure 1b (with 400 sampled peers), where the distance between two nodes corresponds to the number of torrents that the peers shared in common; in other words, peers will be closer to each other if they have downloaded more torrents in common. Here we only plot the peers with a degree being greater than 1, i.e., having relationship with others. We can see that $Q_{twitter}$ is denser than Q_{normal} (with 374 and 291 peers respectively). Intuitively, this implies that more peers in Twitter swarms share clear interests with others and have downloaded at least one torrent in common.

A closer look shows that both graphs are not random, but rather having certain community behaviors. This is quantified by evaluating their *clustering coefficient*[5]. The clustering coefficient of $Q_{twitter}$ is over 0.25, whereas the clustering coefficient of Q_{normal} is 0.2. Both of them are noticeably higher as compared to a random graph (nearly 0), and the Twitter swarms exhibit greater communitized behaviors that could be explored.

V. Whether Peers Can Meet Each Other Again?

Unfortunately, simply having common interests is not enough to enable efficient sharing among these peers. A more critical question is whether these peers have similar/overlapping online patterns. Otherwise, the peers will have no chance to help their friends at all. Figure 2 presents an illustration of two peers with similar online patterns. These two peers join the BT networks regularly every day and their online time slots are highly overlapped following a clear 7-day pattern. A closer look of Figure 3 shows that peer#313 is not

[5]The clustering coefficient of node i is the fraction of all possible edges between neighbors of i that are present, while the clustering coefficient of a graph is the average of the coefficient across all nodes [23].

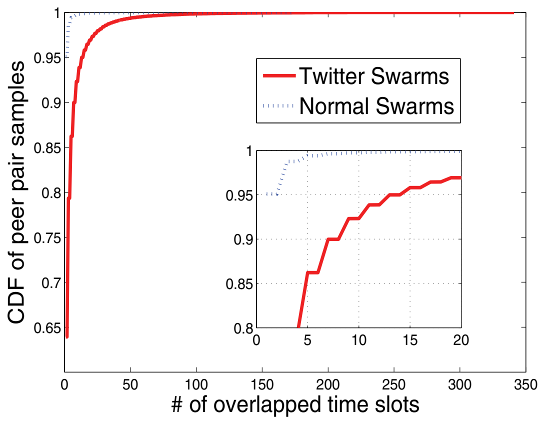


Fig. 4: # of peer encounters

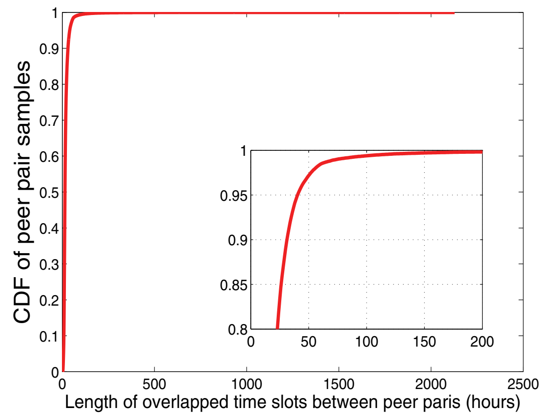


Fig. 5: Length of overlap time slots between peer pair samples (in Twitter swarms)

only overlapped with peer#332 but also very likely to meet other peers in its cluster. However, it is not clear that whether such overlapped patterns are pervasive in BT networks.

To quantitatively evaluate this, we define K as the set of all the trackers, and thus $|K| = 683$. We first collect the online information of the peers from all the trackers. Each tracker k generates a peer availability matrix A_k that indicates the online time slots of the peers: Each component of A_k, $A_k^{i,j}$, is of a binary value, indicating whether peer i is connected to tracker k at time slot j (1-yes, 0-no). In our measurement, the maximum value of i is $43,360$ and the maximum value of j is equal to $120,000$ minutes. We then merge all 683 matrixes together to get a global online matrix G. Each component of G, $G^{i,j}$ is a binary value indicating whether peer i is in the BT networks at time slot j. In our measurement, this matrix refers to the online pattern of $n = 43,360$ peers over $m = 120,000$ minutes. Let $G(i, M)$ denote the ith row of G across time slots M. For example for two peers (n_1 and n_2), we can get their overlapped time slots via the dot product of their availability as:

$$L_{n_1,n_2} = G(n_1, M) \bullet G(n_2, M), \tag{1}$$

Each component of L_{n_1,n_2}, $L_{n_1,n_2}(j)$ is a binary value, indicating whether peer n_1 and n_2 are online at the same time at time slot j. The length (number of online slots) of this overlapped time slots can be described as :

$$K(L_{n_1,n_2}) = \sum_{j=1}^{M} L_{n_1,n_2}(j), \tag{2}$$

where $K(L_{n_1,n_2})$ is an integer indicating the number of the overlapped time slots between peer n_1 and n_2. We also use $K(G(n_1, M))$ and $K(G(n_2, M))$ to denote the total online time of peer n_1 and n_2, respectively.

We first check the number of encounters between the peer pairs, i.e., how many times a peer's online duration is overlapped with another peer's. From Figure 4, we can see that the peers in normal BT swarms are not likely to meet others again; in particular, less then 5% peers can meet others more than once in the BT networks over 80 days. On the other hand, we observe that peers' online patterns are much better overlapped in the Twitter-triggered swarms. In particular, the ratio is increased from 5% to 35%, indicating that more peers are eligible to provide constant helps to others (we call these peers *socially active* peers as discussed in section I). Note that a study from Piatek et al. [24] shows that the peers can hardly have direct data exchange with others again (be assigned as neighbors again). Our study is, however, focusing on peers' online patterns and seeking for the potential to build direct data exchange among social friends.

Given this higher encounter ratio, we further investigate the total length of the overlapped time slots in Twitter swarms. As shown in Figure 5, we can see that most (around 60%) peers overlapped with others for more than 15 hours in our measurement. This is relatively a long time that could be utilized for effectively exchanging data. An intuitive explanation of Figure 4 and Figure 5 is that Twitter emphasizes the up-to-date sharing of instant information among friends. Once a tweet updates a message/torrent, his/her followers will be able to see this message at the same time (through updating notifications) and then start to download. Therefore, the peers are very likely to share common interests and to have very similar online patterns. Since the Twitter communities consist of largely trusted friends, a better sharing incentive can naturally be expected.

VI. How to Identify Socially Active Peers?

Given the existence of overlaps, we now discuss how to identify the socially active peers in this section. We will first examine the unique feature of these peers through trace analysis, and then derive an effective index for identify.

A. Trace Analysis

In this part, our investigation is based on two sets of peers: 1) 200 peers that are highly (meet more than twice) overlapped with others, and 2) 200 peers that are slightly (meet less or equal than twice) overlapped with others. To illustrate their underlying difference, we first analyze their autocorrelation.

Let $\overline{\chi}$ be the complex conjugate of χ. The autocorrelation of $\underline{a}$ for a given shift τ is defined by:

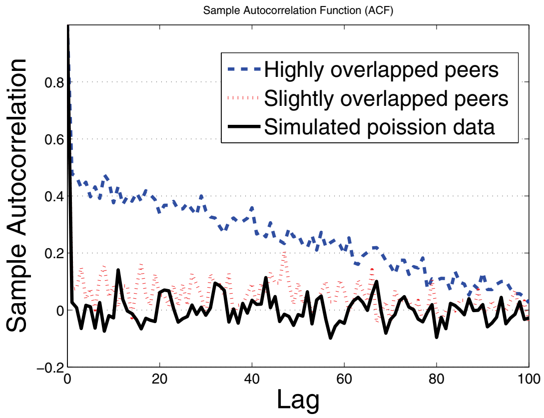


Fig. 6: The autocorrelation function for the data in 6 hours

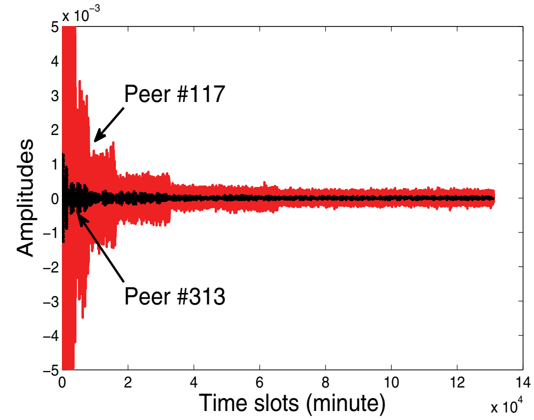


Fig. 7: Amplitude distribution of peer#117 and peer#313

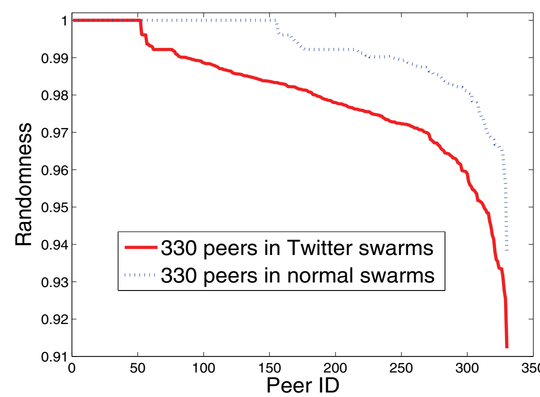


Fig. 8: Randomness of peers' online behaviors

$$C_{\underline{a}}(\tau) = \sum_{i=0}^{t-1} \chi(a_{i+\tau})\overline{\chi(a_i)}, 0 \leq \tau \leq t-1 \tag{3}$$

where τ is a phase shift of the sequence $\{a_i\}$ and the indices are computed modulo t, the period of $\underline{a}$. $\{a_i\} = \underline{a}$ refers to the input sequence (a row in global online matrix G).

Figure 6 shows the autocorrelation coefficient in 6-hour intervals for 200 highly-overlapped peers, 200 slightly-overlapped peers and a simulated Poisson data. It is quite clear that the autocorrelation of highly-overlapped peers decays very slowly, exhibiting a power-law-like curve. On the other hand, the function of slightly-overlapped peers and simulated Poisson data decay very quickly to a close-to-zero level which shows the absence of long-range dependence. When we increase the length of the interval to one day, one week, and one month, respectively, we find that the autocorrelation function of highly-overlapped peers becomes more and more stable around 0.3, and the decay also becomes slower. The autocorrelation function of slightly-overlapped peers and the Poisson data, however, decrease very quickly to near-zero. This observation shows that that the online behaviors of slightly-overlapped peers are relatively random (like Poisson data), which do not have long-range dependence, even we consider a very long time interval, e.g., one month.

Note that the autocorrelation reveals the underlying difference of peers' online behaviors. Yet, itself is not efficient in identifying the overlapped peers, particularly considering its computation overhead.

B. Hadamard Transform based Social Index

From the trace analysis, we can learn that the online behavior of slightly-overlapped peers is very similar with that of the Poisson data. Thus, if we can successfully detect the randomness in this Poisson-like data, other highly-overlapped peers will naturally be identified.

Hadamard Transform [6], also known as Walsh-Hadamard Transform, belongs among a generalized class of Fourier Transforms, and has been wildly used to measure the randomness of binary sequences in signal processing and security fields. The standard Hadamard Transform of $f(x)$ is defined as follows:

$$\widehat{f}(\lambda) = \sum_{x \in T} \chi(\lambda_x)\overline{\chi(f(x))}, \lambda \in T \tag{4}$$

where $f(x)$ is a *trace representation* [25] of the input $\underline{a}$ where $\underline{a}$ is a binary sequence indicating the availability of a peer (1:online; 0:not online). $\overline{\chi(f(x))}$ is the complex conjugate of $\chi(f(x))$. If we define $\mathbb{F} = GF(q)$, the finite field with q elements, and $\mathbb{F}_q^*$, the multiplicative group of $\mathbb{F}_q$. $\mathbb{T}$ will be equal to $\mathbb{F}_q$. The inverse transform is given by:

$$\chi(f(\lambda)) = \frac{1}{q} \sum_{x \in \mathbb{T}} \chi(\lambda_x)\overline{\widehat{f}(x)}, \lambda \in T \tag{5}$$

Let $I(\widehat{f}(\lambda))$ be the number of independent amplitudes of $\widehat{f}(\lambda)$, and L be the length of the sequence. The randomness $r(f(\lambda))$ is given by:

$$r(f(\lambda)) = \frac{I(\widehat{f}(\lambda))}{L}, \lambda \in T \tag{6}$$

Therefore, we define the social index of $f(x)$ as :

$$S(f(\lambda)) = \sum_{\lambda \in T} f(\lambda)(1 - r(f(\lambda))) \tag{7}$$

An illustration of Hadamard Transform is shown in Figure 7, where peer#117 is a peer with regular daily online pattern (for example: being online regularly at each day of the

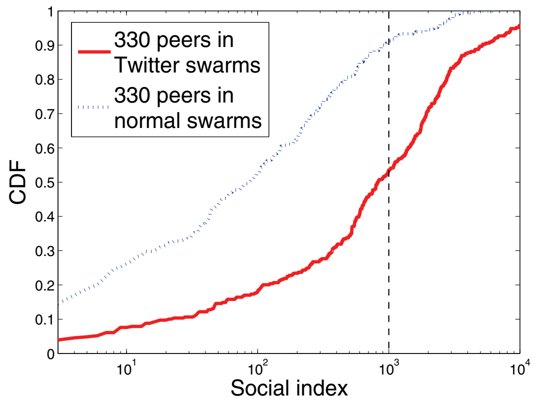


Fig. 9: CDF of social index

week) and peer#313 is not. Figure 7 shows their amplitudes after Hadamard Transform (Eq.4). We can see that the number of independent amplitudes in peer#117 is smaller than that of peer#313. This confirms that peer#117's online behavior is more regular than peer#313.

Eq. 6 quantifies the randomness of the binary sequence $\underline{a}$. This value is between 0 and 1 where $r(f(\lambda)) = 1$ means the peer's behavior is random and no regular pattern can be learnt. In Eq.7, $\sum_{\lambda \in T} f(\lambda)$ refers the total online time of the peer and S is the expectation that the peer will be regularly online to meet other friends. Recall the definition of socially active peers: if we see each swarm as a party and the peer as a person, the social index S is how likely this person will attend multiple parties regularly to meet his/her friends.

To validate whether the social index can well qualify the peer's overlapping behavior, we compute $S(f(\lambda))$ of the swarms in our dataset. As shown in Figure 8 and Figure 9, we randomly select 330 peers in Twitter swarms and 330 peers in normal swarms. As discussed earlier, the peers in Twitter swarms are better overlapped. We find that the peers in Twitter swarms have significantly higher social indices than those of other peers. Based on our trace data, we set a threshold $e = 2,000$ to do the peer identification. We can see that in Figure 9, 35% Twitter peers have social indices greater than $2,000$, which is consistent with our observation in Section 5. It is also worth noting that the computation of the peers' social indices does not require the comparison between peer pairs. The complexity of computing $\widehat{f}(\lambda)$ is $nlogn$ [26], which is efficiently enough to be applied in the real world systems.

VII. Can Social Networks Accelerate Content Sharing?

In this section, we will discuss the performance gain of the peer incorporation based on social relationship. A possible social network based protocol is proposed and evaluated through preliminary Planet-lab experiment based on our trace.

A. Collaboration Among BT Peers: A Simple Solution

We assume that peers' social relationships can be obtained by the trackers (either by the interaction with social applications, such as Twitter, or by our proposed social index), and the trackers will select the majority, but not all, of the peers' social friends to build the peers' neighbor lists (with a maximum of 8 social friends out of 10 neighbors in our design).

The standard choking algorithm is designed by only changing who's choked once every 10 seconds. This is processed by unchoking the 4 peers which it has the best downloading rates. If a leecher has completed the downloading (became a seeder) it will use its uploading rate rather than its downloading rate to decide whom to unchoke (note that the optimistic unchoking is not discussed in here).

It is worth noting that for any leecher who wants to fetch data from other leechers, the key requirement is that this leecher should be interested by others. This design guarantees the instant rewards for every bit that the leechers uploaded (except for optimistic unchoking cases), which is considered robust to peers' possible selfish behaviors. However, it also hinders decent peers or peers of close relations from more efficient cooperations; for example, the friend peers in social networks. Therefore, we make a very simple modification to leechers' choking algorithm. In particular, the leechers will use the uploading rate to choke their social friends (as a seeder in the standard BT protocol). In our design, the leechers will unchoke the 3 peers among its social friends, with which it has the highest uploading rates (in the past 10 seconds), and 1 other peer that has the highest uploading rate.

B. Evaluation

To evaluate the benefit of social network based content delivery, we carry out Planet-lab experiments with a modified version of BT client. In particular, we use Planet-lab nodes to run as peers. Considering the peer arrival/departure, most peers are joining the network at once, i.e. the flash crowd scenario. For each torrent, there is one original seeder that will always stay online (with 400Kbps uplink bandwidth). Our evaluation contains two parts: First, to investigate the possible gain in an extreme case where all the peers are social friends; Second, to furthe rclarify this benefit in hybrid swarms with a small set of social friends only.

In the first experiment, we investigate the sharing efficacy in two BT swarms S_{social} and S_{normal} (both with 350 peers). S_{social} consists of social friends, and S_{normal} consists of normal peers. The standard BT protocol is applied to the clients in S_{normal}, whereas our modified choking algorithm is applied to the clients in S_{social}. The content size is 900MB with the piece size of 1024kB. We used a local server in our campus network to run both the seeder and tracker functions, and the seeder's maximum uploading capacity is set to 10M. There are 350 peers arriving over a very short period of 2 minutes. Note that the peers in S_{normal} will leave the swarm as soon as they finish the downloading. On the other hand, the social peers will continue to contribute their uploading if their friends are still downloading the content[6].

Figure 10 shows the download completion time of swarms S_{social} and S_{normal}. It is easy to see that the social-relation-based enhancement significantly improves the peers' download completion time. In S_{social}, 70% peer will finish their downloading within 800 seconds, and the maximum download

[6]This is an reasonable assumption because peers' online patterns are indeed better overlapped in Twitter swarms.

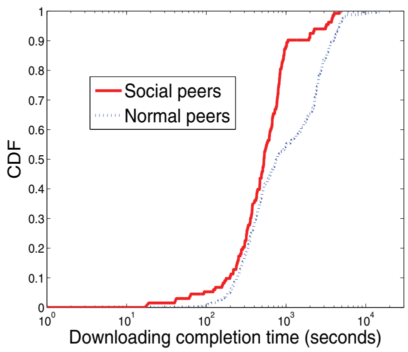


Fig. 10: Completion time

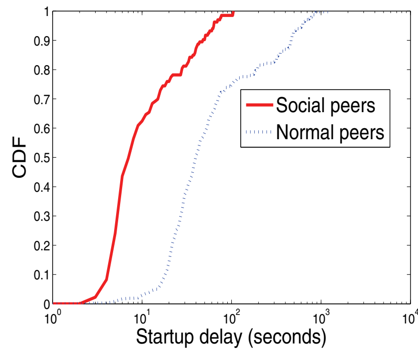


Fig. 11: Startup delay

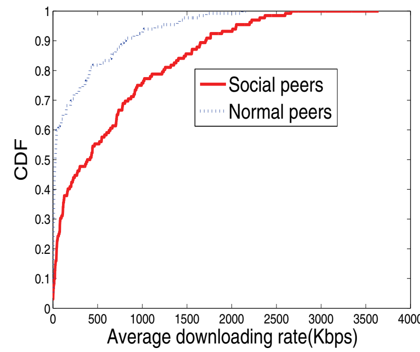


Fig. 12: Downloading rate

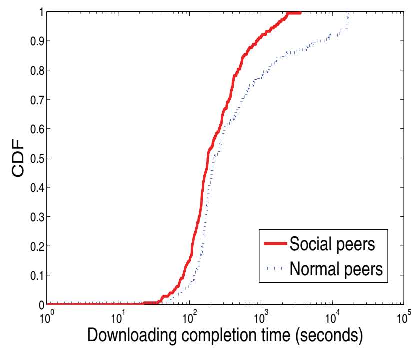


Fig. 13: Downloading completion time (larger swarm)

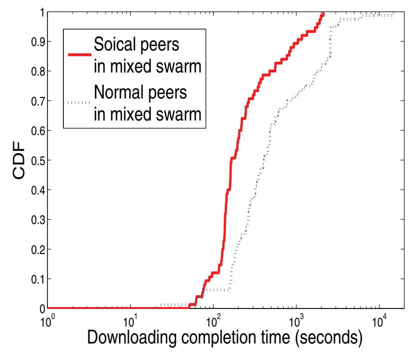


Fig. 14: Downloading completion time (mixed swarm)

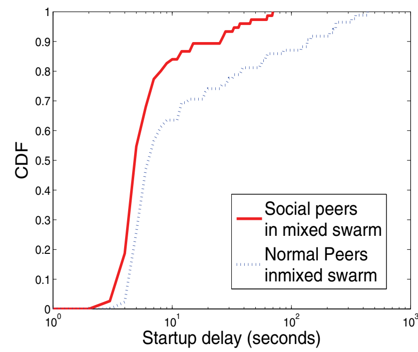


Fig. 15: Startup delay(mixed swarm)

completion time is $4,000$ seconds. In S_{normal}, only 40% peer can finish their download within $4,000$ seconds and the maximum downloading completion time reaches over $20,000$ seconds. Figure 11 further shows that the startup delay (the delay till receiving the first data piece) is also greatly improved. In our new protocol, most peers (90%) in S_{social} receive their first piece within 1 minute. Yet only 60% peers in S_{normal} can achieve this speed with the conventional optimistic uncorking. We believe that it is because the peers' average downloading rate is greatly improved with the social-relation-based enhancement. As shown in Figure 12, 30% peers in S_{social} can achieve a downloading rate of 1M, while less than 10% peer can have such a high rate in S_{normal}.

We have also examined their performance in larger swarms, and a typical result for a 550-peer and 4-seeder systems is shown in Figure 13. Comparing to Figure 10, it is easy to see that the peers in both swarms benefit from the increasing number of seeders. Since the seeders are not selfish and act like "common friends" to all the peers, when we keep increasing the number of seeders in the swarm, the downloading performance of S_{social} and S_{normal} will become closer. However our experiments shows that, the peers' downloading completion time in S_{social} remains much faster than S_{normal}.

C. Performance with a Hybrid System

The above experiment demonstrates the gain with entirely collaborative peers in BT swarms, i.e., all peers are social friends in S_{social}. Before social networks become truly pervasively and seamlessly integrated with BT, however, the real world swarms will still include normal (selfish) peers, which may even dominate. It is thus necessary to see whether the peers can still benefit in such a hybrid swarm with a small set of social friends only.

To this end, we use the trace from a real world Twitter swarm that consists of 350 peers. Using our proposed social index, we find that most peers (280 peers) in this swarm have very low (some of them have even near-zero) social indices whereas the rest of them have clear social friend properties (with social indices larger than $2,000$). The content size is also 900MB with a default piece size of 1024kB. We test the downloading of this swarm on PlanetLab with exactly the same peer configuration. We modified the client of 100 social peers with our proposed uploading choking algorithm and applied the standard BT protocol to other peers. The social peers will use the uploading rate based choking algorithm to communicate with their friends and use standard choking algorithm to communicate with other peers. All the peer arrivals are within a relatively short period of less than 1.5 minutes. The normal peers will leave the swarm as soon as they finished downloading. The social peers however will continue uploading if their friends are still downloading the content. Note that the tracker in this experiment is modified to achieve biased neighbor selection according to peers' social index (with a maximum of 8 social friends out of 10 neighbors in total). The availability information of each peer is also obtained from our real world trace.

In Figure 14, we can see that the social collaboration of a small number of peers can still lead to considerable benefit to peers' downloading. 80% social peers will finish their downloading around 6 minutes whereas less than 50% normal peers can finish the downloading within 6 minutes. The startup delay in Figure 15 also indicates the benefit of using social networks to accelerate BT, where most social peers can receive their first piece within 15s. It is also worth noting that when we compare figure Figure 14, Figure 15 with Figure 10 and Figure 11, we can see that the deployment of social network based enhancement will not harm the downloading performance of normal peers. In fact, all the peers will more or less benefit from this enhancement. This is indeed consistent with earlier discoveries on the benefit of clustering in the BT System [27].

VIII. Further Discussions

The state-of-the-art BT is known to be a well-design system that achieves high efficiency under diverse network and user settings. Our study has provided evidences that, with knowledge about social relations, the peers can be organized in an even better way to achieve even better sharing efficiency. With the help of social friends, a higher content availability can naturally be expected. Our solution could be applied to other applications such as storage systems with highly collaborative users

Our work remains an initial attempt toward accelerating peer-to-peer file sharing with social relationships. There are still many open issues that can be further explored, and we hereby list three in which we are particularly interested.

Complex social relationships: In our investigation, the peers are classified into two categories, namely, either being friends or not. In the real social networks, however, not all the friendships are equal, and a peer may not care about the downloading of all its friends. Therefore, obtaining and applying social relations with different weights are worth further investigation. Other more complex social relations, beyond the simple friendship, may also be examined.

Optimal content delivery algorithm: There have been studies on the optimality of the BT sharing in homogenous networks [2]. Our focused sharing environment is heterogenous with different types of peers. We thus naturally ask this question: What is the optimal sharing in a hybrid swarms with a small (or major) set of social friends? We are also interested in the impact of the seeder's category, i.e., whether the first seeder is in the social network or not.

Free riding among social friends: Free riding is a very important issue in BT networks. In our evaluation, the modified (uploading rate based) choking protocol is applied among social friends; thus, free riders outside of the social communities will not affect the overall performance. However, if free riders reside within the community, smarter detection and prevention are to be developed.

IX. Conclusions

In this paper, we for the first time examined the challenges and potentials of accelerating peer-to-peer file sharing with Twitter social networks. Our trace analysis showed that the BT system has enough potential to apply social network based enhancements. The PlanetLab experiments further indicated that the incorporation of social relations remarkably accelerates the downloading time even in a hybrid system with a small set of socially active peers only. Given the growing trend of spreading torrents through social networks, we believe that there is a great opportunity to improve the data distribution efficiency in peer-to-peer file sharing systems, which is worth further explorations.

References

[1] T. Locher, P. Moor, S. Schmid, and R. Wattenhofer, "Free riding in bittorrent is cheap," in *Proc. ACM HOTNETS, 2006*.

[2] N. Andrade, M. Mowbray, A. Lima, G. Wagner, and M. Ripeanu, "Influences on cooperation in bittorrent communities," in *Proc. ACM P2PECON, 2005*.

[3] P. Dhungel, D. Wu, Z. Liu, , and K. Ross, "BitTorrent darknets," in *Proc. IEEE INFOCOM, 2010*.

[4] Facebook. [Online]. Available: http://www.facebook.com/

[5] Twitter. [Online]. Available: http://twitter.com/

[6] B. J. Fino and V. R. Algazi, "Classification of random binary sequences using Walsh-Fourier analysis," *Proc. IEEE Trans. Electromagn. Compat.*, vol. EMC-13, no. 3, pp. 74-77, 1971.

[7] D. Qiu and R. Srikant, "Modeling and performance analysis of bit torrent-like peer-to-peer networks," in *Proc. ACM SIGCOMM, 2004*.

[8] B. Cohen, "Incentives build robustness in BitTorrent," in *Workshop Economics Peer-to-peer Syst. 2003*.

[9] R. Axelrod, "The Evolution of Cooperation," in *Basic Books, 1985*.

[10] A. Bharambe, C. Herley, and V. Padmanabhan, "Analyzing and improving a BitTorrent network's performance mechanisms," in *Proc. IEEE INFOCOM, 2006*.

[11] B. Fan, D.-M. Chiu, and J. Lui, "BitTorrent-like file sharing protocol design," in *Proc. ICNP, 2006*.

[12] M. J. Neely and L. Golubchik, "Utility optimization for dynamic peer-to-peer networks with tit-for-tat constraints," in *Proc. IEEE INFOCOM, 2011*.

[13] N. Liogkas, R. Nelson, E. Kohler, and L. Zhang, "Exploiting BitTorrent for fun (but not profit)," in *Proc. USENIX IPTPS, 2006*.

[14] M. Piatek, T. Isdal, T. Anderson, A. Krishnamurthy, and A. Venkataramani, "Do incentives build robustness in BitTorrent?" in *Proc. USENIX NSDI, 2007*.

[15] M. Meulpolder, L. D'Acunto, M. Capota, M. W. andJ.A. Pouwelse, D. Epema, and H. Sips, "Public and private bittorrent communities: A measurement study," in *Proc. USENIX IPTPS, 2010*.

[16] L. Guo, S. Chen, Z. Xiao, E. Tan, X. Ding, and X. Zhang, "Measurements, analysis, and modeling of BitTorrent-like systems," in *Proc. ACM/USENIX IMC, 2005*.

[17] M. Piatek, T. Isdal, A. Krishnamurthy, and T. Anderson, "One hop reputations for peer to peer file sharingworkloads," in *Proc. USENIX NSDI, 2008*.

[18] D. Choffnes, J. Duch, D. Malmgren, R. Guierm, F. Bustamante, and L. A. N. Amaral, "Strange bedfellows: Community identification in BitTorrent," in *Proc. USENIX IPTPS, 2010*.

[19] BitTorrent. [Online]. Available: http://www.bittorrent.com/

[20] uTorrent. [Online]. Available: http://www.utorrent.com/

[21] Planetlab. [Online]. Available: http://www.planet-lab.org/

[22] J. Liu, H. Wang, and K. Xu, "Understanding peer distribution in global internet," *IEEE Netw. Mag., 2010*.

[23] D. Watts and S. Strogatz, "Collective dynamics of small-world networks," *Nature*, vol. 393, no. 6684, pp. 409, 1998.

[24] M. Piatek, T. Isdal, A. Krishnamurthy, and T. Anderson, "One hop reputations for peer to peer file sharing workloads," in *Proc. USENIX NSDI, 2008*.

[25] Z. Dai, G. Gong, H.-Y. Song, and D. Ye, "Trace representation and linear complexity of binary eth power residue sequences of period p," *Proc. IEEE Trans. Inf. Theory*, vol. 57, no. 3, pp. 1530-1547, 2011.

[26] B. J. Fino and V. R. Algazi, "Unified matrix treatment of the fast Walsh-Hadamard transform," *Proc. IEEE Trans. Comput.*, vol. C-25, no. 11, pp. 1142-1146, 1976.

[27] A. Legout, N. Liogkas, E. Kohler, and L. Zhang, "Clustering and sharing incentives in BitTorrent systems," in *Proc. ACM SIGMETRICS, 2007*.

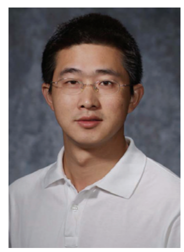
Haiyang Wang received the PhD degree in the Department of Computing Science from Simon Fraser University, Burnaby, British Columbia, Canada in 2013. He is currently an Assistant Professor in the Department of Computer Science, University of Minnesota Duluth. His research interests include cloud computing, bigdata, peer-to-peer networks, multimedia systems/networks and IP routing.

Feng Wang received both the Bachelor's degree and Master's degree in Computer Science and Technology from Tsinghua University, Beijing, China in 2002 and 2005, respectively. He received the PhD degree in Computing Science from Simon Fraser University, Burnaby, British Columbia, Canada in 2012. He is a recipient of the Chinese Government Scholarship for Outstanding Self-financed Students Studying Abroad (2009) and IEEE ICME Quality Reviewer Award (2011). He is currently an Assistant Professor in the Department of Computer and Information Science at the University of Mississippi, MS, USA. His research interests include peer-to-peer networks, wireless sensor networks, cyber-physical systems, socialized content sharing and cloud computing. He is a member of IEEE. He serves as TPC member in IEEE ICME'2011, IEEE ICME'2012 and IEEE GLOBECOM'2013. He serves as TPC member in IEEE ICME2011-2013, IEEE GLOBECOM2013, IEEE CloudCom'2013 and ACM MM'2013.

Jiangchuan Liu is currently an Associate Professor in the School of Computing Science, Simon Fraser University, British Columbia, Canada, and was an Assistant Professor in the Department of Computer Science and Engineering at The Chinese University of Hong Kong from 2003 to 2004. His research interests include multimedia systems and networks, wireless ad hoc and sensor networks, and peer-to-peer and overlay networks. He is a Senior Member of IEEE and a member of Sigma Xi. He is an Associate Editor of IEEE TRANSACTIONS ON MULTIMEDIA, an editor of IEEE COMMUNICATIONS SURVEYS AND TUTORIALS, and an Area Editor of *Computer Communications*. He is TPC Vice Chair for Information Systems of IEEE INFOCOM'2011.

Chuang Lin is a professor of the Department of Computer Science and Technology, Tsinghua University, Beijing, China. He is a Honorary Visiting Professor, University of Bradford, UK. He received the Ph.D. degree in Computer Science from the Tsinghua University in 1994. His current research interests include computer networks, performance evaluation, network security analysis, and Petri net theory and its applications. He has published more than 400 papers in research journals and IEEE conference proceedings in these areas and has published four books. Professor Lin is a senior member of the IEEE. He serves as the Technical Program Vice Chair, the 10th IEEE Workshop on Future Trends of Distributed Computing Systems (FTDCS 2004); the General Chair, ACM SIGCOMM Asia workshop 2005 and the 2010 IEEE International Workshop on Quality of Service (IWQoS 2010); the Associate Editor, IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY; and the Area Editor, *Journal of Computer Networks*.

Ke Xu received the B.S., M.S. and Ph.D. degrees in computer science from Tsinghua University, China in 1996, 1998 and 2001 respectively. Currently he is a Professor in the department of computer science of Tsinghua University. His research interests include next generation Internet, switch and router architecture, P2P and overlay network. He is a Senior Member of IEEE and a member of ACM.



Dr. Chonggang Wang received the Ph.D. degree from Beijing University of Posts and Telecommunications (BUPT), China in 2002. He is currently a Senior Research Staff with InterDigital Communications with focuses on Machine-to-Machine (M2M) communications and Internet of Things (IoT) R&D activities including technology development and standardization. Before joining InterDigital in 2009, he had conducted various researches with NEC Laboratories America, AT&T Labs Research, University of Arkansas, and Hong Kong University of Science and Technology. He (co-)authored more than 100 journal/conference articles and book chapters. He is on the editorial board for several journals including *IEEE Communications Magazine* and IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT. He is and was co-organizing several special issues respectively for IEEE SENSORS JOURNAL, IEEE JSAC, *IEEE Network Magazine*, *IEEE Communications Magazine*, IEEE COMMUNICATIONS SURVEYS AND TUTORIALS, etc. He received Outstanding Leadership Award from IEEE GLOBECOM 2010 and InterDigital 2012 Innovation Award. He is the vice-chair of IEEE ComSoc Multimedia Technical Committee (MMTC).