Inter-Swarm Content Distribution Among Private BitTorrent Networks

Chengchen Hu, *Member, IEEE*, Danfeng Shan, Yu Cheng, *Senior Member, IEEE*, and Tao Qin, *Member, IEEE*

***Abstract*—Private BitTorrent (PT) is a new trend in Peer-to-Peer file sharing system, which provides high incentives for its users to seed after download by maintaining an upload-to-download ratio in the tracker for each registered community member. From the data we collected from six active PT sites, we discover that the population of both users and contents in any single PT site is much less than the public BitTorrent, and the intersection of content sets in different PTs is quite small. Based on this observation, we propose a content sharing/distribution framework among PTs (named CrossPT), as well as its sharing mechanism. In addition, we investigate the sharing strategy of the PT participants in CrossPT using game theory and the fetch strategy by modeling the scenario to a Neighbor Selection Problem (NSP). We prove NSP to be NP-complete and propose a heuristic algorithm to solve it. The evaluations with the input of crawled data from six PT sites demonstrate the efficiency of our mechanism. The content sizes of the six PT sites can be increased by 113.95%-438.46% with CrossPT. Also, the content distribution process can be done in less than one second, excluding the delivery time of the content itself.**

***Index Terms*—Private BitTorrent, P2P networks, content distribution.**

I. Introduction

PEER-to-Peer (P2P) networking is a distributed and self-organizing architecture that processes the tasks among peers. The peers constructing the P2P network nodes play the roles of both resource providers and resource consumers in the network, while a node is either a server or a client in a traditional centralized Client-Server (CS) model [1]. The P2P network is quite successful and its traffic nowadays starts to dominate the Internet. It has been reported in [2] that P2P networks collectively have accounted for roughly 43% to 70% of all Internet traffic depending on geographical location. It is confirmed by the Cisco recently that P2P traffic is still the largest share of Internet traffic today [3].

One of the most important application of P2P system is file sharing. BitTorrent is one of the most commonly employed P2P file sharing protocol for transferring large amount of files over the Internet and the huge amount of popularity gained by the BitTorrent can be substantially attributed to its efficient and scalable resource allocation. When requesting content, the tracker in a BitTorrent files sharing system provides a list of peers containing that content. Originally, a public BitTorrent tracker accepts everyone to connect to it. So, it is easy for people to join such a public BitTorrent community and download their interested contents. However, in such a setting, there is no incentive for peers to stay in the system to seed after the download is completed (as we called this problem "free-riding") from a public BitTorrent community.

Recently, the emerging of Private BitTorrent Communities, also known as the BitTorrent Darknets or Private Trackers (PTs), is a good practice of P2P content distribution system [4]–[6] to increase seeding. A PT community only opens its registration to a limited number of invited users and the tracker maintains an upload-to-download ratio (*i.e.*, sharing ratio) for each registered community member. Only the members, whose sharing ratios are above a predefined threshold, are allowed to download in the community. Such a Sharing Ratio Enforcement (SRE) policy provides incentives to members to keep seeding after downloading.

In general, private PTs well solve the free riding problem in a single community. However, the number of contents in a single PT network is quite small compared to a public BitTorrent community. Even the largest PT network, the number of contents in it (which is 612, 012) is one magnitude less than the number in the public ecosystem (which is 4, 111, 837), as shown in [4]. In the other hand, the overlap of the contents in different PTs is very little. This fact is demonstrated by four very large PTs in [4] and six randomly selected PTs in our study (details will be shown in Section II). As a result, it is not uncommon that a single person struggles to register in several PT communities as so to enlarge the probability to obtain his/her interested files. Since the invitations of a PT community are quite limited, it is not easy to have accounts in a large number of PTs. If only a person registers only in one community and could obtain the contents in other PTs, just like how Internet works today: we connect to the Internet in a domain by an Internet Service Provider (ISP), and besides the sites in the same domain, we can access to others in any different domains through inter-domain routing. In this paper, we advocate a framework to share the contents among PT communities while keeping the flexibility and freedom for each PT network by setting their own strategies.

Although previous investigations on PTs have provided remarkable results through measurement, modeling and simulation, they mainly focus on the intra-swarm efficiency/behavior, and to the best of our knowledge, no previous attention has

Manuscript received February 25, 2012; revised July 14, 2012.

C. Hu, D. Shan and T. Qin are with MOE key lab for Intelligent Networks and Network Security, Department of Computer Science and Technology, Xi'an Jiaotong University, China (e-mail: huc@ieee.org, shandanf@gmail.com, tqin@sei.xjtu.edu.cn)

Y. Cheng is with Department of Electrical and Computer Engineering Technology, Illinois Institute of Technology, USA (e-mail: cheng@iit.edu).

The authors would like to thank Qiuzheng Huang for his help on PT's data collection. This paper is supported by the 863 plan (2013AA013501), the National Science and Technology Major Project under grant (2013ZX03002003-004), the NSFC (61272459, 61221063, 61103240), the Fundamental Research Funds for Central Universities.

Digital Object Identifier 10.1109/JSAC.2013.SUP.0513012

been received on the inter-swarm resource sharing among PTs. The authors of [4] showed that there are over 900 private BitTorrent communities in the Internet and compared public tracker sites and PTs comprehensively from macroscopic, medium-scopic and microscopic perspectives through measurement study. The investigation in [7] has presented extensive measurements of over half a million peers in two public and three private BitTorrent communities. The main findings are 1) the download speeds in private communities are 35 times higher than in public communities, 2) more peers are connectable, 3) the seeder/leecher ratios are at least 10 times as those in public communities, and 4) peers seed for a significantly longer duration. The literature [8] studied 13 PTs and 2 public tracker sites from the metrics of user viscosity, torrents evolution, user behaviors, content distribution, *etc*. The paper traced 17 private trackers, 2 public trackers and 1 BitTorrent search engine for 6 months [6], and it was indicated that private trackers have highly levels of member and torrent engagement. The same authors further introduced Predator-Prey model in ecology to analyze and achieve the stable Seeder-to-Leecher Ratio (SLR) range to solve Poor Downloading Motivation problem [5]. The PT's SRE is studied in-depth in [9], which pointed out that PTs are more incentive and effective for members to seed/upload in terms of a game theoretic model and further proposed an upload entropy scheme to prevent collusions. Zhao et al. combined related behaviors to a single lifecycle and proposed a model on the swarm evolution to show the long term characteristics [10].

We have made three main contributions in this paper.

- To the best of our knowledge, we are the first to advocate content sharing among PTs and have successfully proposed a framework called CrossPT for the task.
- Detailed theoretical analysis is deduced to study the strategies of each PT that plays in CrossPT.
- With the help of measurement study, we have collected data from six active PTs. The evaluations are performed based on the data and demonstrate the efficiency of CrossPT. The content sizes of the 6 PT sites can be increased by 113.95%-438.46%. And, the content distribution process can be accelerated a lot.

The reminder of this paper is organized as the following. Section II describes our measurement results on 6 PT networks, which disclose the problem addressed in this work. In Section III, we propose the framework of CrossPT to support efficient content sharing among PT networks and in Section IV, we analyze the strategies of PTs joining CrossPT. The evaluation results based on crawled data from active PTs are presented in Section V, which show the correctness of our analysis and the efficiency of the proposed framework. The discussions are given in Section VI. And finally in Section VII and Section VIII, we present the related work and conclusion remarks of the paper.

II. Measurement and Problem

We have crawled data from six PTs for about three months from Dec. 7, 2011 to Mar. 3, 2012. The dataset is used to verify the poor sharing problem among PTs and to later evaluate the framework/mechanism proposed in this paper.

The statistical information about the dataset we obtained is demonstrated in Table I. As the consideration on privacy, we omit the names of the sites in the paper and instead, we use PT1-PT6 to identify them. Each PT site has its own taxonomy of content categories. In order to make a uniform comparison, we classify the contents into four types by grouping some of their original categories[1]: "video", "Music", "Game" and "Others". Obviously from the table, contents of category "video" are the majority according to either the number of torrents or the volume of the files except PT5. Although the number of the videos is less than the number of music in PT5, videos still dominate the file volume. Overall, about 65.22% of the contents are videos, which occupy 80.99% of the storage to keep all of these contents. The variance of file volume is between 51.72% to 90.89% among these six PTs. The "Music" category is the second largest one in most of PTs (PT1-PT5), which shows 23.13% of the files containing 12.24% of volume. The distribution and the percentage of other kind of content can also be easily calculated by Table I. We depict the daily increase of the contents in the six sites in Figure 1. Both the number of contents and the size of contents show an increase trend.

With the above data collected from active PTs, we check the occurrence of same contents in different PT sites. To identify the same contents in different PTs, we employ two methods.

First, each torrent has a unique fingerprint called an "infohash". This infohash value is utilized to ensure the completion of content download to be uncorrupted. So if the files have the same *infohash*, they are definitely the same contents. However, PT sites may make minor modifications to the original content (*e.g.*, adding watermark of the sites), and the infohash value could be changed. Although these minor modifications are made, it is proper to still consider that these revised ones carry the same contents with the original ones, as [4] did. With this purpose, we have the second method based on the principle that classifies the torrents with the same content name and similar size of the same contents. Please note that the content name is extracted from the file name but it is not the same as the file name. The torrent title usually contains, taking videos as example, the video name, the date, the codec, etc. So we need extract the real content name from a torrent's title before we comparing the content size and picking out the same contents. The detailed algorithm to extract the content name and then detecting the same content is described in Appendix A. Table II illustrates the percentage of copies of the same contents among the six PT networks, or in other words, over 91.15% of the contents do not have any copies in other PT networks and less than 8.85% of contents have occurred at least twice. If one wants to find content in any of the six PTs, the chance is only 0.0021%.

The measurement results have clearly demonstrated that the content sharing degree among different PTs is quite low. For this reason, it is very frequent for the users of a PT community to encounter the difficulty in finding their interested contents. In the other hand, the download in public BitTorrent sites is

[1]Type "videos", "teleplays", "documentaries", "cartoons", etc. are grouped as a single group"Video"; "Music", "MVs" etc. are merged into the type "Music"; and "PC games", "PSP", etc. are all viewed as type "Game"; and all the categories, like "Software" are classified as "Others".

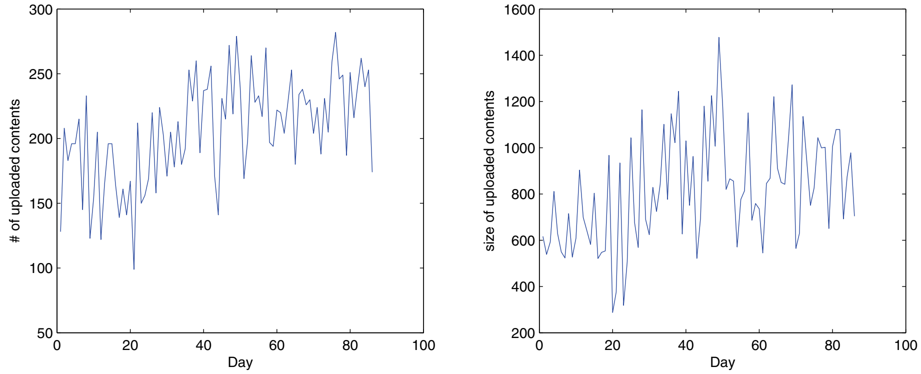

Fig. 1. Daily increase of the contents in six PTs. The left figure shows the number of uploaded contents and the right figure shows the size of uploaded contents.

TABLE I
THE STATISTICS ON THE DATASET WE HAVE COLLECTED FROM SIX PT SITES

	Video		Music		Game		Others	
Site	# Torrents	Size (GB)	# Torrents	Size (GB)	# Torrents	Size (GB)	# Torrents	Size (GB)
PT1	30350	188085.1	5697	14809.4	2194	6119.6	7101	22902.6
PT2	6348	46796.2	1096	3722.6	168	644.8	3493	9099.5
PT3	4573	47949.7	964	4111.4	3	4.36	211	688.7
PT4	10545	105018.8	1363	5117.4	92	416.2	515	1246.9
PT5	187	4032.1	1121	3756.6	3	5.32	2	1.41
PT6	54033	95689.6	1992	302.7	3047	9514.1	1495	59.6

TABLE II
FREQUENCY OF OCCURRENCE OF THE SAME CONTENTS IN DIFFERENT PT SITES.

occurrence	1	2	3	4	5	6
%	91.1498	6.3943	1.6941	0.6317	0.1280	0.0021

always slow or the contents disappear soon due to the lack of seeding incentive. The best way left for people today seems to be trying to register multiple accounts in different PTs for what they want. In fact, the situation does not improve much because of the control over user population and the less content diversities in a single PT compared with a public BitTorrent. The population of the users in one PT can not be arbitrarily large due to the management cost of each user's download/upload information. If we could enable the sharing of contents among the PTs, it could be much easier for users to get their interested contents and also a PT involved in a "sharing system" could therefore attract more users.

The idea of building a content sharing system among PTs is similar as the routing in Internet. The Internet is divided into a lot of Autonomous Systems (AS), each of which is usually operated by an Internet Service Provider (ISP) presenting a common, clearly defined routing policy to the Internet [11]. An AS runs an intra-domain routing protocol (*e.g.*, IS-IS, OSPF) to select the paths along which to send traffic inside its own domain, and meanwhile it typically has multiple independent connections to other ASes adhering to its clearly defined routing policy in the inter-domain routing protocol (*e.g.*, BGP). The current state of content sharing mechanism inside a PT or public BitTorrent is just as the function of intra-domain routing to the overall Internet routing, and what we aim to do in this paper is to add the missing "inter-domain" part of the content sharing among PTs.

III. DESIGN OF CROSSPT

In this section, we present the framework of sharing contents among PTs, which is called CrossPT. The high level framework of CrossPT works in two planes. In the *data plane*, CrossPT defines a mechanism so that the data can be transferred among PT networks and in the *control plane*, different PT networks determine their own *sharing strategies* (how much contents to share) and *fetch strategies* (where to fetch the interested contents) under CorssPT relationships/rules. With the help of CrossPT, one PT network can connect with several other PT networks. Its connected PT neighbors are called *friend PTs* and this PT network itself is recognized as the *home PT* of all its registered users.

A. Data Plane

If a PT network (*e.g.*, PT network A) participants in CrossPT, when a content requested by its member is not available in its own community, CrossPT will check this content with its friend PTs. If this content is verified by a friend PT network (*e.g.*, PT network B), PT network A will copy the specific content from PT network B. However, a PT network originally prevents the download from the outside of its PT community. To enable the data distribution among sharing PT networks with only small modifications to current PT data distribution mechanism, CrossPT allows any two PT communities to set a tunnel between them for content sharing.

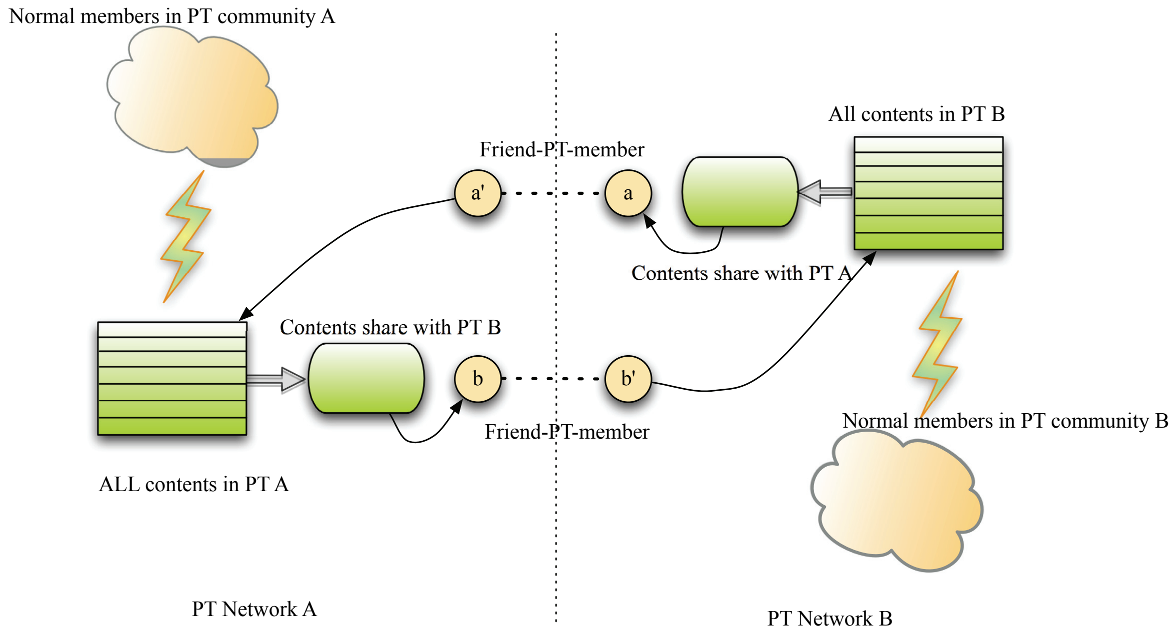

Fig. 2. Descriptive architecture of the content distribution between two PT communities.

The basic idea is to create a member account in a PT community for each of its friend PT networks, and the membership (which we call *friend-PT-member*) acts as the bridge between two friend PT networks. Suppose we have two participant PT communities, A and B. PT network A creates a member for PT network B (as "b" in Figure 2) and PT network B also allows a friend member for network A (denoted as "a" in Figure 2). The friend-PT-member automatically starts to download the requested content from the friend PT network (as long as the strategy allows) and then works as the seeder of this content in its home PT network after completion of the download. For the example in Figure 2, the home PT networks of "a" and "b" are PT network A and B, respectively. It is worth noting that the friend member has a "primary image" in its home PT networks. It is a logical member account in the home PT network used to seed the content. As shown in Figure 2, "a′ "and "b′ " are the primary images of "a" and "b". Let us imagine the following scenario for illustration. Suppose that a registered user (say "s") in PT network A requests a content X which its home PT does not have. PT network A checks its friend PT members in its friend PTs and finds that PT network B has it. In this case, the member "a" starts to downloads X from PT network B. Every downloaded bit of X is copied to the primary image "b′" of "b" and the requested user "s". The download process from PT network B is transparent to "s" and the experience of "s" just like a normal download from another member "b′" in the same PT community. After "b′" obtaining 100% of the data, it automatically becomes a seed. As a result, PT network A now owns the content X. A friend-PT-member and its primary image together provide a proxy-like function to connect the two PT networks and thus it is different from the concept of "super node" in P2P networks, which is a single node working as both relayer and proxy servers.

For the ease of implementation, in this paper, we suggest each participant PT to set a dedicate member for each friend PT. These memberships and all download from friend PTs can be physically operated in a single machine/server. Although it increases the cost for a PT to participant CrossPT, PTs still have incentive to do so as we presented in Section III-C. Also, it is possible to select an already existing member acting as the friend-PT-member. For example, the one requesting the content only available from other PTs. Such kind of solution will not bring more traffic for the member, who acts as the friend-PT-member.

B. Control Plane

Each PT network holds its own list of torrent files. The whole list of all the available torrents is available to its registered members, and information of part or all the available torrents can be shared to other PT networks. A PT network sets a "sharing strategy" to determine which content to be shared and only the strategy-compliant torrent information is visible to other PTs. In addition, a "fetch strategy" is also applied in each PT network, which selects a PT network to download since it is possible that the interested contents can be found in multiple PT networks. CrossPT gives the flexibility to participant PTs and they can select their own strategies. Although the detailed sharing strategy and fetch strategy are left to each PT networks, we can analyze how rational PT participant select its sharing strategy and study what is the optimal fetch strategy to a optimal problem. We will discuss it more in Section IV.

C. Coordination Rule and Incentives

Before we analyzing how a PT selects its own strategies in the next section, we first present the coordination rules defined by the CrossPT for the PTs which provides incentive to PTs to share their contents. A coordination rule among the sharing

PTs is required to guarantee the rights and benefits of each PT network. Similar as the incentive stimulating mechanism in a single PT, a download-to-supply ratio (R) is recorded for each of the PT participant. Differently, there is no central controller to maintain such information. Since one PT knows how much a friend PT has download from itself and how much it has retrieved from a friend PT, a list of R can be maintained in each PT network for all its neighbors friend PTs. Two neighbor friend PT networks freely exchange content between them if the download-to-upload ratio is less than a predefined threshold T. And if one of them downloads too much (*i.e.*, $R > T$), its further request on sharing would be ignored unless they could upload more first. We call the relationship of the two friend PTs applying this rule as peering relationship (two more flexible relationships are discussed in Section VI). With this strict rule, in the initial state, no one can download and thus no one could provide contents. So a bootstrap would trigger in the initial: each PT has a small quantum for download without the calculation of R, after the consumption of the quantum, the above rule start to apply.

PTs have incentives to join the CrossPT for several reasons. First, friend PTs have complementary contents, which attract the interests of PT to get. Second, to download from friends requires sharing to others before, and this stimulates PT to provide contents to others. Such a similar mechanism is proved to be successful in a single PT. Third, the calculation of R is distributed in each PT and can be verified by all the PT neighbors. No one can game on the coordination rule. In this way, the mutual content exchange between the two participants is relative comparable.

IV. Strategies

In this section, we will provide the method and the analysis on how PTs select their sharing strategies and fetch strategies.

A. Sharing strategy

Since CrossPT only involves the relationship between two PT networks, we consider in our analysis a two-player game between two PT communities, and the game can be defined as the following.

- Players: PT network A, PT network B.
- Actions: Each network determines the (amount of) contents to be shared. We denote the contents' amount as S_{AB} (A provide to B) and S_{BA} (B provide to A).
- Utility: The expected utilities of the parties are U_A and U_B.

$$U_A = b_A(S_{BA}) - c(S_{AB}) \tag{1}$$

$$U_B = b_B(S_{AB}) - c(S_{BA}) \tag{2}$$

Where $c(\cdot)$ is the cost function and this function is supposed to be the same for all the PT networks, and $b_i(\cdot)$ is the benefit that one PT network could obtain from a friend PT network. The value of downloading from other PT networks can vary for PTs, so a subscript i is used to denote the differences. Basically, the choices of the $c(\cdot)$, $b_i(\cdot)$ do not affect our analysis only as long as they are monotonically increasing functions of the sharing bytes. Please note that the subtraction on cost function and the benefit function to achieve utility function is only meaningful when the two functions' values are renormalized to a same unit. Without explanation, the following formulations of $c(\cdot)$, $b_i(\cdot)$ are renormalized.

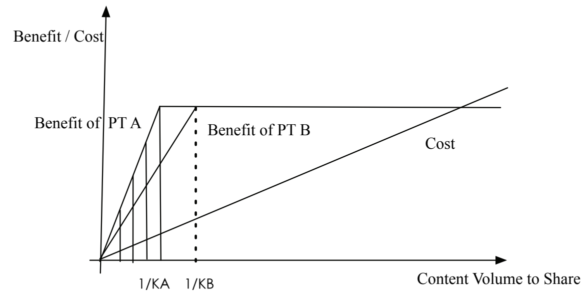

Fig. 3. Benefit and cost function in the analysis. The shadow area stands for area of the feasible solutions.

Since the cost is mainly the bandwidth consumed to upload the shared contents to others, in the analysis, we define $c(\cdot)$ as a function that is proportional to the bytes shared with others. The coefficient a in the following form of the function is a pre-defined constant.

$$c(S) = aS; \tag{3}$$

More downloading surely increases the benefit and the overall utility, but the benefit reaches the maximum when all the requested/interested contents are downloaded. So it makes sense that the desired benefit function is a proportional curve and remains unchanged after reaching the maximal value. Formally in this paper, we use the following.

$$b_i(S) = Min\{K_i S, M\}; \tag{4}$$

The parameter M is normalized to be the same for all the PTs, which is predetermined to denote for the maximal benefit. And K_i is also a content parameter that makes the benefit reach the maximum after all the requested contents are obtained. That is why PTs have different preferences over different contents and we use the subscript i of K_i to reflect the preferences. The two functions are depicted in Figure 3.

After the definition of the cost function of benefit function, we rewrite utility function of the two PT networks during the game.

$$U_A = Min\{K_A S_{BA}, M\} - aS_{AB}. \tag{5}$$

$$U_B = Min\{K_B S_{AB}, M\} - aS_{BA}. \tag{6}$$

Suppose that S^*_{AB} and S^*_{BA} are the Nash equilibrium actions for PT network A and B in the game. According to game theory [12], both PT network A and PT network B try to maximize their own utilities in the Nash equilibrium.

$$Max.\ Min\{K_A S^*_{BA}, M\} - aS_{AB} \tag{7}$$

$$Max.\ Min\{K_B S^*_{AB}, M\} - aS_{BA} \tag{8}$$

Also, we have

$$S^*_{BA}/S_{AB} \le T \tag{9}$$

$$S^*_{AB}/S_{BA} \le T \tag{10}$$

Since each of the players aims to maximize his own utility, it is not difficult to find the Nash equilibrium in Figure 3 as

$$S^*_{AB} = S^*_{BA} = \begin{cases} 1/K_A \text{ if } K_B \leqslant K_A \\ 1/K_B \text{ if } K_A \leqslant K_B \end{cases} \tag{11}$$

It is worthy of noting that the Nash Equilibrium in this case does not reflect the largest social welfare. The social welfare of the two-player game in Figure 3 is the sum of the utilities of the two PT networks. When Nash Equilibrium is met, the social utility is $SU_{ne} = (K_A+K_B-2a)/K_A$. The largest social welfare should be $SU_{max} = 2-2a/K_B$ in the case of Figure 3 if $K_B > 2$ and both two PT networks provide content of size $1/K_B$.

B. Fetch Strategy

For an interested content, a PT network can have several choices to select a friend neighbor for downloading, and thus there is a neighbor selection problem (NSP): which friend neighbors should be selected to download the contents. In this section, we first formally define the problem and then propose a method to determine how to select friend neighbors to download the contents.

Each PT community makes its own decision independently. Without loss of generality, we formulate the selection behave of one PT network. We denote the set of interested contents in all other PT communities as $\mathbb{N}$ and the number of such contents is $n = |\mathbb{N}|$. $\mathbb{L}$ is denoted as the set of all the friend PT networks and $l = |\mathbb{L}|$ is the total number of friend neighbor networks. $\mathbb{D} = \{d_j\}$ is used to denote the size of the interested content j of the PT network. $\mathbb{M} = \{m_{ij}\}$ is an indicator variable matrix, where $m_{ij} = 1$ if the friend neighbor PT network i have a copy of the content j and $m_{ij} = 0$ otherwise. In the decision matrix $\mathbb{Q} = \{q_{ij}\}$, $q_{ij} = 1$ if the PT network finally selects to download from i for the content j and $m_{ij} = 0$ otherwise. The quantum that the PT network can download from PT network i is denoted by S_i. The NSP can be formulated as follows.

$$Max. \quad \sum_i \sum_j q_{ij} d_j \tag{12}$$

$$St. \quad \sum_i q_{ij} \leq 1 \tag{13}$$

$$\sum_j q_{ij} d_j \leq S_i \tag{14}$$

$$q_{ij} \leq m_{ij} \tag{15}$$

$$q_{ij} = \{0,1\}; m_{ij} = \{0,1\} \tag{16}$$

In the above problem definition, (12) is the objective that tries to maximize satisfied demand of the PT network. (13) indicates that the same content is only downloaded at most once from some friend neighbor. Next, we give the constraint on the download quantum on each friend neighbor PT network in (14). And the constraint in (15) limits the download from the PT network that contains the interested content j.

Theorem 1: NSP is an NP-complete problem.

Proof:

The above problem can be reduced to the well-known Generalized Assignment Problem (GAP) [13]. The GAP can be described as follows. Given a number of *items* (*e.g.*, x) and *knapsacks* (*e.g.*, y), each item j has a profit of p_{ij}, as well as a weight w_{ij} if assigned to knapsack i. In addition, each knapsack i has a capacity that the greatest weight it can carry is c_i. The GAP is to assign each item to exactly one knapsack so that the profit assigned is maximized, without assigning to any knapsack a total weight larger than its capacity. To reduce our NSP to GAP, the contents are mapped to items and the friend neighbor PT networks are reduced to knapsack. $\forall i \in \{1,2,3,\cdots,l\}, w_{\cdot j} = d_j$. If $m_{ij} = 1$, $\forall i \in \{1,2,3,\cdots,l\}, p_{ij} = d_j$, otherwise, we set p_{ij} to be zero or a negative value. In this way, the items (contents) will not be assigned to a knapsack (friend neighbor PT network) if an unexpected assignment between them is defined ($m_{ij} = 0$). Till now, we still have a small gap between GAP and NSP. GAP expects an exactly assignment of each item ($\sum_j q_{ij} = 1$), while the NSP relaxes this constrain to $\sum_i q_{ij} \leq 1$. To fill this gap, an additional knapsack of capacity $c_{y+1} = x$ is introduced to the GAP instance, and $p_{y+1,j} = 0, w_{y+1,j} = 1$ for $\forall j$. With the above reduction, the solution of q_{ij} is also recognized as the solution of GAP. In this way, GAP and NSP can be reduced to each other in polynomial time, and since GAP is known as an NP-complete problem, NSP is also NP. Furthermore, it is obvious that the verification of NSP can be done in polynomial time, therefore, NSP is also NP-complete.

■

Algorithm 1 Heuristic algorithm for NSP

Initial: $q_{ij} = 0$;
sort the requesting contents from the largest to the smallest in the set N;
while N is not empty **do**
  $j \leftarrow$ the first element in N;
  remove the first element from N
  $MaxS = 0$;
  $x = 1$;
  while $x <= l$ **do** ▷ Search for the largest S_i with content j
    if $m_x j = 1$ **then**
      if $S_x > MaxS$ **then**
        $i = x$;
        $MaxS = S_x$;
      end if
    end if
    $x++$;
  end while
  if $S_i \geq d_j$ **then**
    $S_i = S_i - d_j$;
    $q_{ij} = 1$;
  end if
end while

To solve NSP, we propose an approximate algorithm. The heuristic behind the algorithm is that a PT prefers to download content with the largest size from a friend PT having the largest rest capacity.

V. Evaluation

With the help of CrossPT, contents will distribute among PTs and after such a distribution process, each PT obtains more contents from its friend PTs. In the evaluation part, we mainly check two aspects of the performance. First, we aim to find out how much contents a PT participant in CrossPT could obtain from other friend PTs. Second, we exam the distribution delay of the contents over CrossPT. The entire delay consists

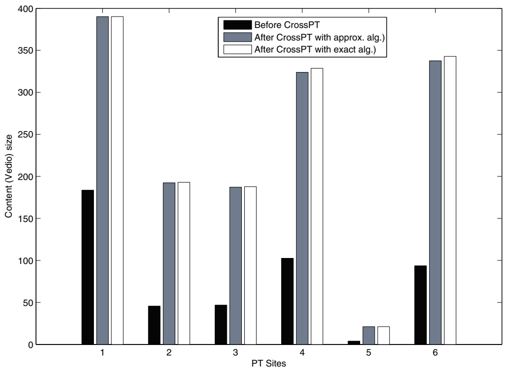

Fig. 4. Content distribution improvement with CrossPT

PT2	55.23				
PT3	55.61	53.74			
PT4	75.95	73.40	72.49		
PT5	36.12	36.37	30.45	34.80	
PT6	30.89	40.05	43.79	30.44	4.72
	PT1	PT2	PT3	PT4	PT5

Average interval of uploaded date (measured by day)

PT2	916.11				
PT3	1024.07	825.99			
PT4	997.53	827.25	744.45		
PT5	363.75	248.85	311.13	71.40	
PT6	1014.70	822.30	507.21	535.06	8.06
	PT1	PT2	PT3	PT4	PT5

Maximum interval of uploaded date (measured by day)

Fig. 5. Content Distribution Delay before employing CrossPT

TABLE V
COMPUTATIONAL TIME OF THE FETCH STRATEGY.

PT Sites	PT1	PT2	PT3	PT4	PT5	PT6
Approximation	4.2 s	5.5 s	5.7 s	5.3 s	5.8 s	2.3 s
Exact Algorithm	164 s	956 s	958 s	284 s	958 s	115 s

of the delay in both data plane (content delivery) and control plane (strategy determine) of CrossPT. The content delivery time is proportional to the size of the content and we only perform experiments to show the delay in the control plane. Again, we use the data collected from the six running PT sites, whose detailed information has been described in Section II (Table I). Since most of the contents of these PT sites are videos according to the statistics in Table I, we carry out the evaluation experiments on video contents.

In our experiment, each PT keeps a list of all its own contents (with content name and size) and has knowledge of the content in other PTs. Each PT first decide how much to share according to the analysis we derived in Section IV-A, and then calculate its fetching strategy by solving the NSP model in Section IV-B. Beside Algorithm 1 proposed to solve the NSP model, we also compute the solution using general optimizer LINGO [14]. LINGO seeks the optimal value using the branch-and-bound technique, which can be viewed as an exact algorithm. The results from LINGO are employed to make comparisons with our heuristic algorithm. The improvement of contents diversity in each PT sites are illustrated in Figure 4 and Table III. The “approximation” in the table stands for the calculation time using Algorithm 1 and the “exact alg.” is related to the calculation using LINGO [14]. The content size of any PT site in our experiment has a significant increased from 113.95% to 438.46%. The absolute increase of content is proportional to the original content size of a PT site (as shown in Figure 4), as we analyzed in Section IV that, one site will benefit more by sharing more. But the improve percentage in the size is larger for the sites that has less content before joining CrossPT (as shown in Table 4). It is also indicated in Figure 4) and Table III that, the optimization result using approximate algorithm is quite close to the results obtained from exact and general optimizer LINGO. Also, we exam the overlapping content after employment of CrossPT. Table IV indicate the percentage of contents that appear at least in x PT sites, where $x = 1, 2, 3, 4, 5, 6$. Compared with Table II in Section II, the content sharing degree is improved a lot. Around 90% of the contents would be included in at least two PT sites, and more than 80% of the contents would distributed to half of the PT sites (here is three PT sites in our experiment).

In Table V, the calculation time for both our heuristic and LINGO is presented. The calculation time is sure determined by the service we used to run the algorithms. In the experiment, we use a server, which is equipped with two Intel E5606 CPUs (2.13GHz, four cores in each CPU) and 32 GB memory. It is indicated in Table V that the approximate algorithm accelerates the computation by 39-174 times, compared to the exact computation using LINGO. Without CrossPT, PTs could have a small number of contents in common. We compare the upload time of these same contents in every two PTs and list the differences of the upload time of a same content. Figure 5 presents statistics of the average and the maximum interval of uploaded date of the same video on different sites. The result shows that the average interval is about one to three months while the maximum interval could be as long as three years. Although we can not get exact number on the content delivery speed between any two PTs, the above results can give us some insights on how much CrossPT could accelerate the content distribution process.

Furthermore, we implement a prototype system with 3 nodes in Xian (China), Beijing (China), Chicago (US) and each node plays the role of PT manager, where the requests, replies are sent and the strategies are calculated. We have a pool of 20000 contents, and each PT randomly selects part of them as the initial contents in the PT community. In the experiment on the prototype system, we aim to check the total time latency required for a PT to decide its strategy to request contents from friends. Each PT manager first generates requests to friend PT for the contents that are not in its own PT community. On receiving a request from other CrossPT participants, a PT manager replies a list of requesting contents it has and the content size. Based on the responses from friend PTs, the requesting PT calculates its strategy using the methods proposed in the previous section of this paper. The measured delay in this experiment is illustrated in Figure 6. The latency in this small prototype costs only several seconds.

TABLE III
INCREASED CONTENT SIZE WITH CROSSPT IN EACH PT SITE.

PT Sites	PT1	PT2	PT3	PT4	PT5	PT6
Approximation	112.43%	320.84%	299.73%	215.68%	437.65%	261.13%
Exact Algorithm	112.43%	322.17%	301.42%	220.43%	437.65%	266.81%

TABLE IV
OVERLAPPING CONTENT IN DIFFERENT PT SITES. THE PERCENTAGE OF CONTENTS SIZE THAT APPEAR AT LEAST IN x ($x = 1, 2, 3, 4, 5, 6$) PT SITE

At least in x sites	$x = 1$	$x = 2$	$x = 3$	$x = 4$	$x = 5$	$x = 6$
Approximation	1.0	0.8644	0.8274	0.4975	0.4724	0.0542
Exact Algorithm	1.0	0.9204	0.8589	0.4790	0.4321	0.0536

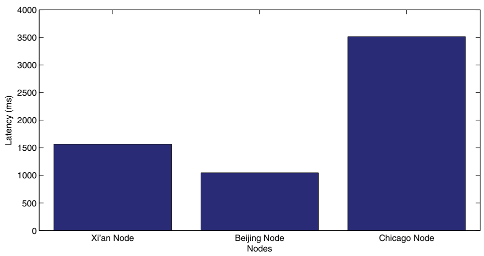

Fig. 6. Latency measured in the prototype system

VI. DISCUSSION

Besides the peering relationship we mentioned in Section III, two more relationships between friend PT neighbors can be introduced if "pay for contents" is allowed. The first one is *provider-to-customer* relationship. As a provider-to-customer relationship, the PT network acting as the customer needs to pay for the contents it copies from the provider PT network with a price of per unit (the unit can be B, MB or GB). Also, there can be a more flexible relationship than peering and provider-to-customer, which we call *mutual-helper* relationship. With this mutual-helper relationship, two participants in fact maintains peering relationship when the download-to-upload ratio is less than a predefine threshold T. If one of them wants download more, it needs to pay for the extra bytes that make $R > T$.

We illustrate an example for intuitive explanation of the mutual-helper relationship. Suppose we have a participant PT network A downloads from another PT network B S_{BA} bytes and the participant B downloads S_{AB} from the PT network A. The download-to-upload ratios of the PT network A and B are $R_{BA} = S_{BA}/S_{AB}$ and $R_{AB} = S_{AB}/S_{BA}$, respectively (obviously, $R_A = 1/R_B$). If both PT network A and B well control the ratio between downloads and uploads, *i.e.*, $R_A < T$ and $R_B < T$, the two PT communities freely exchange their contents. But in case that PT network A wants to download more from the PT network B over its free quantum, *i.e.*, if $R_A = S_{BA}/S_{AB} > T$, PT network A needs to pay $M(D_A - TD_B)$ to PT network B. Please note that S_{BA} and S_{AB} are computed once after a fixed time interval to determine the payment, for example, every week or every month or every year.

The strategies investigated in this paper indicate that one can only download the whole content from one neighbor. It is obvious that the share ratio would be much better if we allow users to download a same content from different friend PTs. In order to get the optimal sharing in this settings, we need to change the definition of q_{ij} from $\{0, 1\}$ binary variable to a real number variable valued between 0 and 1. The heuristic developed in this paper can be extended to this problem easily. However, it is a big overhead for a PT to coordinate the downloading from different friend PTs.

VII. RELATED WORK

BitTorrent has attracted a large number of academic researches. A big problem in such a P2P file sharing system is the so-called "free-riding" problem: peers do not contribute after download [15, 16]. Even several greedy BitTorrent clients dedicated for free riding have been developed to optimized the performance of users without upload anything, *e.g.*, BitThief [17] and BitTyrant [18]. The impact of greedy strategies in BitTorrent network was studied in [19]. Game theory has been used to model the selfish behavior of P2P participants and to solve the free-riding problem [20, 21]. in [20], the authors studied the interaction of strategic and rational peers with the help of game theory. And [21] proposed an incentive mechanism based on reputation and model it as an infinitely repeated game. In the paper of [22], the sharing ratio enforcement mechanism was investigated and four different strategies have been proposed to solve the oversupply effects. The inter-swarm resource allocation mechanisms are studied in [23], which basically tried to maximize the throughput and does not solve the sharing problem addressed in our paper.

There are also several literatures working on PTs. M. Meulpolder et al. have presented extensive measurements of over half a million peers in two public and three private BitTorrent communities [7]. They obtained several observations in PT communities through their study: 1) the download speeds are 35 times higher than in public communities, 2) more peers are connectable, 3)the seeder-to-leecher ratios are at least ten times as those in public communities, 4) peers seed for a significantly longer duration. C. Zhang et al. studied more than 800 PTs based on geography and content distribution in [4], where they compared public tracker sites and PTs comprehensively from macroscopic, medium-scopic and microscopic perspectives. Z. Liu et al. studied in-depth SRE in PTs [9] and pointed out that PTs provide more incentives for members to seed/upload. They also proved that SRE is effective with the help of a game theoretic model and proposed an upload entropy scheme to prevent collusions . X. Chen et al. studied 13 PTs and 2 public tracker sites from the user viscosity, torrents evolution, user behaviors, content distribution and other metrics [8]. Another work has crawled data from 17 private trackers, 2 public trackers and 1 BitTorrent search engine for 6 months [6], and showed that 1) private trackers have highly levels of member and torrent engagement, 2) the Poor Downloading Motivation problem is existed. The Poor Downloading Motivation problem is further studied in [5], where the authors introduced Predator-Prey model in ecology to analyze and achieve the stable SLR range [5]. In addition, Y. Zhao et al. combined related behaviors to a single lifecycle and measurement peer-to-peer system in the lifecycle based view [10], they also modeled the swarm evolution to show the long term characteristics.

VIII. Conclusion

A private BitTorrent maintains an upload-to-download ratio for each of its users, and the ratio is utilized to decide the content volume that could be downloaded. In this way, the PT users have strong incentive to seed the contents after downloading. However, the maintenance cost of user information limits the population of a PT BitTorrent site and thus the contents contributed by the users are a little bit limited compared with a public BitTorrent site. Although there are over 1000 PT sites nowadays, no any content sharing among these sites are available. In order to remedy the content distribution problem among PT sites, in this paper, we have proposed CrossPT as a systematic solution. Under CrossPT framework, we further investigate the sharing strategy and fetch strategy of PT participants through game theory and modeling. In addition, we have collected real torrent information from six active PT sites, and leveraging the data, we evaluate what we have proposed. The evaluation results demonstrate the efficiency of CrossPT in both increasing content diversity and decreasing content distribution delay.

Appendix A
Detection of same content from Torrent Tittle and content size

Torrent tittle (hereinafter to be referred as tname) cannot completely represent for content name. The steps for the extraction are concluded as follows:

1) If tname contains year, then extract substring before the year from it as new tname. E.g., suppose that tname is Transformers Dark of the Moon 2011 DVDRip XviD-TWiZTED. After extracting, tname is changed to Transformers Dark Of The Moon.
2) If tname contains bluray, minihd, dvdrip, bdrip, brrip, hdtv, tvrip, webrip, hddvd, ***x*** (* represents from a number?similarly hereinafter), ***p, ****p, etc. (Case insensitive), then extract substring before them as tname. E.g., suppose that tname is Transformers Dark Of The Moon 720p Bluray x264-MHD. Then it becomes Transformers Dark Of The Moon after extracting.
3) If tname contains DVD, rmvb, mkv, iso, mp4, then extract substring that before them as tname.
4) Extract substring that starts after the last character] in tname. E.g., if tname is [Tamas Wells] A Mark On The Pane FLAC, then it turns into A Mark On The Pane FLAC after extracting.

After these four steps, the string we get (hereinafter to be referred as mName) can roughly represent for the content name. In order to increase the accuracy, we use the first three rules repeatedly. However, the same mName doesnt necessarily mean the same movie. Consequently, there are two rules to compare the torrents: the same mName and the difference between sizes of movies is less than 1% of the size of each content. The number of torrents is considerable and we need to compare the contents in 6 sites. As a result, we should take an efficient algorithm. We can sort the data of every site by mName, size of files, uploaded date. After that, find out the minimum data of 6 sites and compare it with data of other sites. The time complexity is $O(n \log n)$, where n is the number of contents. Flow chart of this algorithm is as shown in Figure 7.

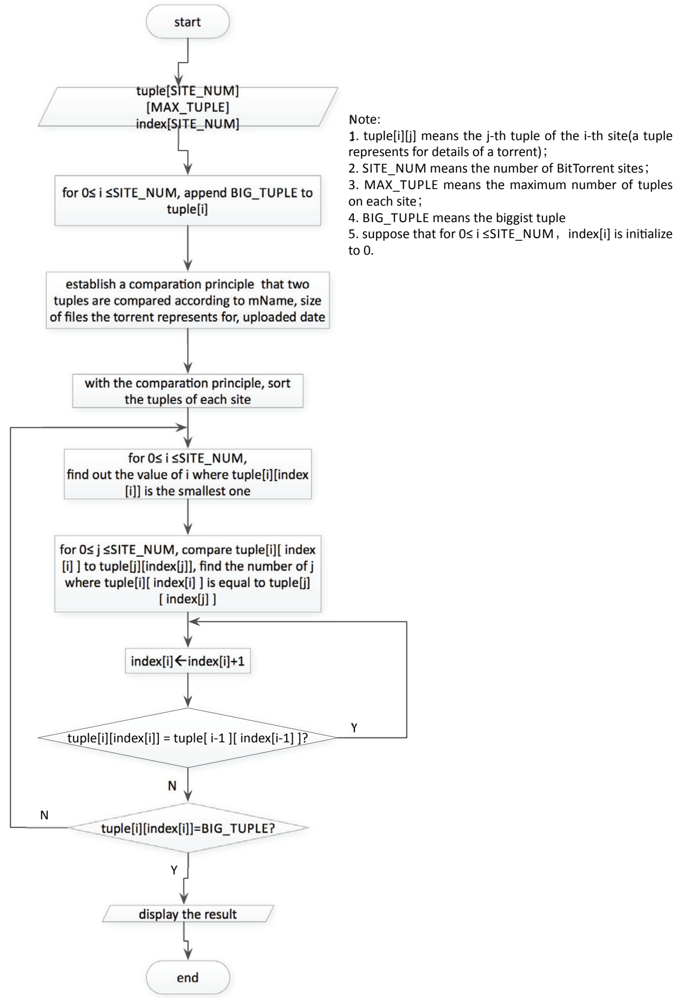

Fig. 7. Algorithm to classify the same content.

References

[1] R. Schollmeier, "A definition of peer-to-peer networking for the classification of peer-to-peer architectures and applications," in *P2P '01*, Washington, DC, USA, 2001.
[2] H. Schulze and K. Mochalsk, "Internet Study 2008/2009," ipoque, Tech. Rep., Feb. 2009.
[3] VNI, "Cisco Visual Networking Index: Forecast and Methodology, 2011-2016," Cisco, Tech. Rep., May 2012.
[4] C. Zhang, P. Dhungel, D. Wu, Z. Liu, and K. Ross, "Bittorrent darknets," in *INFOCOM'10*, Mar. 2010.
[5] X. Chen, X. Chu, and Z. Li, "Improving sustainability of private p2p communities," in *ICCCN'11*, Aug. 2011.

[6] X. Chen, X. Chu, and J. Liu, "Unveiling popularity of bittorrent darknets," in *Global Telecommun. Conf. (GLOBECOM 2010), 2010 IEEE*, Dec. 2010.
[7] M. Meulpolder, L. D'Acunto, M. Capotă, M. Wojciechowski, J. A. Pouwelse, D. H. J. Epema, and H. J. Sips, "Public and private bittorrent communities: A measurement study," in *Proc. 9th International Conf. Peer-to-peer Syst. (IPTPS'10)*, 2010, pp. 10–10.
[8] X. Chen, Y. Jiang, and X. Chu, "Measurements, analysis and modeling of private trackers," in *2010 IEEE Tenth International Conf. Peer-to-Peer Comput. (P2P),*, Aug. 2010, pp. 1–10.
[9] Z. Liu, P. Dhungel, D. Wu, C. Zhang, and K. W. Ross, "Understanding and improving ratio incentives in private communities," in *2010 IEEE 30th International Conf. Distributed Comput. Syst. (ICDCS)*, June 2010, pp. 610–621.
[10] Y. Zhao, Z. Zhang, L. Guo, and B. Fang, "Revisiting the swarm evolution: A long term perspective," in *2011 IEEE Symp. Comput. Commun. (ISCC)*, July 2011, pp. 1062–1067.
[11] J. Hawkinson and T. Bates, "Guidelines for creation, selection, and registration of an autonomous system (as)," *RFC 1930*, 1996.
[12] R. Gibbons, *A Primer in Game Theory*, 1st ed. London: Harvester Wheatsheaf, 1992.
[13] H. Kellerer, U. Pferschy, and D. Pisinger, *Knapsack Problems*. Springer-Verlag, 2004.
[14] LindoSystem, "Lingo 13.0 - optimization modeling software for linear, nonlinear, and integer programming," [Online]. Available: http://www.lindo.com/index.php?option=com_content&view=article&id=2&Itemid=10, 2012.
[15] M. Feldman, C. Papadimitriou, J. Chuang, and I. Stoica, "Free-riding and whitewashing in peer-to-peer systems," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 5, pp. 1010–1019, May 2006.
[16] J. Mol, J. Pouwelse, D. Epema, and H. Sips, "Free-riding, fairness, and firewalls in p2p file-sharing," in *P2P '08.*, Sept. 2008.
[17] T. Locher, P. Moor, S. Schmid, and R. Wattenhofer, "Free riding in bittorrent is cheap," in *HotNets-V*, Irivine, CA, USA, 2006.
[18] M. Piatek, T. Isdal, T. Anderson, A. Krishnamurthy, and A. Venkataramani, "Do incentives build robustness in bittorrent?" in *NSDI*, Cambridge, MA, USA, 2007.
[19] D. Carra, G. Neglia, and P. Michiardi, "On the impact of greedy strategies in bittorrent networks: The case of bittyrant," in *Eighth International Conf. Peer-to-Peer Comput.*, 2008.
[20] C. Buragohain, D. Agrawal, and S. Suri, "A game theoretic framework for incentives in p2p systems," in *Third International Conf. Peer-to-Peer Comput.*, 2003.
[21] R. Gupta and A. Somani, "Game theory as a tool to strategize as well as predict nodes' behavior in peer-to-peer networks," in *11th International Conf. Parallel Distributed Syst.*, 2005.
[22] A. Jia, R. Rahman, T. Vinko, J. Pouwelse, and D. Epema, "Fast download but eternal seeding: The reward and punishment of sharing ratio enforcement," in *2011 IEEE International Conf. Peer-to-Peer Comput. (P2P),*, 31 2011-Sept. 2 2011.
[23] M. Capota, N. Andrade, T. Vinko, F. Santos, J. Pouwelse, and D. Epema, "Inter-swarm resource allocation in bittorrent communities," in *2011 IEEE International Conf. Peer-to-Peer Comput. (P2P)*, 2011.

Chengchen Hu (S'04-M'09) received his B.S. degree from the Department of Automation, Northwestern Polytechnical University, and his Ph. D. degree from the Department of Computer Science and Technology, Tsinghua University, in 2003 and 2008 respectively. He worked as an assistant research professor in Tsinghua University from Jun. 2008 to Dec. 2010 and is now an associate professor in the Department of Computer Science and Technology, in Xi'an Jiaotong University. His main research interests include computer networking systems, network measurement and monitoring. He severed in the organization committee and technical program committee of several conferences, *e.g.*, IWQoS 2010, INFOCOM 2012/2013, GLOBECOM 2010-2012, ICC 2011-2013, *etc*.

Danfeng Shan is an undergraduate student in department of Computer Science and Technology in Xi'an Jiaotong University and will be graduate in 2013. His research interests are in the areas of computer network.

Yu Cheng (S'01-M'04-SM'09) received the B.E. and M.E. degrees in Electrical Engineering from Tsinghua University, Beijing, China, in 1995 and 1998, respectively, and the Ph.D. degree in Electrical and Computer Engineering from the University of Waterloo, Waterloo, Ontario, Canada, in 2003. From September 2004 to July 2006, he was a postdoctoral research fellow in the Department of Electrical and Computer Engineering, University of Toronto, Ontario, Canada. Since August 2006, he has been with the Department of Electrical and Computer Engineering, Illinois Institute of Technology, Chicago, Illinois, USA, and now as an Associate Professor. His research interests include next-generation Internet architectures and management, wireless network performance analysis, network security, and wireless/wireline interworking. He received a Postdoctoral Fellowship Award from the Natural Sciences and Engineering Research Council of Canada (NSERC) in 2004, and a Best Paper Award from the conferences QShine 2007 and ICC 2011. He received the National Science Foundation (NSF) CAREER Award in 2011. He served as a Co-Chair for the Wireless Networking Symposium of IEEE ICC 2009, a Co-Chair for the Communications QoS, Reliability, and Modeling Symposium of IEEE GLOBECOM 2011, a Co-Chair for the Signal Processing for Communications Symposium of IEEE ICC 2012, and a Technical Program Committee (TPC) Co-Chair for WASA 2011. He is an Associated Editor for IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY and the New Books & Multimedia Column Editor for *IEEE Network*.

Tao Qin (S'08) received the B.S., M.S. and PHD degrees in information engineering from Xi'an Jiaotong University, Xi'an, China, in 2004, 2006 and 2010 respectively. He is currently an assistant professor with the Systems Engineering Institute and MoE KLINNS Laboratory of Xian Jiaotong University. His research interests include internet traffic analysis, abnormality detection and traffic modeling.