On the Impact of P2P Incentive Mechanisms on User Behavior

Manaf Zghaibeh
Technology Management Department
Stevens Institute of Technology
Hoboken, NJ 07030, USA
Email: Manaf.Zghaibeh@stevens.edu

Kostas G. Anagnostakis
Institute for Infocomm Research
21 Heng Mui Keng Terrace
Singapore 119613
Email: kostas@i2r.a-star.edu.sg

***Abstract*—In this paper we report on the results of a large-scale measurement study of two popular peer-to-peer systems, namely BitTorrent and eMule, that use practical and lightweight incentive mechanisms to encourage cooperation between users. We focus on identifying the strategic behavior of users in response to those incentive mechanisms.**

Our results illustrate a gap between what system designers and researchers expect from users in reaction to an incentive mechanism, and how users react to those incentives. In particular, we observe that the majority of BitTorrent users appear to cooperate well despite the existence of known ways to tamper with the incentive mechanism; while users engaging in behavior that could be regarded as cheating comprised around 10% of BitTorrent's population.

In the eMule system, we identify two distinct classes of users based on their behavior. The first class contains users who appear to perceive cooperation as a good strategy, and openly share all the files they obtained. The second class comprises users who engage in more subtle strategic choices, by actively optimizing the number and types of files they share in order to improve their standing in eMule's waiting queues; they tend to remove files for which downloading is complete and keep a limited total volume of files shared.

I. Introduction

Cooperative distributed systems, commonly known as "peer-to-peer systems" refer to a class of systems and applications that employ distributed resources to perform critical functions in a decentralized manner. The users of the system or the *peers*, are anonymous entities, sharing their resources to benefit each other within a scalable, stable and reliable global service.

Peer-to-Peer systems, *P2P*, have the potential of providing a wide range of services such as file sharing, content distribution and distributed data processing. They are often contrasted with client-server systems, because participants typically blur the distinction between client and server by acting as both consumers of the service (e.g., clients) as well as producers (e.g., servers).

These systems enable massive resource and information pooling at low cost per participant and at scales that are difficult to achieve with traditional client-server systems, while local autonomy and network effects provide resilience against failures and attacks.

One distinguishing characteristic of P2P systems compared to more traditional client-server designs is the widespread cooperation between participants by sharing resources and information. As all the benefits of these systems are deeply rooted in cooperation, they are inherently vulnerable to large-scale non-cooperative behavior. It is therefore necessary for the system to be designed so that participants generally cooperate. The mechanisms that are embedded in the system for this purpose are called *incentive mechanisms*.

Incentive aspects in P2P networks have been in the enhancement and the development loop since the emergence of this class of systems. Theoretically, and to fully utilize the resources of a P2P network, the system architect must have a precise understanding of the payoff of each individual user joining the network in order to construct an appropriate incentive mechanism [3]. However in practice, the architect is not able to have a clear vision of the requirements of every user. Even if he did, it is difficult to implement the incentives that would actually work. The difference between what researchers designed, and how users actually react to the proposed incentives is dependent on many factors. First, users in a P2P network perceive the proposed incentives in different ways depending on their own utility functions. Second, most of the proposed incentive mechanisms do not adapt to changes in users' behavior. Third, there is no extensive and reliable measurement study that would help P2P network designers evaluate the *real* performance of their incentive mechanisms with the *expected* performance. Most of the measurement studies completed on existing P2P networks demonstrate the success or the failure of such systems with no analysis or explanation on why these systems succeeded or failed.

In general, a failure of a P2P system is related to the lack of incentive acceptance among users, who disregard the system if they sense that it does not provide them with satisfactory payoffs. For example, some of the *payment-based* systems, such as the *Mojonation* system, have lost ground to other known P2P systems like BitTorrent and eMule [1]. One of the reasons of this failure was the limitation of the payment-based approach where all transactions performed in the system had to be cleared by a centralized entity.

Furthermore, when researchers designate a P2P system as successful, they usually relate the success to the system public acceptance and approval without providing detailed evidence supporting the evaluation. That is, there is no proof that these systems are successful because they have strong incentive mechanisms, robust against security attacks, or some other explanations. Moreover, some of these systems have not been thoroughly tested against security attacks for different reasons. First, there is a vague understanding of what strategies the attackers can possibly follow. Second, users might not attempt to subvert the incentive mechanism because they already get a satisfactory level of performance even though the incentive mechanism of the P2P system is vulnerable to attacks.

In this paper, we focus on the incentive aspects of P2P systems. We intend to analyze and investigate the behavior of users in two major P2P systems: BitTorrent and eMule.

Both of BitTorrent and eMule have incentive mechanisms embedded in their designs. For each of these systems, there is some utility function that represents the expected users' behavior in that system. That is, the incentives in that system yield perfect behaviors on the part of the users. However, if in practice users do not behave perfectly as expected, then it must be that their utility function does not reflect what it has been set for. So, another way of viewing our investigation is an attempt to reverse-engineer the *real* utility function by looking at users' behaviors under different sets of incentives.

We present the results of our large-scale measurement study of BitTorrent and eMule. Our paper is an analysis of how users recognize the incentives provided in both of BitTorrent and eMule. Moreover, we question how users respond to the incentives, and whether the incentives provided in both systems are sufficient enough to drive users towards the desired level of cooperation.

The rest of the paper is organized as follows: In Section 2 we provide a brief background on BitTorrent and eMule. Related work is in Section 3. In Section 4 we present our experimental analysis. Results and discussion are in Section 5, while in Section 6 we conclude our paper.

II. Background

A. BitTorrent

BitTorrent [6] is a content distribution system in which peers cooperate in downloading files. It relies on lowering the cost of sharing by redistributing the load from one peer to multiple peers. Each peer can be connected to several others to download parts of the file. However, although a peer might be connected to many other peers, it typically uploads only to a small number of them. A peer decides which users to serve based on the following three rules. First, a remote peer is *unchoked* based on his current upload rate – if he provides reciprocal service, he will be given service in return. Second, a peer is *optimistically unchoked* even if he does not provide reciprocal service, in an attempt to provoke him to cooperate. Third, peers that are currently unchoked but do not provide any useful services are periodically *choked*, assuming that they are uncooperative or do not have any useful parts of the file to offer (this is called *anti-snubbing*).

BitTorrent uses a central node called a *tracker* that assists peers who are joined in a torrent to communicate with each other. When initiating a download, each peer is required to contact the tracker in order to request the identities of other peers who are involved in the file distribution process. After joining the torrent, peers periodically contact the tracker to *refresh* the list of the identities they have. Finally, a *seeder* in BitTorrent is a peer who has all the pieces of the file, while a *leecher* is a peer who does not have the complete file.

B. eMule

A lightweight, *pair-wise* credit system is implemented in the eMule system [8]. The goal of the credit system is to reward users contributing to the network by reducing their waiting time in the upload queue. For each request in the upload queue the peer computes the *Queue Rank* based on a scoring function that depends on the current waiting time for the request, as well the upload and download volumes for the peer. The main advantage of this scheme is simplicity: there is no communication overhead and a peer only needs to maintain the upload and download volumes for each peer it has communicated with. The approach is cheat-proof in the sense that peers have no reason to tamper with the credit file. However, anecdotal evidence [18] suggests that the approach does not consistently provide a clear performance advantage to users who contribute resources to the network.

III. Related Work

BitTorrent's incentive techniques have been discussed in many publications. [9] argues that the tit-for-tat mechanism employed in BitTorrent has helped increase the cooperation level among peers. Similarly, [2] emphasizes that BitTorrent's reciprocity algorithm makes the system appealing to be used and that the design of BitTorrent increases cooperation among peers. The paper also suggests that in some cases and due to the large number of seeders, BitTorrent fails in reducing freeriding since there is no specific mechanism embarked to limit their gains. On the other hand, [5] suggests that the tit-for-tat mechanism is not efficient enough in deterring unfairness; and relates this inadequacy to the heterogeneity of peers' bandwidths. Moreover, [14] argues that BitTorrent lacks fairness: It does not punish freeriders effectively neither it does reward users who contribute properly.

The unchoke mechanism is also discussed in some of the literatures. [7], a paper by the creator of BitTorrent, presents the unchoke mechanism as the only efficient method used in BitTorrent to maximize peers' download rates. [16] claims that the optimistic unchoke mechanism fortifies the system's robustness by giving leechers chances to connect to other fast leechers or seeders. On the other hand, [15] evaluates the unchoke mechanism in BitTorrent and questions its efficiency in providing reasonable reciprocation in balancing upload and download rates. The paper concludes that the unchoke mechanism provides fairness to leechers in connecting to others and a reasonable level of reciprocation in general.

Other studies investigated BitTorrent's performance. [17] studied the system's performance characteristics such as availability, integrity, flashcrowds handling and download speeds. The paper also suggests some modifications on BitTorrent's architecture to provide incentives for peers who become seeders in order to stay connected for longer times.

[11] performed an extensive trace analysis on BitTorrent-like system. The study finds that the availability service in BitTorrent is poor and does not attain a satisfactory level. Furthermore, the study relates the existing problems, such as the fluctuating in downloading performance and exhibiting unfairness to peers, to the exponentially decreasing arrival rate of peers.

On the other hand, [13] traced a torrent for five-month period in which thousands of peers were involved. The paper concludes that BitTorrent is an effective system which allows peers to obtain high download rates comparing to other P2P systems, and that it reacts robustly towards flashcrowds. Likewise, [4] concludes similar outcomes, however, it traced larger number of contents that varied in size and popularity. Furthermore, it explored the side of files' disseminating and load balancing in BitTorrent.

On eMule side, and to the best of our knowledge, there is no direct detailed study that involves measuring or evaluating its performance. However, we found a workload measurement on eDonkey which is an eMule-like system. The measurement traced eDonkey's clients in order to get information about their shared contents. The distribution of the shared contents in that study shows that there is a degree of geographical and interest-based clustering in the eDonkey system. The paper also suggests that this type of clustering could be exploited in enhancing search mechanisms for such systems.

An incremental work on [10] is presented in [12]. The work was also conducted on eDonkey in order to evaluate the semantic relationships observed among peers. The paper questions how to capture and exploit the semantic relationships among peers without involving them directly.

IV. Methodology

In our BitTorrent data collection we utilized a modified BitTorrent client that is able to aggressively request new peers from the main tracker; and to connect to as many peers as possible in order to track their download progress and their connection time. The tool is developed using a robust C-language BitTorrent client called *ctorrent*. We modified the ctorrent client into *ctorrent-bigbro* to be able to only connect to the remote peers and monitor their progress without being involved in any download or upload activity, to avoid participating in the distribution of protected contents. The ctorrent-bigbro provided us with remote peers' download volumes and their connection times only. The remaining attribute, which is the uploaded volume, was calculated from the IP-ID field which is embedded within each packet sent by remote peers. Provided that BitTorrent is not necessarily the only software that is sending and receiving packets, in our measurements for the uploaded volumes by the remote peers, we only monitored the IP-ID field for the TCP ports that sent BitTorrent's data.

In our data collection process on eMule we used a client stripped from the original eMule client's GUI. The client is called *xmule-crawler*. Xmule-crawler is capable of collecting detailed information that is not provided by using other passive monitoring techniques. Such information include requests issued by the user, aborted requests, download progress and files being deleted or removed from the shared directory.

On the BitTorrent side, we traced around 1000 torrents, involving more than a hundred thousand peers. On eMule, we obtained a total population of about 3000 peers with around ten days worth of tracking for each peer.

V. Results and discussion

We analyze our data in an attempt to understand how users in BitTorrent and eMule systems perceive and respond to the incentives provided; and whether the incentives are strong enough to drive users to a satisfactory level of cooperation.

In BitTorrent, we investigate how users follow different strategies by monitoring their download and upload behaviors and in that fashion deducing their cooperation levels. The strategies are set by fine-tuning the parameters of the BitTorrent client. Some of these parameter changes could boost the download progress of users and advance their gain compared to others who are probably not aware of the existence of these options or their significance. Such behavior induces freeriding. Currently, there is no clear understanding about the magnitude of freeriding in BitTorrent. Most of the work on BitTorrent is likely to focus on the optimistic unchoke mechanism and particularly whether it is an effective technique to deter freeriding. Although the optimistic unchoke process is used to incite peers who do not provide reciprocal services to cooperate, we suspect that it also furnishes opportunities for peers to freeride.

Similarly, in eMule, we investigate how effective the pairwise credit mechanism is in motivating users to cooperate. In general, the credit mechanism in eMule implies that sharing is the sole factor affecting users' positions in the upload queues and therefore rewarding them by reducing their wait-times. In practice, users may not follow this basic strategy. We suspect that they may, instead, tend to develop their strategies by undertaking different sets of *sharing actions* based on what fits their interests in downloading and uploading.

For an eMule peer, the shared directory is the only accessible directory to the rest of the network in which the peer saves the content he wishes to share with others. Simply put, the shared directory is the place where an eMule client saves the ongoing downloads. If a peer does not want to share a content, he can simply remove it from the shared directory and place it in another inaccessible directory. Therefore, by constantly following changes in the sizes of the files in the shared directories, we can realize peers' cooperation levels. Moreover, by tracking how often users remove the completed files we shall be able to deduce their sharing strategies.

A. BitTorrent

As mentioned in Section 2, BitTorrent uses the unchoke mechanism to reciprocate services to nodes that cooperate and periodically choke nodes that choose not to cooperate.

It also employs the tit-for-tat strategy that is known to provide a strong incentive for cooperation.

After weeding out the uninformative samples from our data, we ended up with 700 samples that showed distinctive trends in users' download and upload speeds[1]. In each of these remaining samples, we isolated four different clusters of BitTorrent users. Each cluster is characterized by different level of cooperation exhibited by its members.

The first cluster we isolated includes users who profited from the system as much as they benefited it: The volume downloaded by those users almost equals the volume they uploaded. Contemplating in this behavior, it seems that the members of this cluster, who comprise the majority of users in our samples, did not follow any definite approach in downloading or uploading their files. Or in other words, they just simply ran the BitTorrent client on its default without changing any of its settings.

The second cluster we isolated in our samples contains peers who we classify as the major contributor leechers in the system. Their high cooperation level is clearly seen in the considerable volume they uploaded which is more than twice than the volume they downloaded. This behavior gives the impression that the members in this cluster were either altruistic or, as the members in the first cluster, did not follow specific strategy in their interaction with other peers. However, they are differentiated from the peers in the first cluster by their higher bandwidth capacities. Nonetheless, we suggest that members in clusters 1 and 2 are the honest peers who responded to the incentives provided since they exhibited medium to high levels of cooperation. However, it is not clear whether their reaction to the incentives was driven by altruism, carelessness, or by the lack of knowledge of the possibility in exploiting the resources of the system and freeriding.

The third cluster in our samples includes freeriders: Users who completed their downloads without contributing to the system. The maximum upload speed we observed from those users was less than 2 Kbps. On the other hand, they were fast enough downloading with speeds reached up to 300 Kbps. Moreover, as an expected selfish conduct, those peers left the system as soon as they completed their downloads.

The last group contains users who downloaded at least as twice as they uploaded. They are also characterized by disconnecting immediately after they completed their downloads. They remarkably benefited from the system as well. However, their limited contribution to the system makes them distinguishable from freeriders. This behavior is explicable if we assume that those peers are fairly acquainted to BitTorrent's mechanisms, or at least, aware of the parameters put available in most of BitTorrent's GUI clients, thus, allowing them to follow advanced sharing strategies compared to peers in clusters 1 and 2. Based on this classification, it is obvious that peers in cluster 2 are structuring an enjoyable downloading source to peers in this cluster and to freeriders.

[1]Because of the discrepancy of the IP-ID method that we used in measuring the uploaded volume by the remote peers, we ignored the samples in which the total uploaded volume did not match or was not close to the total downloaded volume by all peers.

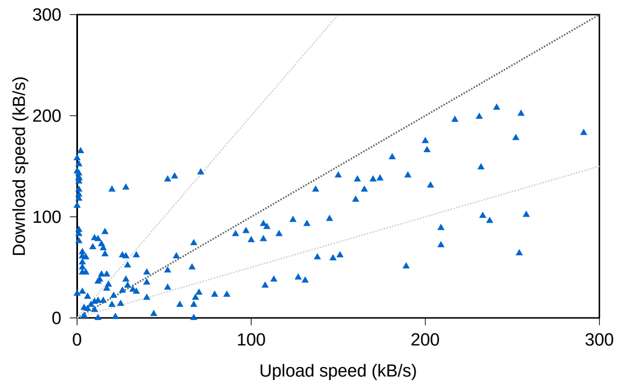

Fig. 1. Upload speed vs. download speed for a sample torrent.

In general, using the operational parameters in BitTorrent helps users to strategize themselves to augment their gain on the account of other users. However, we suspect that the use of such parameters counter effects the functionality of the optimistic unchoke mechanism. For example, a peer can easily change the maximum number of incoming connections that are allowed, which would increase the probability of having incoming connections that periodically unchoke optimistically. This presumably makes the optimistic unchoke mechanism in BitTorrent susceptible to parameters tweaking. Nevertheless, since the population of such peers is so far limited in BitTorrent, the system is still capable of performing significantly and is still an attractive choice for users. Yet, if the population of such peers significantly increased, we believe that the optimistic unchoke would be incompetent in handling such scenario.

On the other hand, freeriders' behavior is possibly related to either more advanced strategies compared to those adopted by peers in cluster 4, or to cheating. With some modifications to a BitTorrent client, users will be able to apply several threat scenarios that allow them to completely freeride without contributing to the system. Some threat scenarios are very well known now: Serving false pieces, connecting directly to seeders, and using multiple identities. Therefore, since the upload volumes of such peers is very limited, we are inclined to believe that this behavior is closer to cheating than parameters tweaking.

Fig. 1 demonstrates the distribution of peers joined in a torrent based on their download and upload speeds. The figure shows that freeriders are positioned on the download speed axis, users in cluster 4 are located between the download speed axis and the $y = 2x$ line, where the $x = 2y$ line separates users in clusters 1 and 2.

To get a comprehensive view of the clusters we identified above, fig. 2 illustrates the fraction of each cluster in each torrent in our samples. The figure shows that the majority of users belong to cluster 1 and cluster 2. Furthermore, we notice that at least 50% of users in all our samples uploaded

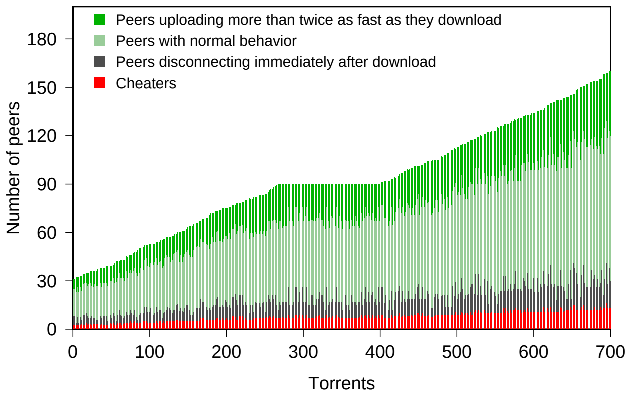

Fig. 2. Distribution of peers joined in 700 torrents.

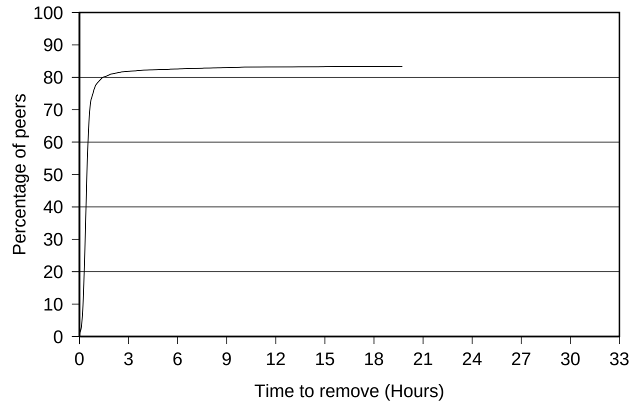

Fig. 3. CDF for the time to remove files from the shared directory.

as much as they downloaded, i.e., cluster 1; where users in cluster 2 who have been taken advantage from formed about 17%. On the other hand, cluster 4, users who disconnected immediately after they completed their downloads, comprised about 10% of the total number of users in a torrent, while freeriders population reached about 10%.

B. eMule

The credit mechanism in eMule rewards users who allocate resources for sharing by reducing their waiting time in the upload queue and thus expediting their downloads.

In general, users realize incentives in different ways and react to them according to their comprehensions. To obtain a better understanding on how users in our eMule samples respond to incentives, we categorized them into two clusters based on their sharing behavior.

The first cluster we isolated contains users who seem to perceive that sharing is the sole principal behind advancing their positions in the waiting queue. Therefore, they commit themselves to this primal action without undertaking any advanced strategy in sharing their contents. Their interaction with the system implies that they offer most of, if not all, their downloaded contents for sharing. This basic effortless behavior, which could be related to idleness, causes the shared directories of such users to grow huge. However, and to distinguish between the behavior of the users in this group and idleness, we monitored their activities during the collection of our data and we found that the majority of those users were active.

The second cluster we identified in our samples contains users who seem to recognize that there is no need to accumulate all the files they obtained in the shared-directories; or that there is a certain level of sharing which is sufficient enough for them to be rewarded. Thus, their behavior suggests that they advanced their sharing strategies to be based on keeping a minimum size of contents shared while removing other contents. Those minimum sizes of what they shared differ among them and probably depend on personal preferences or

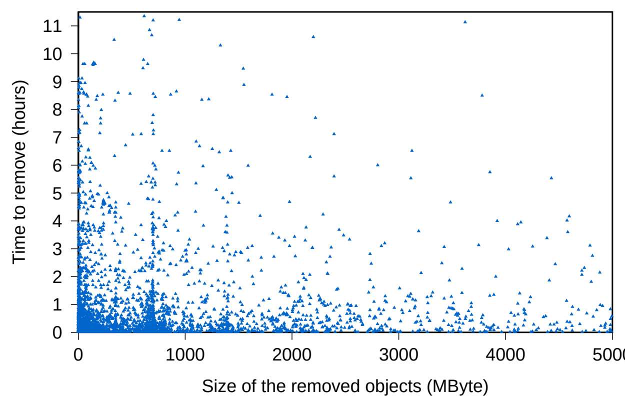

Fig. 4. Time to remove vs. size of the removed object(s) for sizes less than 5000 MBytes.

factors such as the popularities, the sizes or the categories of the contents.

Our data illustrate that the members of this cluster frequently removed the complete downloaded contents from the shared directories. Nevertheless, the lapse of time in which they removed those contents varies: Some of the users removed the contents immediately after they downloaded them, others waited longer.

The distribution of times between completing the download of contents and removing them from the shared directories gives us an insight on the volume of each cluster in our samples. Fig. 3 presents the cumulative distribution function for the times to remove the files[2]. It shows that about 18% of the users in our sample shared all the files they downloaded without removing any of them. On the other hand, 75% of the users removed their contents from the shared directories within one hour of completing the download.

The 18% of users mentioned above represents the first cluster we isolated: Peers who we assume that they did not

[2]In this distribution, each peer removed at least one file.

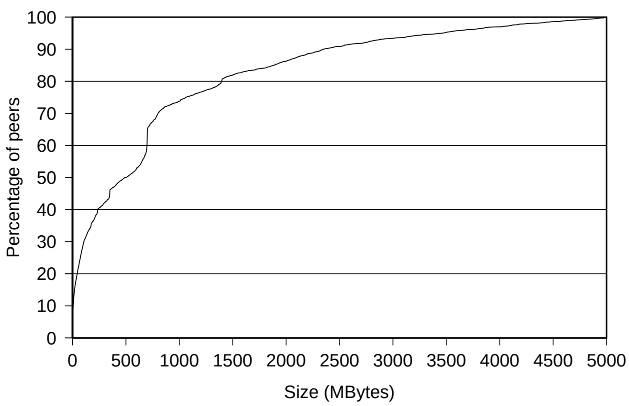

Fig. 5. CDF for sizes of the removed files excluding removed files with sizes more than 5000 Mbytes.

take any specific strategy in sharing their contents. During our sampling, some of those peers shared up to 100 GBytes of contents without removing any of them.

The 75% of users signifies members in the second cluster who frequently removed contents from the shared directories. However, users delete or remove files for several reasons not only for strategizing themselves in a download queue. Therefore, in an attempt to focus on sharing and strategy related motives, we assumed that any size reduction that is more than 5 GBytes is related to non-behavioral basis such as space constraint or disk cleanup.

Fig. 4 presents the distribution of the removed sizes that are less than 5 GBytes from the shared directories. The figure shows that the members of this cluster were more likely to remove small to midsized completed files. The cumulative distribution function of the sizes of the removed files, Fig. 5, demonstrates that 70% of those files ranged between 0 to 1400 Mbytes. Moreover, about 20% of them had sizes close to 700 Mbytes, which is usually a typical size of CD images, movies and other media files.

VI. Conclusion

In this paper we presented a measurement analysis study of BitTorrent and eMule P2P file sharing systems.

Our findings support that peers in BitTorrent might remarkably enjoy higher download speeds comparing to other P2P file-sharing systems. It appears that the incentive mechanisms adopted in BitTorrent are succeeding in promoting cooperation among peers until now. However, the level of cooperation in BitTorrent does not seem to be as satisfactory as expected.

Our results show that while the majority of peers in our samples contributed to the system as much as they benefited from it, 17% of the peers were the main contributors to the system, and more than 10% of peers downloaded as twice as they uploaded to others. Moreover, our results show that freeriding in BitTorrent is more widespread than it was previously assumed. The percentage of freeriders in our samples reached up to 10% of the total population joined in a torrent. This is an indication that BitTorrent may not be providing a strong enough incentive to reduce freeriding since there is no explicit mechanism in its design to punish or at least discourage freeridng.

On eMule side, it looks like peers comprehend the incentives differently. Some of the peers seem to understand that sharing is good. Therefore, they tend to share most of their downloaded contents to advance their positions in other peers' upload queues. In our samples, 18% of peers kept all the files they downloaded in their shared directories. On the other hand, another cluster of eMule peers seem to realize that they only need to share a limited amount of contents to expedite their downloads. Thus, they frequently removed the downloaded files from their shared directories. Our results show that 75% of users in our eMule samples removed their contents within one hour of completing their downloads.

References

[1] K. G. Anagnostakis and M. B. Greenwald. Exchange-based incentive mechanisms for peer-to-peer file sharing. In *Proc. 24th International Conference on Distributed Computing Systems (ICDCS 2004)*, March 2004.

[2] N. Andrade, M.Mowbray, A. Lima, G. Wagner, and M. Ripeanu. Influences on cooperation in bittorrent communities. In *3rd Workshop on Economics of P2P Systems (P2P Econ)*, August 2005.

[3] P. Antoniadis, C. Courcoubetis, and R. Mason. Comparing economic incentives in peer-to-peer networks. Technical report, Special Issue on Network Economics, Computer Networks, 46(1), 2004.

[4] A. Bellissimo, B. N. Levine, and P. Shenoy. Exploring the use of bittorrent as the basis for a large trace repository. Technical Report 04-41, University of Massachusetts Amherst, June 2004.

[5] A. R. Bharambe, C. Herley, and V. N. Padmanabhan. Analyzing and improving bittorrent performance. Technical Report MSR-TR-2005-03, Microsoft Research, Microsoft Corporation One Microsoft Way Redmond, WA 98052, USA, February 2005.

[6] BitTorrent Specifications Website. http://www.bittorrent.org/.

[7] B. Cohen. Incentives build robustness in BitTorrent. In *Proceedings of the 1st Workshop on Economics of Peer-to-Peer Systems*, June 2003.

[8] The EMule Project. http://www.emule-project.net/.

[9] M. Feldman and J. Chuang. Overcoming free-riding behavior in peer-to-peer systems. *ACM Sigecom Exchanges*, 6(1), July 2005.

[10] F. L. Fessant, S. Handurukande, A.-M. Kermarrec, and L. Massoulie. Clustering in peer-to-peer file sharing workloads. In *3rd International Workshop on Peer-to-peer systems (IPTPS 04)*, February 2004.

[11] L. Guo, S. Chen, Z. Xiao, E. Tan, X. Ding, and X. Zhang. A performance study of bittorrent-like peer-to-peer systems. In *IEEE Journal on Selected Areas in Communications (IEEE JSAC), first Quarter of 2007*.

[12] S. Handurukande, A.-M. Kermarrec, F. L. Fessant, and L. Massoulie. Exploiting semantic clustering in the edonkey p2p network. In *SIGOPS European Workshop*, pages 109–114, September 2004.

[13] M. Izal, G. Urvoy-Keller, E. Biersack, P. Felber, A. A. Hamra, and L. Garces-Erice. Dissecting bittorrent: Five months in a torrent's lifetime. In *Passive and Active Network Measurement 5th International Workshop (PAM 2004)*, April 2004.

[14] S. Jun and M. Ahamad. Incentives in bittorrent induce free riding. In *ACM SIGCOMM Workshop on Economics of Peer-to-Peer Systems (P2PECON)*, August 2005.

[15] A. Legout, G. Urvoy-Keller, and P. Michiardi. Rarest first and choke algorithms are enough. Technical Report inria-00001111, INRIA, Sophia Antipolis, September 2006.

[16] N. Liogkas, R. Nelson, E. Kohler, and L. Zhang. Exploiting bittorrent for fun (but not profit). In *The 5th International Workshop on Peer-to-Peer Systems (IPTPS 06)*, February 2006.

[17] J. A. Pouwelse, P. Garbacki, D. Epema, and H. Sips. A measurment study of the bittorrent peer-to-peer file-sharing system. Technical Report PDS-2004-003, April 2004.

[18] The eMule Project. Official eMule-Board. http://www.emule-project.net/board/index.php.