Journal of Network and Computer Applications 41 (2014) 47–55

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

Incentive mechanism for P2P file sharing based on social network and game theory

Tin-Yu Wu [b], Wei-Tsong Lee [a], Nadra Guizani [c,*], Tzu-Ming Wang [a]

[a] *Department of Electrical Engineering, TamKang University, Taiwan, R.O.C*
[b] *Department of Computer Science and Information Engineering, National Ilan University, Taiwan, R.O.C*
[c] *Purdue University, USA*

ARTICLE INFO

Available online 13 November 2013

Keywords:
Peer-to-peer (P2P) file sharing
Reward/penalty system
Social network
Game theory

ABSTRACT

According to statistics, most P2P applications use Gnutella or BitTorrent protocols to combat free riders. However, BitTorrent restrains free riders using the choking algorithm that chokes free riders only. Whereas, Gnutella identifies and prevents malicious nodes using the EigenTrust algorithm that has been proven to be imperfect. Therefore, both of these schemes are inefficient. According to a research conducted in 2005, 85% of Gnutella network users are free riders and only 1% of the users share new files and resources voluntarily. In this paper, by considering users' bandwidth, computing power and energy, our proposed system architecture gives users corresponding counters, which are stored and managed by the server. Moreover, the file-sharing model of our system can be divided into real-time streaming media sharing and file sharing. As for real-time streaming media sharing, users can use their counters to participate in the auction and bid on the admission of high-quality real-time streaming. As for file sharing, users have to pay a certain number of counters for every unit of download bandwidth. That is to say that all system users must use the counters to bid on or purchase services, which further enables users' spontaneous resource sharing.

1. Introduction

Recently, P2P (Peer-to-peer) technology has been extensively applied to the fields of file sharing, real-time streaming media sharing and communication. Also, because of the advancement of networking devices all over the world and the development of developing countries, P2P gradually becomes one important technology over the Internet. Up to now, many famous P2P-based file sharing platforms have been presented, like BitTorrent (Obele et al., 2009) and Gnutella (Rozario et al., 2011). However, free-riding is a well-known problem in P2P networks. Free riders in a P2P network refer to peers who keep downloading resources but are reluctant to contribute resources in return. Owing to such a free-riding phenomenon, the system performance might be degraded, those who have contributed resources to the network spontaneously cannot get worthwhile files, and more and more peers stop sharing resources. Therefore, a novel resource-sharing mechanism in P2P networks is necessary. A study conducted in 2005 showed that 85% of all Gnutella users are free riders and only 1% of the users spontaneously share new files and resources (Hughes et al., 2005).

In our proposed system architecture, we give corresponding counters to users according to their bandwidth, computing power and energy. The counters will be stored and managed by the server and users can use the counters for strategic decision-making: to bid on or purchase services. Also, based on social network and game theory, we propose a novel incentive mechanism that stimulates users to contribute resources spontaneously. Because resources in P2P systems is as if public resources that everyone can access, most users are reluctant to share resources and the system performance might be degraded. Therefore, based on the social network platform, this paper designs a Novel Incentive Mechanism (NIM), that encourages peers to share resources and excludes free riders by the relationship between peers. Finally, with the aim to combat free riders effectively and achieve fair resource allocation, we use game theory to analyze users' decision-making and corresponding rewards in different situations (Wu and Chan, 2010; Wu et al., 2011).

The rest of this paper is structured as follows: Section 2 outlines the background and introduces P2P platforms, including BitTorrent, Gnutella and Private tracker. Also, social networking habits and features of users, basic concepts and applications of game theory, and the model of Pareto Efficiency are stated. Section 3 describes our proposed social-network-based NIM and the system architecture. Simulation results and performance analysis are given in Section 4, which introduces the simulated scenarios and analyzes the weighted values and variations of the system

* Corresponding author.
E-mail addresses: tyw@niu.edu.tw (T.-Y. Wu), wtlee@mail.tku.edu.tw (W.-T. Lee), nguizani@purdue.edu (N. Guizani), roy11226@gmail.com (T.-M. Wang).

1084-8045/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.jnca.2013.10.006

parameters in different conditions. By using game theory, we can define users' strategy space and relative rewards in different situations and find out users' Nash equilibrium to prove our system performance and the efficiency in combatting free riders. Section 5 concludes this paper and gives objectives for the future.

2. Background and related works

2.1. P2P file sharing architecture

Different from the general centralized system architecture that relies on the computing power and bandwidth of the central core server, P2P technology depends on the computing power and bandwidth of system users to avoid the network congestion caused by the increased user traffic that servers cannot handle immediately in the centralized architecture. P2P allows peers to connect with each others for file sharing and users can connect with several nodes simultaneously to access files. P2P provides high data rates and high scalability because all P2P users can contribute hardware and bandwidth resources to the system. Whenever a peer node joins a P2P system and requests data from other nodes, the capacity of the entire network increases as well. To replicate data over peers, P2P architecture increases robustness because there will be no single point of failure in pure P2P systems (Kim et al., 2009; Wang et al., 2012).

In recent years, P2P technology has been widely applied to various file-sharing and communication programs, like BitTorrent, Gnutella and Skype. According to the presence of a central server, P2P systems can be classified into pure peer-to-peer systems, centralized peer-to-peer systems and hybrid peer-to-peer systems. Depending on the network topology, we can classify the existing P2P architecture into structured, unstructured and decentralized networks. Detailed description of BitTorrent architecture, Gnutella architecture and Private tracker architecture (Chen et al., 2010) will be given below.

2.1.1. BitTorrent

As the most popular centralized P2P architecture for distributing large amounts of data over the Internet, BitTorrent (BT) includes a central server to supply system operation. In BT systems, all peers must install BT software first. For file-sharing, a peer converts a file to a torrent file, which contains metadata, encryption and authentication of the file. Saved by the tracker, the torrent file is recorded in a tracker list that is kept in a global registry of all the uploaders and downloaders. To download the file, peers have to connect to the tracker to obtain the torrent file. Also, with the central tracker, BT allows peers that are interested in the same file to know from which other peers to download the pieces of the file. After being responded by the tracker with a list of peers having the requested file, users are able to download the file. Each tracker can supervise several downloads of multiple files simultaneously. At the same time, the peers have to announce themselves to the tracker every 30 min. Users are authenticated by the tracker while first joining the system. Finally, after getting the torrent file from the neighboring peers, the peers start to download the pieces of the file, each of which is approximately 256 KB. When a certain chunk is completely downloaded, the peer can share the file chunk. Peers that have the full original files are called seeds. With the expansion of BT usage, the Kademlia DHT has been added to BT as a supplement of the central tracker to avoid the single-point-of-failure of a central tracker (Qi et al., 2008).

Distributed Hash Table (DHT), is a distributed system and Kademlia DHT refers to a distributed hash table for decentralized P2P networks. Kademlia specifies not only the architecture of the network, but also the exchange of data among nodes. Even in the worst case, it takes only n steps to find out the target node. By using Kademlia DHT, BT can operate without the tracker: each node is responsible for routing in a small range and storing some data. Consequently, users who cannot connect to the tracker can find out the nodes interested in the same file through DHT network and start to download the file. According to Qi et al. (2008), DHT enhances the performance of BT 5 times better.

To restrain free-rider phenomenon, BT uses a choking algorithm to control the connections. In BT systems, a client must be unchoked by the target peer first before downloading data. Only when the unchoke notification is answered can the client begin downloading the fragments of a file. Every BT client unchokes a certain number of its peers and whether to use a tit-for-tat algorithm to determine who to unchoke depends on whether the peers are seeds or not. If yes, the client chooses to unchoke the peer that contributes most resources to accelerate its download and make it acting as a seed node to enhance the system performance (Mehyar et al., 2007). If not, the client uses the tit-for-tat policy but whether it is worthwhile to unchoke the peer must be considered in advance. Supposing a client fails to download a requested data chunk from a specific peer within a period of time, BT assumes that the client is snubbed by the peer and will not answer its unchoke notification hereafter. Before optimistic unchoking finds out another peer having a better downloading rate, the client still can get a minimum downloading rate from contributors, which induces free-riding.

2.1.2. Gnutella

Gnutella is a protocol for fully decentralized P2P file-sharing systems, with an estimated market share of more than 40%. Without any central servers, the Gnutella network continues to exist as long as there are at least two clients. To this day, Gnutella, more than a piece of software, refers to an open protocol.

We briefly explain how Gnutella works. To join the network, the clients must first execute a bootstrapping function to discover other peers in the network. A new client can use different methods of bootstrapping: a pre-existing address list of other peers in the network, updated web caches of known nodes, or UDP host caches. After answered by a list of working addresses, the client tries to connect to the peers in the network. Since version 0.6 of the Gnutella protocol, peers are divided into ultra nodes (ultrapeers), each of which can connect to 32 leaf nodes, and leaf nodes, each of which can connect to approximately 3 ultrapeers only. However, most peers do not stay in the network for long and their requests for files are often discarded during the transfer, which easily leads to network congestion.

As observed in Portmann et al. (2001), the number of nodes in the Gnutella networks affects the system performance directly. Because Gnutella is a pure P2P network, when there are more than a few thousand nodes in the network, the cost of peer discovery and searching increases super-linearly and even the required bandwidth will exceed the resources of a typical client. Therefore, to enhance the performance of the Gnutella network, most applications using the Gnutella protocol restrict the size of the network to a few thousand nodes, which generates a power-law topology in file-sharing applications to share files having the same nature. Currently, owing to a power-law topology, the Gnutella network cannot get beyond a network size of a few thousand nodes and thus loses its diversity of files and global search capabilities. In the Gnutella networks, a client that has nothing to share still can request for downloading files. Thus, EigenTrust algorithm in Gnutella cannot solve the free-rider problems efficiently.