Performance of Peer-Assisted File Distribution

Cristina Carbunaru
National University of Singapore, Singapore

Yong Meng Teo
National University of Singapore, Singapore

INTRODUCTION

The Internet is becoming a pervasive medium for file distribution, and growing number of users and file sizes introduce new performance challenges in the efficient distribution of files. Global Internet traffic has increased by eightfold in the last five years and is estimated to increase by another fourfold over the next five years (Cisco, 2010). File sizes have grown from several tens of megabytes, such as music files, to several gigabytes, for movies today. Traditionally, file distribution is based on the simple *client-server approach* that has limited scalability (Das, et al., 2006). Recently, *peer-assisted approach* has been proposed (Cohen, 2003; Kazza). In this approach, each peer (client) downloads a file from the server and at the same time disseminates the file to other to improve download performance.

Previous work on performance analysis of peer-assisted systems focuses on *steady-state* (Guo, et al., 2007; Qiu & Srikant, 2004). This is reasonable in file-sharing because peers remain in the system to continue sharing the file after completion of download. In contrast, peers in file distribution systems download the file as fast as possible and leave the system. Furthermore, the performance of peer-assisted file distribution is affected by *flash crowds.* In a typical file distribution system, peer arrival rate over time is characterized by two main phases as shown in Figure 1. When a new file is introduced, peer arrival surges causing flash crowd but as file popularity drops, the system goes into steady-state. Flash crowds increase user download time and thus have serious consequences on business revenue. For example, Amazon estimates that for every 100 ms delay caused by flash crowd, sales decrease by 1%, and Google reports that for each half a second increase in waiting time results in a 20% decrease in search traffic revenue (Zhang, et al., 2011). Zhang et al. (2011) show that flash crowds affect download performance for up to 45% of the users in BitTorrent (BT) (Cohen, 2003) systems. Flash crowds demand higher server bandwidth, and introduce performance challenge in provisioning the servers to deliver acceptable quality of service.

This article focuses on recent approaches in performance analysis of peer-assisted file distribution, new results for flash crowds and emerging research challenges. Firstly, basic concepts in file distribution and related works are introduced. Secondly, the article presents new insights in flash crowd performance using both measurement and analytical model, and discusses performance challenges for users, service providers and protocol designers. Lastly, this article highlights issues for achieving efficient file distribution in cellular networks where peer download capacities is lower and connectivity is less reliable.

BACKGROUND

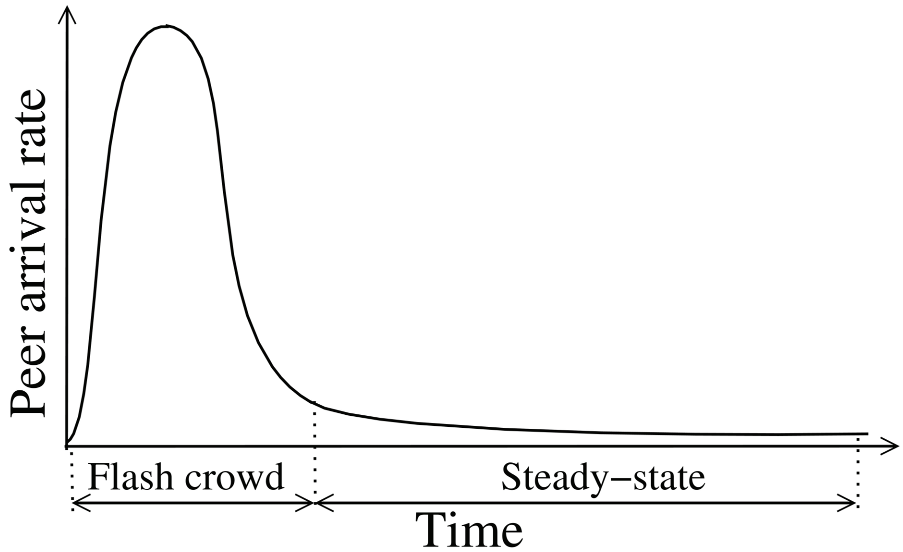
This section reviews file distribution protocols with different incentive schemes, and performance studies of peer-assisted systems at steady-state and with flash crowds.

Peer-Assisted File Distribution Protocols

Peer-assisted file distribution schemes, based on peer-to-peer (p2p) protocols, are designed to scale and self-organize without a central server, and in the presence of a highly transient population of nodes, network, and computer failures. P2p systems have various degrees

DOI: 10.4018/978-1-4666-5888-2.ch457

Figure 1. Evolution of peer arrival rate in swarms



of centralization including completely decentralized systems such as Gnutella (Gnutella) to partially centralized systems such as Kazaa (Kazza) and BitTorrent (Cohen, 2003). File-sharing and file distribution are among the most important applications of peer-to-peer (p2p) protocols. Some examples are: Napster, Gnutella, Kazaa (Kazza), BitTorrent (Cohen, 2003), Avalanche (Gkantsidis & Rodriguez, 2005), Antfarm (Peterson & Sirer, 2009), FairTorrent (Sherman, et al., 2009). Considering the importance of p2p protocols, it is challenging to understand and predict the expected peer performance. Peers in these systems decide how and when to contribute to system's service based on different incentive schemes.

BitTorrent (BT) is one of the most popular p2p protocols used for file-sharing and file distribution. A file is divided into chunks, called *blocks*, and multiple blocks form a piece. Peers in a BT swarm cooperate to download large files, initially only available on a few nodes that are called *seeds*. A new peer connects to a tracker to obtain a *torrent* file, containing a list of active peers and their list of blocks. Peers simultaneously download and upload different parts of a file from other peers, as well as directly from the seeds. After the download is completed, BT peers can decide to stay in the swarm and become seeds, or leave the system. An incentive scheme called *choke/unchoke* regulates the exchange of blocks among peers, where each node attempts to upload blocks to the peers that offered the best download rates during the last download interval. The unchoked peers (that receive blocks from a peer) are chosen based on the best download rates, while one unchoke, called an *optimistic unchoke*, is randomly chosen from the remaining requests the peer received. A new BT torrent needs at least one seed, the peer that wants to share the file, to start the download. A peer requests blocks from other peers based on *rarest-first policy*.

Many studies (Fan, et al. 2006, 2009; Izal, et al., 2004; Liao, et al., 2007; Qiu & Srikant, 2004; Yang & Veciana, 2006) have been conducted to analyze the performance of BitTorrent-like systems, which are based on a choke/unchoke incentive scheme. These studies analyze the system as a whole using the metric of effectiveness of file-sharing to characterize the incentive mechanism. Some of the studies proposed new incentive schemes (Liao, et al., 2007; Peterson & Sirer, 2009; Sherman, et al., 2009) that improve upon performance of BT. Considering the different degrees of centralization in p2p incentive schemes, server bandwidth provisioning is a challenge. For example, the impact of server capacity on the performance of peer-assisted systems has been studied using fluid models (Das, et al., 2006; Sun, et al., 2009). Various methods for server bandwidth allocation among different swarms and peers have been shown to improve performance for content distribution (Sun, et al., 2009). Other proposed methods, such as content and dynamic