Offloading Servers with Collaborative Video on Demand

Paweł Garbacki, Dick H.J. Epema and Johan Pouwelse
Delft University of Technology
{*p.j.garbacki,d.h.j.epema,j.a.pouwelse*}@*tudelft.nl*

Maarten van Steen
Vrije Universiteit Amsterdam
steen@cs.vu.nl

Abstract

*The peer-to-peer (*P2P*) paradigm provides a data distribution model that may be attractive for Video on Demand (VoD) as it allows to decrease the costs and to increase the scalability of video distribution. However, VoD is more challenging for* P2P *technology than file sharing or live streaming, and so, practically feasible VoD systems proposed to date rely on a backend server infrastructure as a fail-over solution. In this paper we investigate how the dependency on servers can be decreased by optimizing the video piece-selection strategy and by allowing multiple peers to form a collaboration for obtaining a single video. We prove analytically that the amount of bandwidth provided by the servers to guarantee a certain video bitrate can be reduced to arbitrarily low values by increasing the collaboration size if only the number of peers in the system is sufficiently large. In a set of simulations of a trace-based system model we show that for systems as YouTube the proposed optimizations would result in saving of as much as 70% of the server bandwidth.*

1. Introduction

With the increase of the link capacity offered to Internet users, Video on Demand (VoD) services are rapidly gaining popularity. Services such as YouTube [1] that allow their users to post video files on-line are visited by millions of people on a daily basis. Providing VoD to a large population of users requires a significant amount of bandwidth, which effectively becomes the scalability bottleneck of VoD infrastructures. For instance, the bandwidth provisioning costs of YouTube servers are estimated at $6M per month [1].

The peer-to-peer (P2P) resource sharing model provides an attractive architectural solution for bandwidth-limited applications. Peers employ their upload bandwidth to redistribute the downloaded content, decreasing the dependency and the load on the servers [7]. The content redistribution capability of a P2P network is, however, conditioned on the willingness of the peers to contribute their bandwidth. Relying on the altruism of the users eager to donate their bandwidth does not suffice to guarantee service of high quality [10], and so, economically rational incentives are needed to stimulate bandwidth contributions of the peers.

The most feasible in practice incentive mechanisms proposed up to date for file sharing [4] and live streaming [11, 8] P2P networks establish bartering relationships between peers that exchange data pieces. In this paper we investigate the applicability of bartering incentives to VoD systems. We measure the efficiency of bartering in a particular P2P system in terms of the system entropy which quantifies the probability of establishing a bartering relationship between two randomly selected peers. In a VoD system the entropy and thus also the bartering possibilities are negatively affected by the fact that peers at different playback positions are interested in different pieces of the video file. Addressing this problem we propose a *biased random* piece selection strategy which optimizes the order in which pieces are downloaded by a VoD application.

The results of performed analytical study suggest that the piece selection strategy affects the number of bartering possibilities in a VoD system only to a certain extent. A further improvement in the number of bartering possibilities can be achieved only by decreasing the playback bitrate or by increasing the bandwidth available for a peer. Since the playback rate determines the video quality, only the improvements in the amount of available bandwidth are reasonable. To this end, we propose a *collaborative VoD* protocol that increases the bandwidth available for a peer by using the idle bandwidth of multiple peers collaborating in obtaining a single video file rather than requesting the bandwidth from servers.

In an analytical study we investigate the impact of the peer bandwidth capacities, video playback bitrate, the number of video pieces, and the collaboration size on the server bandwidth consumption. The study indicates that the amount of server bandwidth spared by the collaborative VoD protocol increases rapidly with the collaboration

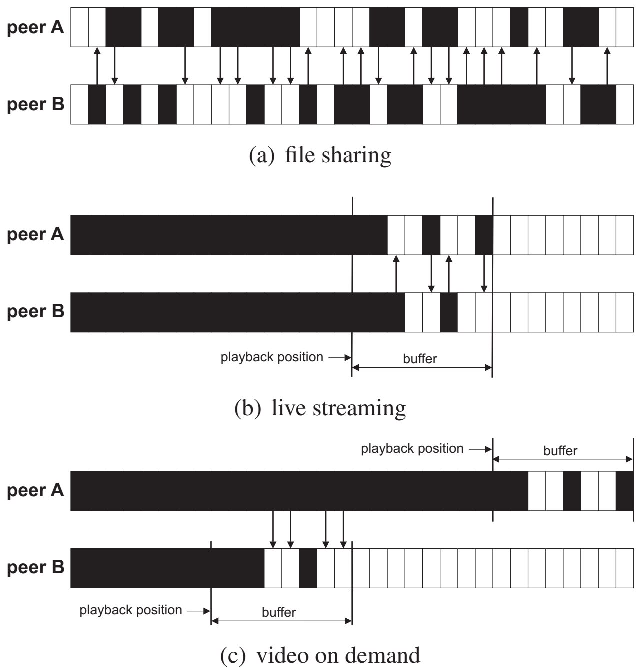

Figure 1. Piece exchange possibilities between two peers for different types of P2P applications. Black and white rectangles represent pieces already obtained by a peer and still missing pieces, respectively. An arrow indicates a possible transfer of a piece between peers. Piece exchange is possible only if there is at least one arrow from peer A to peer B and at least one arrow in the opposite direction.

size. The conclusions of the formal analysis are confirmed in a series of simulations using traces of the YouTube community. The results of the simulations suggest that the biased random piece strategy and the collaborative VoD protocol could reduce the server bandwidth consumption by more than 70%.

2. P2P VoD with incentives

Each byte of video content served by a peer saves one byte of server bandwidth. Assuming rational behavior [10], peers are willing to contribute their bandwidth only when clear incentives to do so are provided. The incentive models in P2P networks that are arguably the most feasible in practice, establish *bartering relationships* [4] between pairs of peers. Data transfer between bartering peers is possible only on an exchange basis. More precisely, peer A can obtain a piece of data from peer B only if peer A can give peer B some other piece in return.

2.1. Entropy as a measure of bartering efficiency

The number of bartering possibilities can be expressed in terms of the system *entropy*, which is defined as the probability that two randomly selected peers do not have any pieces of data to exchange. For instance, if all peers have the same set of pieces, no exchange can occur, and the entropy is maximal (equal to 1). If pieces are distributed randomly across all peers, the entropy is low. P2P protocols based on the bartering concept can improve the efficiency of data dissemination by optimizing (decreasing) the entropy. However, the possibility of decreasing the entropy depends on the properties of the data dissemination protocol. We now discuss the entropy in each of the three models of video distribution.

Entropy in offline file-sharing systems. It is easy to achieve a low entropy in file-sharing systems where the data pieces can be downloaded in a random order (see Figure 1(a)). Assuming that every piece has the same chance of being selected for download and that pieces have the same number of replicas in the system, we can compute the entropy in a file sharing system following a reasoning similar to one introduced in [9]. The entropy in a file sharing system can be found from Eq.(7) in [9] as being roughly equal to $\ln N/N$, where N is the number of file pieces.

Entropy in live streaming systems. Live streaming is similar to file sharing in the sense that at a given time, all peers are interested in the same content pieces (see Figure 1(b)). Streaming peers usually start the playback with a certain delay, which allows peers to buffer pieces ahead of the initial playback position. Pieces in the buffers of different peers can be exchanged in a bartering fashion. As shown in [11], if all peers have buffers of the same size N, then the entropy in a live streaming system can be approximated by $1/N$.

Entropy in VoD systems. In VoD systems the buffer overlap may be not sufficient to establish a bartering relationship between peers (see Figure 1(c)). In particular, if the size of the buffer is smaller than the distance between the playback positions of any two peers, the entropy equals 1, which means that bartering is not possible at all.

The entropy in a VoD system is directly correlated with the amount of server bandwidth required to guarantee a video playback with a low data loss. If each peer is bartering for video fragments with k other peers at a time then the probability that none of those peers has any data to exchange equals E^k, where E denotes the system entropy. Hence the fraction of the bandwidth coming from the servers in a VoD system equals to E^k. The conducted measurements of the YouTube community, described in more detail in Section 5, indicate that the number of users watching the same video at a given point in time is small, even in a flashcrowd [2] on average equal to 5. Although this issue is vastly ignored by other VoD P2P systems, any protocol assuming that the number of bartering partners k can be arbitrary large is unrealistic. Reduction in the server bandwidth consumption can be thus achieved only by decreasing the value of the entropy.

A reduction of the entropy can be accomplished by optimizing the strategy determining the order in which pieces are downloaded and by increasing the amount of bandwidth available in the system. In this paper we propose mechanisms facilitating improvements on those two fields. Before presenting those mechanisms we introduce a model of a VoD system which we use in the analysis of the presented mechanisms.

2.2. System model

For the purpose of the analysis we assume that all peers have the same upload and download bandwidth capacities denoted μ and c, respectively. The video playback rate is denoted by s. The values of μ, c and s are expressed in units representing the number of video pieces transferred per second. This way we avoid introducing a parameter defining the size of a piece. We assume that $\mu \leq c$ and $s \leq c$.

It seems natural to assume that each peer maintains a buffer of pieces directly after the playback position. The size of this buffer is negligible compared to the length of the video. We integrate bartering incentives into our model by assuming that a peer can download data at the rate not higher than the upload link capacity, which imposes the constraint $s \leq \mu$. While selecting a piece to download, a peer chooses a piece in the buffer with probability s/μ and a piece beyond the buffer with probability $1 - s/\mu$. Under the assumption that a peer receives data at a rate equal to its upload link capacity, the selected probabilities guarantee the buffer filling rate to be equal to the playback rate.

The scope of the analysis is limited to the set of peers playing a single video file. We denote by N the number of video file pieces. We assume a uniform distribution of the peer playback positions over the video length, so the probability that a randomly selected peer has i pieces equals $1/N$ regardless of i. We consider the least altruistic scenario where a peer that has downloaded all video pieces refuses to upload any more pieces to other peers in the system.

In addition to the peers, the system contains a number of *servers*, i.e., content injectors that possess the entire video file and serve it to the peers without asking any data in return. The bandwidth at the servers is a scarce resource and its consumption should be minimized while making sure that there is enough server bandwidth available to guarantee close-to-zero data loss (which means a piece arrives too late for playback, or not at all). Peers are competing for the server bandwidth. Ideally the bandwidth allocation algorithm at the server should treat all peers evenly giving each peer access to the same amount of server bandwidth.

3. Piece selection strategy for VoD

A *piece selection strategy* determines the next video piece selected for download by the peer downloading VoD and its helpers. Obviously, the next piece to be downloaded by a peer has to be selected from among the pieces that are possessed by at least one of the bartering partners of the peer. A piece selection strategy is a function that computes the piece number based on the information available locally at the peer.

3.1. The biased random strategy

An obvious candidate for a piece selection strategy is to select first pieces closest to the current playback position. We will further refer to this strategy as *earliest first*. The earliest first strategy leads, however, to a strong bias in the number of piece replicas in the system. Namely, pieces with small numbers are highly replicated while pieces close to the end of the video are possessed by only a few peers. Consequently, earliest first leads to a bottleneck in obtaining the tail pieces of the video file.

Another possibility for piece selection is the *rarest first* strategy adopted directly from file sharing networks [4]. Rarest first increases the entropy in the system, but it also results in an effect opposite to the one produced by the earliest first strategy. Namely, a peer using the rarest first strategy will concentrate on pieces closer to the end of the video file ignoring the pieces closer to the playback position. To better understand that phenomenon, let's observe that the number of replicas of a piece depends on the position of that piece in the video file with earlier pieces having more replicas. The rarest first strategy will try to balance the number of replicas across the pieces by requesting pieces closer to the end of the video file. Pieces immediately following the playback position are disregarded which affects the playback reliability and smoothness.

We propose the *biased random* strategy that optimizes for the entropy by taking into account piece rarity but at the same time not excluding for selection pieces close to the playback position. According to the biased random strategy, each peer selecting the next piece to prefetch, chooses a piece randomly with a probability that is inversely proportional to the number of replicas of that piece. The number of piece replicas is computed by each peer from the locally collected information about the pieces possessed by other peers. The probabilities of selecting individual pieces are normalized across the set of pieces available to download for a peer to guarantee that the peer will always select one of the pieces. More formally, if $\{i_1, i_2, \ldots, i_k\}$ is the set of numbers of the pieces that a peer could prefetch and $r(i_1), r(i_2), \ldots, r(i_k)$ are the numbers of piece replicas as discovered by the peer, then the peer will select piece i_l with probability $r(i_l)^{-1} / \sum_{m=1}^{k} r(i_m)^{-1}$.

Note that according to the biased random piece selection strategy, pieces with fewer replicas will be selected more frequently. At the same time, the introduced nondeterminism gives any piece, so also pieces directly after the playback position, a chance of being selected, even if this piece has more replicas than some other pieces.

A possible extension of the biased random strategy is to include in the piece selection probability the distance between the piece and the current playback position, giving priority to closer pieces. However, piece rarity and the distance from the current playback position are correlated in the sense that pieces further away usually have fewer replicas. Therefore, optimizing rarity will have a negative impact on distance from the playback position and vice versa — minimizing distance will negatively affect the rarity. Striking a balance between these two conflicting objectives would require a piece selection probability function much more complex than the one that we propose here. In this paper we concentrate on the basic biased random function which is more intuitive and easier to analyze. In future work we will address possible improvements to this basic function.

3.2. The entropy in a VoD system

Having defined a piece selection strategy, we can now compute the value of the entropy in a VoD system. Due to space limitations we present here only the key results of the elaborate analytical study which is included in an extended version of this paper [5].

Assuming the system model introduced in Section 2.2,

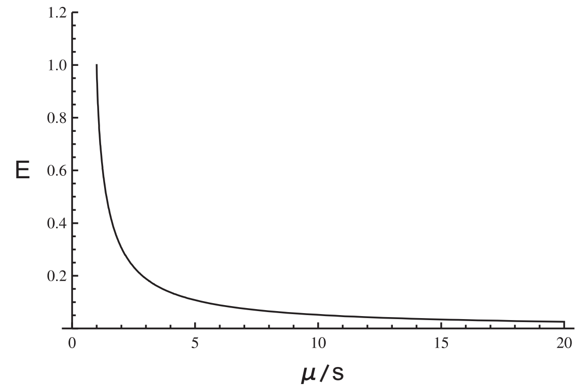

Figure 2. The entropy as a function of the ratio of the uplink capacity and the playback rate.

for large enough values of N, the entropy E in a VoD system employing the biased random piece selection strategy can be estimated as

$$E = 1 - \left(\frac{\mu}{s} - 1\right) \ln \frac{\frac{\mu}{s}}{\frac{\mu}{s} - 1} + O\left(\frac{lnN}{N}\right). \quad (1)$$

The component $O(lnN/N)$ of Eq. (1) encapsulates the probability that the peer cannot obtain a piece from outside of its buffer. Note that this probability exhibits a similar trend as the entropy in file sharing P2P systems (see Section 2.1), which is intuitive as pieces from outside of the buffer are exchanged in a fashion similar to piece exchange in file sharing systems.

Note that contrary to file sharing and live streaming systems, the entropy in a VoD system cannot be reduced to an arbitrarily low value by increasing the number of pieces N into which the (video) file is divided. For large values of N, the last term in Eq. (1) is small, and the value of the entropy is determined by the ratio μ/s. Figure 2 presents the decreasing trend of the entropy value as the ratio μ/s increases ignoring the component $O(\ln N/N)$ in Eq. (1). Note that the entropy converges to 1 when s is close to μ. This is intuitive as a peer that plays the video at its download rate does not have any bandwidth to spend on obtaining pieces ahead of its playback position, which would result in more piece selection options.

Obviously, the entropy can be decreased by reducing the playback rate s, which would have a direct impact on the video quality. In the next section we propose a protocol that increases the amount of upload bandwidth μ available for a peer, resulting in a decrease of the entropy while preserving the current playback rate.

4. Collaborative video on demand

In this section we introduce a protocol that supplements the VoD system with bandwidth shared by idle peers, effectively decreasing the entropy without sacrificing the video playback quality.

4.1. Idle bandwidth sharing

In our previous research [6] we have shown that the performance of file sharing P2P networks can be significantly improved by allowing peers to form *collaborations* with idle peers having excess bandwidth, the so called helpers. Formally, a *helper* is a peer that is not directly interested in the content it is downloading but that employs its idle bandwidth to fetch content pieces for a peer requesting the content. Helpers forming a collaboration with a peer downloading data act on behalf of that peer and use their bandwidth to barter with peers in other collaborations. Helpers may be attached exclusively to a single peer and download pieces that are not present at that peer [6], or they can act as microseeds and be shared by all interested peers in the system [12]. In this paper we assume the former model in which a helper acts on behalf of a single peer at a time.

VoD systems open a new area of application for the collaborative bandwidth sharing concept. VoD imposes stricter service quality requirements than file sharing as each video fragment has to be obtained before the playback reaches its position. The high instability of P2P architectures caused by their dynamics has a negative impact on the probability that a piece will be obtained from the P2P network on time. This probability obviously depends on the amount of bandwidth available for a peer to download its pieces, which in turn is a direct consequence of the number of helpers.

4.2. The impact of helpers on the entropy

Each additional helper increases the total upload bandwidth capacity of a collaboration, which is defined as the aggregate upload bandwidth of all peers in the collaboration that can be used for bartering with peers in other collaborations. We denote the upload bandwidth capacity of a collaboration by μ_h, where h is the number of helpers in the collaboration. Of course, a helper has to divide its upload bandwidth between obtaining data from other peers (by bartering) and forwarding the downloaded data to the peer playing the video. A helper cannot send data to the peer playing the video faster than it is downloading the data (so, in particular, not faster than half of its upload link capacity) and the peer playing the video cannot receive data faster than its download bandwidth c. This gives us the following formula for μ_h:

$$\mu_h = \mu + h\mu - min(c, h\frac{\mu}{2}), \qquad (2)$$

where h is the number of helpers.

Replacing μ with μ_h in Eq. (1) gives us the following formula for the value of the entropy in a collaborative VoD system when each peer uses h helpers

$$E_h = 1 - \left(\frac{\mu_h}{s} - 1\right) \ln \frac{\frac{\mu_h}{s}}{\frac{\mu_h}{s} - 1} + O\left(\frac{lnN}{N}\right). \qquad (3)$$

Since the value of μ_h increases linearly with h for h large enough (when $h\mu/2 > c$, or when $h > 2c/\mu$), the shape of the entropy as a function of h is similar to the shape presented in Figure 2.

5. Experimental evaluation

We assess the impact of the optimizations proposed in this paper on the server bandwidth consumption in a series of simulations. Before presenting the results of the simulations we discuss the experimental setup.

5.1. Experimental setup

For the purpose of the simulations we have crawled the YouTube site collecting statistics about over almost 1.4 million randomly selected videos. The statistics contain the video duration, date and time when the video was added, and the total number of views. We simulate distribution of a single video file with a running time qual to an average duration of a YouTube video which is 265 seconds.

The collected YouTube statistics do not include the exact times when each video has been viewed. Since the content popularity in on-line communities usually follows a flashcrowd pattern, we use a flashcrowd model proposed in [2] to generate peer arrivals. The average number of concurrent views computed from the statistics by dividing the total number of views by the video age results in a flashcrowd model that peaks at 8 views, exhibits an average of 5 views during the flashcrowd and 0.4 views outside of the flashcrowd. The peer bandwidth model uses uniform values for all peers in the system with 1500 kbps download and 384 kbps upload link capacity. The specific link capacity values describe the most common Internet connection type of a P2P network user [3]. The number

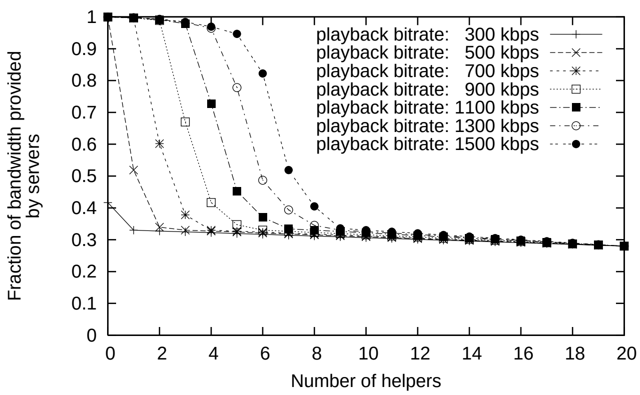

Figure 3. The fraction of the bandwidth provided by the servers.

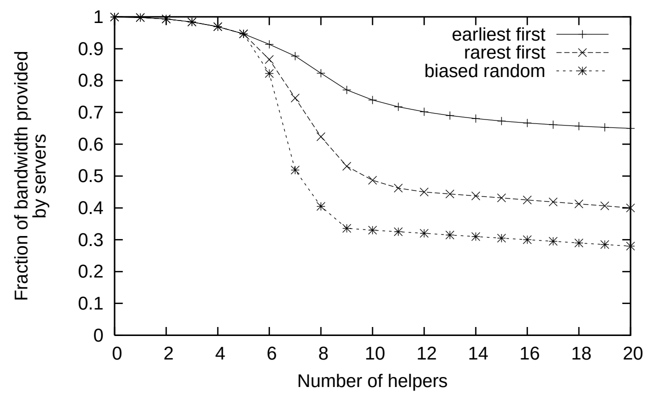

Figure 4. Server bandwidth usage for different piece selection strategies and playback bitrate equal to 1500 kbps.

of helpers is the same for all peers in a single simulation but it varies across different simulations.

Each of the simulated peers maintains a list of randomly selected bartering partners. The number of bartering partners is set to 4 which is the default value in BitTorrent [4] — the most popular P2P data bartering protocol. A peer always gives priority as a data source to its bartering partner and downloads the a piece from a server only if this piece cannot be downloaded on time from the P2P network. Each video piece has a size of 100 kB.

5.2. Results of the experiments

In the first series of experiments we evaluate how the idle bandwidth provided by the helpers influences the server bandwidth consumption. Figure 3 shows the fraction of bandwidth required to satisfy all peers that is provided by the servers. The results are presented for different numbers of helpers in a collaboration and different playback bitrates. All peers in this experiment use the random biased piece selection strategy.

Obviously, the server bandwidth consumption is lower for higher bitrates and lower numbers of helpers involved in the data distribution. Starting with no helpers and up to a breaking point in which the number of helpers is sufficient to guarantee that the total upload capacity of a collaboration is not lower than the playback bitrate, the server bandwidth consumption decreases slowly. Before reaching the breaking point peers concentrate on obtaining the next piece to be played, generally ignoring pieces further away from the playback position and thus limiting the bartering possibilities. Only after the number of helpers passes over the breaking point, the biased random strategy can start selecting pieces leading to a rapid improvement in bartering. The server bandwidth consumption cannot drop to zero as the servers have to constantly inject pieces to compensate for the peers that leave the network. Observing that the average number of peers watching the video at a given time is 5, the fraction of bandwidth contributed by the server (slightly lower than 0.3) is only two times higher than the fraction of bandwidth consumed by each peer watching the video (equal to 0.14).

In the second set of experiments we investigate the impact of the piece selection strategy on the server bandwidth consumption. Figure 4 presents the fraction of bandwidth provided by the servers for the three strategies described in Section 3. We keep a constant playback bitrate of 1500 kbps and vary the number of helpers.

Similarly as in the first set of experiments, for all three strategies, the reduction in server bandwidth consumption is small until the system reaches the breaking point. After this point the differences between the strategies clearly emerge. The earliest first strategy with its fixed preference of pieces closer to the playback position is the least efficient of the three. The rarest first and the random biased strategies exhibit at similar trent although the latter strategy leads to higher savings in the server bandwidth consumption.

References

[1] Wikipedia. http://en.wikipedia.org/wiki/YouTube.

[2] Ari, B. Hong, E. L. Miller, S. Brandt, and D. D. E. Long. Managing flash crowds on the internet. In *11th IEEE/ACM International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS'03)*, Orlando, FL, October 2003.

[3] A. R. Bharambe, C. Herley, and V. N. Padmanabhan. Analyzing and improving a bittorrent network's performance mechanisms. In *IEEE Infocom 2006*, Barcelona, Spain, April 2006.

[4] B. Cohen. Incentives build robustness in bittorrent. In *P2PEcon'03*, Berkeley, CA, May 2003.

[5] P. Garbacki, D. H. J. Epema, J. Pouwelse, and M. van Steen. Offloading servers with collaborative video on demand. Technical Report PDS-2007-05, Delft University of Technology, November 2007. http://www.st.ewi.tudelft.nl/˜pawel/pub/collvod_techrep.pdf.

[6] P. Garbacki, A. Iosup, D. H. J. Epema, and M. van Steen. 2fast: Collaborative downloads in p2p networks. In *6th IEEE Int'l Conference on Peer-to-Peer Computing (P2P2006)*, Cambridge, UK, September 2006.

[7] C. Huang, J. Li, and K. W. Ross. Can internet video-on-demand be profitable? In *ACM SIGCOMM*, Kyoto, Japan, August 2007.

[8] J. J. D. Mol, D. H. J. Epema, and H. J. Sips. The orchard algorithm: P2p multicasting without free-riding. In *6th IEEE Int'l Conference on Peer-to-Peer Computing (P2P2006)*, Cambridge, UK, September 2006.

[9] D. Qiu and R. Srikant. Modeling and performance analysis of bittorrent-like peer-to-peer networks. *SIGCOMM Comput. Commun. Rev.*, 34(4):367–378, October 2004.

[10] J. Shneidman and D. Parkes. Rationality and self-interest in peer to peer networks. In *2nd Int. Workshop on Peer-to-Peer Systems (IPTPS'03)*, February 2003.

[11] S. Tewari and L. Kleinrock. Analytical model for bittorrent- based live video streaming. In *3rd IEEE International Workshop on Networking Issues (NIME'07)*, Las Vegas, NV, January 2007.

[12] J. Wang, C. Yeo, V. Prabhakaran, and K. Ramchandran. On the role of helpers in peer-to-peer file download systems: Design, analysis and simulation. In *International workshop on Peer-To-Peer Systems (IPTPS'07)*, Bellevue, WA, February 2007.