How Impatience Affects the Performance and Scalability of P2P Video-on-Demand Systems

Samuli Aalto[1], Pasi Lassila[1], Petri Savolainen[2], Sasu Tarkoma[2]

[1] Aalto University, Finland
[2] Helsinki Institute for Information Technology, Finland

1. INTRODUCTION

In the past ten years, peer-to-peer (P2P) networks have challenged the traditional client/server networking paradigm. From the performance point of view, the salient feature of P2P networks is their scalability, which originates from the fundamental idea that the peers not only act as clients downloading content from other peers but also serve the other peers by uploading onwards the downloaded contents.

Qiu and Srikant [7] developed a deterministic fluid model to analyze the performance of BitTorrent-like P2P *file sharing* systems under a steady flow arrival scenario. Among the key parameters are the arrival rate of new peers, λ, the efficiency of P2P file sharing, η, and the abort rate of leechers, θ. For the file sharing application, it is generally accepted that $\eta \approx 1$ whenever the number of pieces is sufficiently high. More fine-grained models for P2P file sharing systems have later been developed, e.g., in [3, 2, 5], where the models are analyzed from the stability point of view. Another approach is presented in [4], which uses $M/G/\infty$ queues to model the self-scaling property of P2P file sharing systems.

Inspired by [7], Parvez et al. [6] developed a fluid model to analyze the performance of P2P *video-on-demand* systems. In [1], we presented a slightly different model. An essential difference between our model and that of Parvez et al. is in the handling of the playback phase and the modeling of selfishness. The model in [6] ignores the playback phase and simply assumes that any seed, whether it has played back the media file or not, departs with a constant rate. Instead of that, we include the playback phase explicitly in [1].

While in the context of file sharing a peer may download the pieces in any order, the pieces of a video file have to be retrieved (almost) in their sequential order to enable the on-line playback as required by the video-on-demand application. Therefore, it is clear that $\eta < 1$ in this case. One of our main conclusions in [1] was that the system operates properly (guaranteeing sufficient playback quality) whenever the efficiency parameter η exceeds the following threshold:

$$\eta > \frac{1}{z}(\frac{1}{\mu} - \frac{k}{\lambda}), \qquad (1)$$

where z refers to the playback time of the video, k to the number of permanent seeds, and μ to the upload rate of a single peer (in file transfers per time unit).

Previous P2P video-on-demand models [6, 1], however, omit the fact that the leechers may become impatient during the transfer phase and leave the system before the video file is completely downloaded. In this paper, we study how impatience affects the performance and scalability of BitTorrent-like P2P video-on-demand systems. We find out that a different approach is needed for modeling. Instead of a deterministic fluid model, we propose and develop an approximative stochastic queueing model describing the steady-state behavior of the system, the accuracy of which is verified by simulations. Based on this model, we come, maybe a bit surprisingly, to a conclusion that the most stringent conditions concerning the playback quality are related to the case with the least amount of impatience.

2. MODEL AND ANALYSIS

Let m denote the size of the video file to be shared (in bits). The video is played back at a constant rate w. New peers arrive at rate λ. Each peer is connected to the network over an asymmetric access link with download capacity d and upload capacity u, where $d > u$. The corresponding download and upload rates are: $c = d/m$ and $\mu = u/m$. As justified in [1], we assume that $d > w$.

The life span of a peer consists of two phases, the file transfer phase and the playback phase, which are overlapping. The video transfer and playback proceed in a parallel manner until the entire video is transferred. The transfer rate should be greater than the playback rate so that the video can be played back without any breaks or delays. In such a case, the playback phase extends beyond the transfer phase.

During the transfer phase, the peer is called a leecher. According to the fundamental P2P principle, leechers help each other. Let $\eta \in [0, 1]$ denote the efficiency of this operation. An altruistic leecher becomes a seed as soon as its own video file transfer is completed. Let $\zeta \in [0, 1]$ denote the fraction of altruistic peers, which continue to upload to leechers even after their own transfer phase. Non-altruistic peers are assumed to leave the system immediately after the transfer

phase, while altruistic peers stay in the system until the end of the playback phase but no longer, which is a conservative assumption. If the video is played back without any breaks or delays, the length of the playback phase equals $z = m/w$. In addition to dynamic seeds, we allow permanent seeds, the number of which is denoted by k.

Fluid model: As a new parameter (when compared to [1]), we introduce the abort rate of leechers, $\theta \in [0, \infty)$. A rather straightforward extension of our previous deterministic fluid model would be as follows:

$$
\begin{cases}
x'(t) = \lambda - \theta x(t) - \phi(t), \\
y'(t) = \zeta \phi(t) - \frac{y(t)}{z - x(t)/\phi(t)}, \\
\phi(t) = \min\{cx(t), \mu(\eta x(t) + y(t) + k)\},
\end{cases}
\tag{2}
$$

where $x(t)$ and $y(t)$ denote the number of leechers and (non-permanent) seeds, respectively, at time t. However, as we demonstrate in our numerical examples, the fluid model does not describe the steady-state behavior of the system with satisfactory accuracy for $\theta > 0$. Therefore, we have to take a new approach for modeling.

Stochastic approach: Below we develop an approximative stochastic queueing model to describe the steady-state behavior of the system more precisely. For that, we assume that new peers arrive according to a Poisson process with rate λ, and leechers leave the system before the transfer completion randomly at rate θ. A leecher that completes the transfer phase leaves the system immediately thereafter with probability $1 - \zeta$. Otherwise it stays in the system as a seed until the end of the playback phase.

First we consider the hypothetical pure download constrained case with $\mu \to \infty$. In this case, the time S_x that a peer is a leecher is clearly distributed as $\min\{A, 1/c\}$, where A is an independent exponential random variable with mean $1/\theta$. It follows that the number of leechers, $X(t)$, behaves like the number of customers in an M/G/∞ queue (cf. [4]) with steady-state mean value

$$
x_{\rm d} := E[X] = \lambda E[S_x] = \frac{\lambda}{\theta}(1 - e^{-\theta/c}).
\tag{3}
$$

In addition, we observe that new seeds constitute a Poisson process with rate parameter equal to $\lambda P\{A > 1/c\}\zeta = \lambda e^{-\theta/c}\zeta$, and the length of the remaining playback phase after the completion of the video file transfer is exactly $z - 1/c$. Thus, the number of (non-permanent) seeds, $Y(t)$, behaves like the number of customers in another M/G/∞ queue with steady-state mean value

$$
y_{\rm d} := E[Y] = \lambda e^{-\theta/c}\zeta(z - 1/c).
\tag{4}
$$

Consider now the other extreme, i.e., the pure upload constrained case with $c \to \infty$. Here we have to make a further simplification by assuming that the random variation of the effective transfer capacity per leecher, $\mu(\eta + (Y(t) + k)/X(t))$, can be neglected and its expectation can be approximated by

$$
\tilde{\mu} := \mu(\eta + (E[Y] + k)/E[X]),
\tag{5}
$$

where the steady-state mean values $E[X]$ and $E[Y]$ are given below in (6) and (7), respectively. Under these simplifying

assumptions, we may proceed as above by just substituting $\tilde{\mu}$ for c everywhere. Thus, we get the following implicit formulas for the steady-state mean values:

$$
x_{\rm u} := E[X] = \frac{\lambda}{\theta}(1 - e^{-\theta/\tilde{\mu}}),
\tag{6}
$$

$$
y_{\rm u} := E[Y] = \lambda e^{-\theta/\tilde{\mu}}\zeta(z - 1/\tilde{\mu}).
\tag{7}
$$

The explicit values for $x_{\rm u}$ and $y_{\rm u}$ can be determined numerically from equations (5), (6), and (7).

According to our numerical experiments, the candidates derived above closely approximate the corresponding mean values determined from simulations when the efficiency parameter η is high enough but shows a qualitatively different behavior when η is below a certain threshold. The critical value η_0 is determined by requiring that the (approximate) transfer rate in the upload constrained case equals the playback rate, i.e., $\tilde{\mu} = 1/z$. It follows that

$$
\eta_0 = \frac{1}{z}\Big(\frac{1}{\mu} - \frac{k\theta z}{\lambda(1 - e^{-\theta z})}\Big),
\tag{8}
$$

If the efficiency parameter is below this threshold, $\eta < \eta_0$, the transfer rate for a leecher stays below the playback rate resulting in playback quality problems. Peers have to stay longer in the system than the actual playback time so that $Y(t) = 0$ and the system is upload constrained. Our observation in this case is that the number of leechers and seeds are well estimated by

$$
x_0 := x_{\rm u}|_{\zeta=0}, \quad y_0 := y_{\rm u}|_{\zeta=0} = 0.
\tag{9}
$$

The threshold η_0 is monotonously decreasing with θ implying that the most stringent conditions for the efficiency parameter η (concerning the playback quality) are related to the case with the least amount of impatience, $\theta = 0$. As a consequence, we find that the playback quality is scalable (i.e., independent of the arrival rate) whenever $\eta > w/u$. A necessary condition for this kind of scalability is clearly $u > w$.

3. NUMERICAL RESULTS

In this section, we validate the accuracy of our approximative model against simulations. Unlike the analysis, which is based on the two extreme cases with $c \to \infty$ or $\mu \to \infty$, simulations allow any finite values for c and μ. In the simulations, peers arrive according to a Poisson process and start downloading the video file with a given fixed size at a rate determined by dynamically evolving $\phi(t)$ (cf. (2)) which is assumed to be evenly shared between all leechers.

We consider a scenario where the parameters correspond to a typical YouTube setting. The users are viewing a video file consisting of 800 pieces each 32kB in size. The video coding rate is $w = 300$ kbit/s, and thus the viewing time is $z = 682$ s. The upload and download bandwidths of the users are $u = 512$ kbits/s and $d = 1024$ kbit/s. Also, we assume that $k = 1$. New users arrive with rate $\lambda = 0.2$ peers/s, and leechers leave due to impatience with rate $\theta = 0.001$. Unless stated otherwise, $\eta = 0.8$, which is a reasonable estimate of the efficiency that can be achieved in the present setting by using a windowed BitTorrent protocol [1].
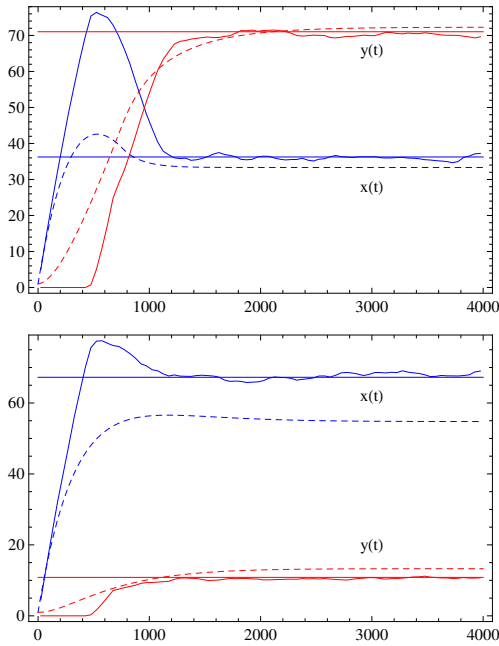
Figure 1: Dynamic evolution of $x(t)$ and $y(t)$ as a function of time in a download constrained case with $\zeta = 0.9$ (upper panel) and an upload constrained case with $\zeta = 0.3$ (lower panel). Dashed lines represent the solution of the fluid model, solid jagged lines the simulated results and the solid horizontal line the solution from the approximative stochastic model.

First we compare the fluid model and the stochastic approach against simulations. The results are given in Figure 1, which depicts as a function of time the mean number of leechers $x(t)$ and seeds $y(t)$ when the system is initially empty. In the upper panel with $\zeta = 0.9$, the system is download constrained, while in the lower panel with $\zeta = 0.3$ the system is upload constrained. As it can be seen, especially in the lower panel, the fluid model is not able to accurately characterize the steady-state performance, while the accuracy of the approximative queueing model is very good.

In Figure 2, we set $\eta = 0.4$, which is below our fundamental quality threshold (8) with the given parameters. In this case, when a leecher downloads the video, it typically takes longer than z seconds and the leecher leaves the system immediately after the download. It can be observed that the steady state is well estimated by equation (9).

Finally, we study the impact of the impatience parameter θ on the steady state. We set $\eta = 0.8$ and $\zeta = 0.9$, and vary $\theta \in (0, 0.01)$. The results are given in Figure 3. For small values of θ, the system is initially download constrained and the values of x and y decrease monotonously as θ increases. However, the steady state solution switches to being upload constrained roughly at $\theta = 0.0050$ at which point there is a discontinuity also in the steady state solution. Comparing with the simulated results, we observe that the correspondence is very good when $\theta < 0.0040$ or $\theta > 0.0055$. In the middle, the simulations indicate a smoother behavior than the approximative stochastic model predicts.
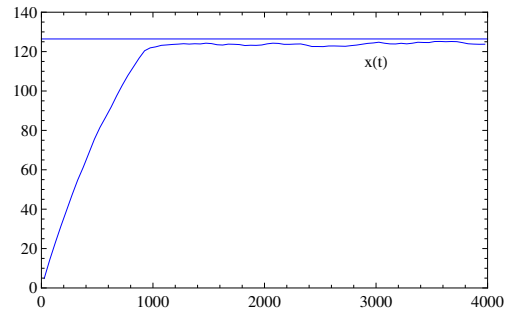

Figure 2: Dynamic evolution of $x(t)$ from simulations (solid jagged line) as a function of time when η is below the threshold (8) compared with the predicted steady state from equation (9) (solid horizontal line).
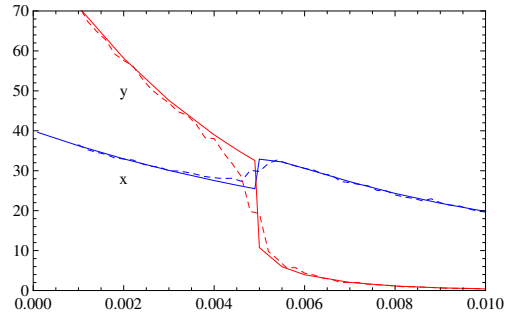

Figure 3: Predicted steady state solutions for x and y (solid lines) from the approximative stochastic model compared with the simulated values (dashed lines) as a function of θ.

In the future, we plan to compare the results against traces from a BitTorrent simulator implementing a windowing algorithm as already done for the case $\theta = 0$ in [1].

4. REFERENCES

[1] S. Aalto, P. Lassila, N. Raatikainen, P. Savolainen, and S. Tarkoma. P2P video-on-demand: Steady state and scalability. In *Proc. of IEEE GLOBECOM*, 2010.

[2] B. Hajek and J. Zhu. The missing piece syndrome in peer-to-peer communication. In *Proc. of IEEE ISIT*, 2010.

[3] L. Massoulié and M. Vojnovic. Coupon replication systems. In *Proc. of ACM SIGMETRICS*, 2005.

[4] D. Menasche, A. Rocha, B. Li, D. Towsley, and A. Venkataramani. Content availability and bundling in swarming systems. In *Proc. of ACM CoNEXT*, 2009.

[5] I. Norros, H. Reittu, and T. Eirola. On the stability of two-chunk file-sharing systems. *Queueing Systems*, 67:183–206, 2011.

[6] K. Parvez, C. Williamson, A. Mahanti, and N. Carlsson. Analysis of BitTorrent-like protocols for on-demand stored media streaming. In *Proc. of ACM SIGMETRICS*, 2008.

[7] D. Qiu and R. Srikant. Modeling and performance analysis of BitTorrent-like peer-to-peer networks. In *Proc. of ACM SIGCOMM*, 2004.