Scalability of the BitTorrent P2P Application

Kolja Eger and Ulrich Killat
Department of Communication Networks, Hamburg University of Technology (TUHH)
{eger, killat}@tu-harburg.de

Peer-to-peer (p2p) networks are popular for their improved scalability compared to the client/server architecture. Hereby, the contribution of upload capacity provided by each peer to the network is the basic principle. BitTorrent is a very popular p2p application for file-sharing which is optimized for fast file dissemination. Key component is its tit-for-tat strategy to decide to whom a peer should upload. This means that an upload to a peer depends on the download from that peer. This gives an incentive for each peer to provide upload capacity to the network.
This work investigates the BitTorrent protocol with respect to scalability. We present a simplified analytical model for file download and compare simulation results of BitTorrent with it. Furthermore, we show that small changes to the protocol can change performance considerably.

Our analytical model describes the download of a file with size S_F, which is hold initially by one peer and $N_P - 1$ other peers are interested in downloading it. This models the flash crowd effect in p2p networks. To accelerate sharing, a file is divided into small pieces, called chunks. So a peer can provide its upload capacity to the network after it has completed only one piece and not the whole file. Assuming all peers have the same access line capacity C, which is the bottleneck in the p2p network, we show that the overall download time for $N_P > 1$ is

$$t_{\text{p2p}} = \frac{S_F}{C} + \frac{S_C}{C} \cdot \left(\lfloor \log_2 (N_P) - 1 \rfloor + \min \left(1, \frac{N_P}{2^{\lfloor \log_2 (N_P) - 1 \rfloor}} - 2 \right) \right). \quad (1)$$

Thereby, S_C is the size of a chunk and no peer leaves the network during or after its download.

Besides the fragmentation of a file the process of selecting other peers for upload, called unchoking, is an important concept in BitTorrent. A peer uploads to a specific number of peers (default = 4) which have the highest upload rates to it measured every 10 seconds. When a peer has completed the download of the file, it is called a seed and unchokes peers based on their download rate rather than the upload rate. An exception is run every 30 s, where a peer is unchoked independently of its rate. This is called an optimistic unchoke. Details about the original implementation of the BitTorrent protocol and architecture can be found in [1]. But it should be noted that different applications implement these rules differently.
To show the influence of small changes on the performance we propose the following modifications to the BitTorrent protocol for the seed algorithm and chunk selection process. In our BitTorrent variant seeds upload to the peers which have the fewest number of chunks. This scheme allows new peers to complete the first chunk quickly and contribute to the network afterwards. Furthermore, we deviate from the strict-priority piece selection rule and switch to the rarest-first strategy when the uploading peer is a seed or when another peer is currently uploading this chunk.

We simulate a 10 MB download of a file in a BitTorrent network with the packet simulator ns-2 [2]. The default piece size of 256 KB is used. Upload capacity of each peer is set to 10 KB/s and download capacity is set 8 times higher. The end-to-end delay between two peers is determined uniform randomly between 25-100 ms.
We present simulation results for the original BitTorrent protocol and our variant, where all peers enter the network at the same time, and compare these results with the analytical model.
The simulations indicate that the overall download time increases slowly for BitTorrent with increasing number of peers. Furthermore, BitTorrent does not reach the values of the analytical model. However, results with our variant are more closely to these theoretical values.

Our work shows that the performance of p2p file distribution can change considerably by small protocol modifications. We conclude that more effort is needed in studying rules for fast and efficient file dissemination. Furthermore, future work will comprise of different simulation scenarios (e.g. Poisson arrival process of peers).

References

[1] B. Cohen. Incentives build robustness in BitTorrent. In *Workshop on Economics of Peer-to-Peer Systems*, Berkeley, CA, June 2003.
[2] *ns-2 (The Network Simulator)*. Sources and Documentation from http://www.isi.edu/nsnam/ns/.