REU 2008 - Power Management for BitTorrent

Jeremy Blackburn, Advisor: Dr. Ken Christensen

Background

BitTorrent is a popular peer-to-peer (P2P) program used for file sharing. It is estimated that 18 to 35 percent of all traffic on the Internet is from BitTorrent[1]. The current implementation of BitTorrent is wasteful of energy as it assumes that computers are powered on 24/7 and maintain TCP connections to peers at all times, even when not actively transferring a file.

Figure 1 shows a BitTorrent network. A file is broken up into several ***pieces***. A ***peer*** is a computer that is interested in the distribution of a file. A ***seed*** is a peer that already has a complete copy of the file being distributed. A ***leech*** is a peer that does not have a full copy of the file being distributed. A ***swarm*** is the set of all peers interested in distributing a file[2].

Hypothesis

By changing a BitTorrent application to decouple peer state from TCP connection state clients can power down for a large part of the day and save energy.

Method

The current implementation of BitTorrent maintains a peer state table. When a peer leaves the swarm its entry in the table is removed. With two simple modifications to the BitTorrent application we allow peers to go to sleep without being forgotten.

1) Seeds that are not currently uploading to any peers are put to sleep but do not remove themselves from the swarm.
2) New peers entering the swarm are able to wake sleeping seeds up.

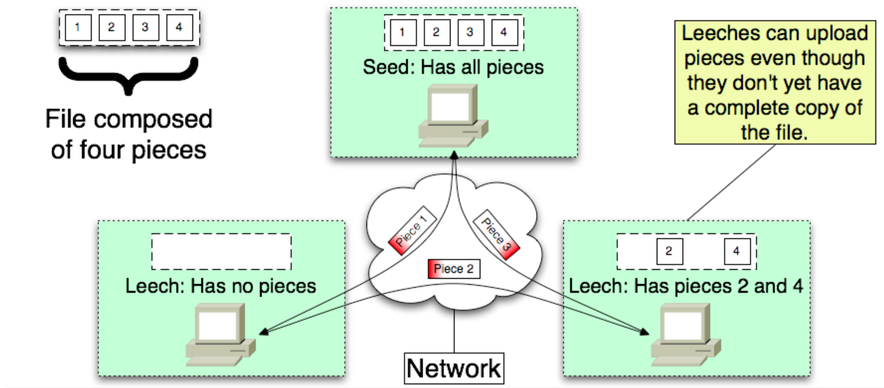

Figure 1: A sample swarm with one seed and two leeches.

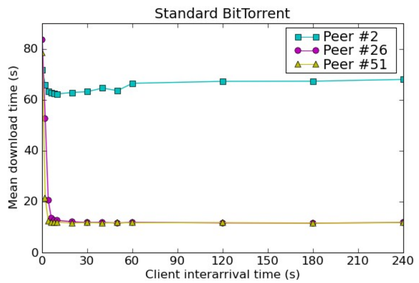

Figure 2: Download times for a 51 peer standard swarm.

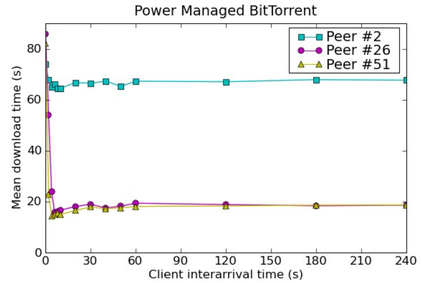

Figure 3: Download times for a 51 peer power managed swarm.

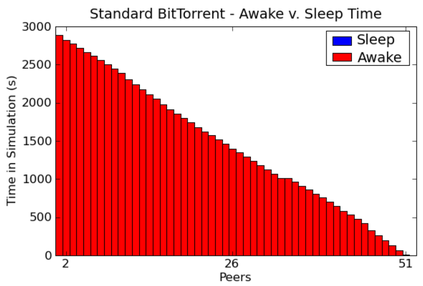

Figure 4: Sleep and awake times for a standard 51 peer swarm with a 60 second average interarrival time.

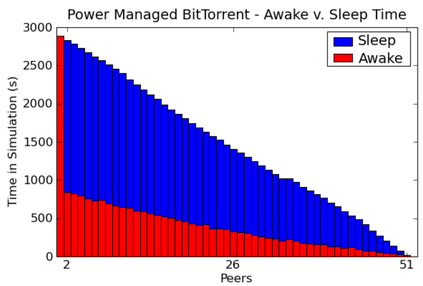

Figure 5: Sleep and awake times for a power managed 51 peer swarm with a 60 second average interarrival time.

Evaluation

Using ns2 we simulate a BitTorrent swarm distributing a 10MB file with 51 peers, one of which is always awake, entering the swarm with exponential interarrival rates of 0, 2, 4, 6, 8, 10, 20, 30, 40, 50, 60, 120, 180, and 240 seconds[3]. We conclude that energy savings of 46%* as seen in figures 4 and 5 are achievable at a performance penalty of 36%* in download time on average as seen in figures 2 and 3. The energy savings possible translates to over $100 million a year.

*Outlier values have been removed from the sample set.

Summary & Future Work

Content distributors are pushing heavily to move the data center into the homes of consumers which in turn distributes the cost of legal content transfers to consumers.

Simulations have shown that simple changes to the BitTorrent application, however, can help reduce energy consumption while still maintaining reasonable download times.

In the future, we will build a real client with our proposed power management functionality to measure the true energy savings possible.

References

[1] The Peer-to-Peer Research Institute, p2presearch.com
[2] B. Cohen, *Incentives Build Robustness in BitTorrent*, In Proceedings of the 1st Workshop on Economics of Peer-to-Peer Systems, 2003
[3] Modified from code made available by K. Eger, T. Hoßfeld, A. Binzenhöfer, G. Kunzmann, *Efficient Simulation of Large-Scale P2P Networks: Packet-level vs. Flow-level Simulations*, 2nd Workshop on the Use of P2P, GRID and Agents for the Development of Content Networks, 2007

USF UNIVERSITY OF SOUTH FLORIDA
Department of Computer Science & Engineering