# Some Observations on BitTorrent Performance

Ashwin R. Bharambe*
Carnegie Mellon University

Cormac Herley
Microsoft Research

Venkata N. Padmanabhan
Microsoft Research

## ABSTRACT

In this paper, we present a simulation-based study of BitTorrent. Our results confirm that BitTorrent performs near-optimally in terms of uplink bandwidth utilization and download time, except under certain extreme conditions. On fairness, however, our work shows that low bandwidth peers systematically download more than they upload to the network when high bandwidth peers are present. We find that the *rate-based* tit-for-tat policy is not effective in preventing unfairness. We show how simple changes to the tracker and a stricter, *block-based tit-for-tat policy*, greatly improves fairness, while maintaining high utilization.

## 1. INTRODUCTION

Recent measurement and analytical studies [2, 3, 4] indicate that BitTorrent is able to handle large distributions effectively, and also scales well. However a number of questions remain unanswered. For example: could BitTorrent achieve even higher download rates than those reported (*e.g.*, by Biersack et al. [2])? How successful are the Local Rarest First (LRF) and Tit-For-Tat (TFT) policies in eliminating the last-block problem and freeloading issues respectively ? If nodes leave after finishing rather than remaining to serve others does the efficacy diminish? The answers depend on a number of parameters that BitTorrent uses.

In this paper, we attempt to answer these questions using a simulator which models the data-plane of BitTorrent.[1] Our discrete-event simulator models peer activity (joins, leaves, block exchanges) as well as many of the associated BitTorrent mechanisms (LRF, TFT, etc.) in detail. The network model associates a downlink and an uplink bandwidth for each node, which allows modeling asymmetric access networks. The simulator uses these bandwidth settings to appropriately delay the blocks exchanged by nodes. The delay calculation takes into account the number of flows that are sharing the uplink or downlink at either end, which may vary with time.

We quantify the effectiveness of BitTorrent in terms of the following metrics: (a) link utilization, (*i.e.*, the ratio of the aggregate traffic flow on all uplinks/downlinks to the aggregate capacity of all uplinks/downlinks) and (b) fairness in terms of the volume of content served. We assume familiarity with the mechanisms of BitTorrent. Readers can consult [2, 3, 4] for good treatments. We use the notation of [1] in which the node degree $d$ is the number of nodes a leecher is in contact with, and $u$ is the maximum number of active upload/download connections.

## 2. IMPROVEMENTS AND EXPERIMENTS

We concentrate on the behavior of BitTorrent when node bandwidth is heterogeneous. A key concern in such environments is fairness in terms of the volume of data served by nodes. We consider two simple mechanisms that can potentially reduce such unfairness: (a) Quick bandwidth estimation (QBE), and (b) Pairwise block-level TFT.

**Quick Bandwidth Estimation:** If a node were able to quickly estimate the upload bandwidth for all its $d$ peers, optimistic unchokes would not be needed. The node could simply unchoke the $u$ peers out of a total of $d$ that offer the highest upload bandwidth. A quick albeit approximate bandwidth estimate could be obtained using lightweight schemes based on the packet-pair principle [5].

**Pairwise Block-Level Tit-for-Tat:** Suppose that node $A$ has uploaded $U_{ab}$ blocks to node $B$ and downloaded $D_{ab}$ blocks from $B$. With pairwise block-level TFT, $A$ allows a block to be uploaded to $B$ if and only if $U_{ab} \leq D_{ab} + \Delta$, where $\Delta$ represents the unfairness threshold on this peer-to-peer connection. This ensures that the maximum number of *extra* blocks served by a node (in excess of what it has downloaded) is bounded by $d\Delta$, where $d$ is the size of its neighborhood. Thus, provided that $\Delta$ is at least one, this policy replaces the optimistic unchoke mechanism and bounds the disparity in the volume of content served.

**Results:** We now present performance results for vanilla BitTorrent as well as the new mechanisms described above with respect to two metrics: (a) mean upload utilization (Figure 1), and (b) unfairness as measured by the maximum number of blocks served by a node (Figure 2). All experiments in this section use the following settings: a flash-crowd of 1000 nodes joins the torrent during the first 10 seconds. In each experiment, there are an equal number of nodes with high-end cable modem (6000 Kbps down; 3000 Kbps up), high-end DSL (1500 Kbps down; 400 Kbps up), and low-end DSL (784 Kbps down; 128 Kbps up) connectivity. We vary the bandwidth of the seed from 800 Kbps to 6000 Kbps.

Figure 1 shows the mean upload utilization of BitTorrent and other policies, as a function of node degree. Figure 2 plots the maximum number of blocks served by a leecher node normalized by the number of blocks in the file. These demonstrate that it is difficult to simultaneously satisfy the

*The author was an intern at Microsoft Research during this work.

[1] We do not consider control-plane issues such as the performance of the centralized *tracker* used for locating peers.

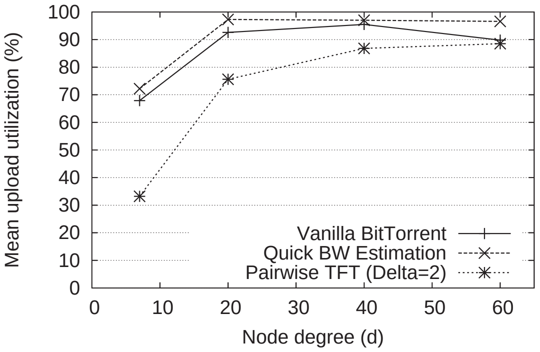


**Figure 1: Mean upload utilization for (a) vanilla BitTorrent, (b) BitTorrent with QBE, and (c) with the pairwise block-level TFT policy.**

competing requirements of having good utilization and fairness. For example, the curve for "Vanilla BitTorrent" achieves excellent upload utilization as shown in Figure 1, but only at the cost of considerable unfairness: Figure 2 shows that some nodes upload up to as seven times as many blocks as they download! It can be seen that the Quick BW Estimation gives some improvement, the bandwidth utilization is excellent, and fairness, as shown in Figure 2, improves a lot, but only when the node degree $d$ is high. On the other hand, Pairwise TFT does an excellent job of ensuring fairness (Figure 2 shows that nodes download only as much as they upload), but this is at the cost of poor bandwidth utilization for small node degrees (the Pairwise TFT curve is considerably below the others in Figure 1). We refer the interested reader to [1] for explanation of the mechanisms that drive these behaviors. One further innovation allows us to achieve improvement in both utilization and fairness.

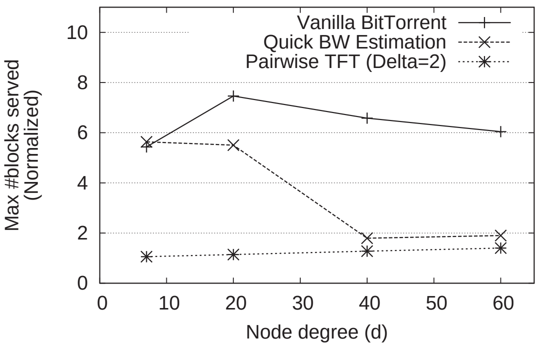


**Figure 2: Maximum number of blocks (normalized by file size) served by any node during an experiment for (a) vanilla BitTorrent, (b) BitTorrent with QBE, and (c) with the pairwise block-level TFT policy.**

**Bandwidth-matching tracker policy:** To alleviate the problems resulting from block transfers between bandwidth-mismatched nodes, we investigate a new *bandwidth-matching tracker* policy. The tracker returns to a new node a set of candidate neighbors biased toward having similar bandwidth to it. This can be accomplished in practice, for instance, by having nodes report their bandwidth to the tracker at the time they join. It can be shown that nodes do not gain by overstating or understating their bandwidths.

Figures 3 and 4 show the upload utilization and fairness metrics, respectively, with the (hybrid) bandwidth-matched

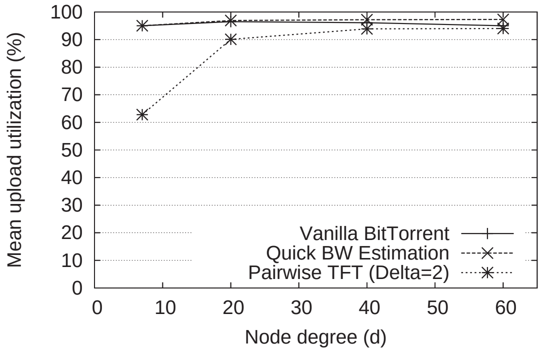


**Figure 3: Mean upload utilization with the bandwidth-matching tracker policy in use for (a) vanilla BitTorrent (but for the new bandwidth-matching tracker policy), (b) BitTorrent with QBE, and (c) with the pairwise block-level TFT policy. Compare with Figure 1.**

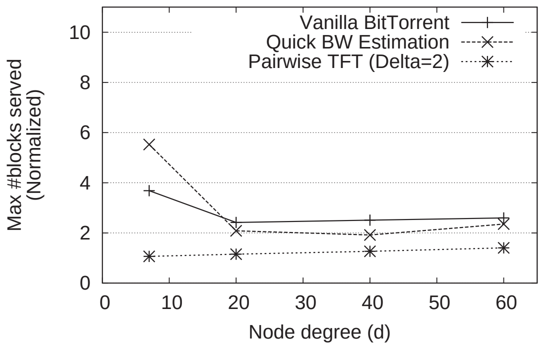


**Figure 4: Maximum number of blocks (normalized by file size) served by any node with the bandwidth-matching tracker policy in use for (a) vanilla BitTorrent (but for the new bandwidth-matching tracker policy), (b) BitTorrent with QBE, and (c) with the pairwise block-level TFT policy. Compare with Figure 2.**

tracker policy in place. We find a significant improvement in both metrics across a range of values of node degree, as can be seen by comparing Figures 3 and 1 and Figures 4 and 2.

**Summary:** Our findings, which we believe have not been reported in the literature to date, are summarized as follows: *(a)* BitTorrent's rate-based Tit-For-Tat (TFT) policy fails to prevent unfairness across nodes in terms of volume of content served. This unfairness arises principally in heterogenous settings when high bandwidth peers connect to low bandwidth ones. *(b)* The combination of Pairwise block-level TFT and the bandwidth matching tracker almost eliminates the unfairness of BitTorrent with a very modest decrease in utilization.

# 3. REFERENCES


[1] A.R. Bharambe, C. Herley, and V.N. Padmanabhan. Analyzing and Improving BitTorrent. Technical Report MSR-TR-2005-03, Microsoft Research, 2005.

[2] M. Izal, G. Urvoy-Keller, E.W. Biersack, P. Felber, A. Al Hamra, and L. Garcés-Erice. Dissecting BitTorrent: Five Months in a Torrent's Lifetime. *PAM*, Apr. 2004.

[3] J.A. Pouwelse, P. Garbacki, D.H.J. Epema, and H.J. Sips. A Measurement Study of the BitTorrent Peer-to-Peer File-Sharing System. Technical Report PDS-2004-003, Delft University of Technology, The Netherlands, April 2004.

[4] D. Qiu and R. Srikant. Modeling and Performance Analysis of BitTorrent-like Peer-to-Peer Networks. *SIGCOMM*, Sep. 2004.

[5] J. Strauss, D. Katabi, and F. Kaashoek. A Measurement Study of Available Bandwidth Estimation Tools. In *IMC*, 2003.