Reading Report #14
Paper: Clustering and Sharing Incentives in BitTorrent Systems
Student: Shuo Yang

This work is an experimental investigation on BitTorrent protocol properties: the clustering of similar-bandwidth peers, the effectiveness of BitTorrent's sharing incentives, and the peers' high uplink utilization. It helps us to better understand BitTorrent protocol. Based on this work, we can do some followup work on BitTorrent protocol. We will look at three aspects: performance, fair sharing and BitTorrent for streaming.

1. *performance*
   One way to improve performance is to explore BitTorrent locality. BitTorrent ignores traffic costs at ISPs and generates a lot of cross-ISP traffic. This is not desirable. We need to improve BitTorrent to enhance its traffic locality, trying to push traffic within an ISP and to minimize traffic across ISPs. Peers in BitTorrent select their neighbors complete randomly. This is the root cause of cross-ISP traffic. If we can make a peer select most of its neighbors within the same ISP, then we can reduce cross-ISP traffic and lead to better performance. To achieve this, BitTorrent needs to be aware of the underlying topology at the ISP level, this definitely complicates the protocol. Another way to improve performance is to encourage nodes to stay as seeds for a longer time. Current BitTorrent lacks incentives for nodes to stay as seeds. If nodes can stay around as seeds after finish downloading, it can help newly joined nodes catch up faster. Due to lack of data pieces they can offer to other nodes, newly joined nodes usually need longer time to reach to a fast and stable download rate. They usually receives data pieces from optimistic unchoking of other leechers as well as from seeds. Therefore having a large number of seeds available can allow them ramp up faster. The question of how to encourage nodes to stay as seeds remains open-ended.

2. *fair sharing*
   BitTorrent has the problem of potentially having peers who consume more than their fair share of the network resources, the so-called free-riders. We need a way to correctly identify free-riders and how to adjust the policy to penalize free-riders and reward non-free-riders. We can also do some work to investigate how these free-riders impact the system performance.

3. *BitTorrent for streaming*
   Though not originally designed for streaming, BitTorrent can be potentially used for streaming. Changes are needed for BitTorrent to be suitable for streaming. The challenge is to meet the sequential playback demands of media streaming applications. One major thing is piece selection: we need better selection policy to satisfy streaming.