

# BISTRO: BitTorrent Based on Space-Time Trade-Off

Roanna Lun<sup>1</sup>, Ye Zhu<sup>1</sup>, Huirong Fu<sup>2</sup>

<sup>1</sup>Department of Electrical and Computer Engineering, Cleveland State University, Cleveland, USA

<sup>2</sup>Department of Computer Science and Engineering, Oakland University, Rochester, USA

Email: r.lun@csuohio.edu, y.zhu61@csuohio.edu, fu@oakland.edu

Received July 2013

## ABSTRACT

Due to the content bundling and the dramatic increase of content size, the download performance in peer-to-peer networks has become a research focus again recently. In this paper, we propose a novel approach to improve the download performance based on the classical space-time trade-off. With the approach, a peer can speed up local downloads in peer-to-peer networks by contributing a portion of local hard disks for the content distribution in peer-to-peer networks. The contribution can bring performance improvement to each peer following the approach and in the meantime improve the overall content distribution performance in a peer-to-peer network. Based on the approach, we propose BISTRO, a BitTorrent based on space-time trade-off. The BISTRO is compatible with the vanilla BitTorrent. Our extensive experiments show that BISTRO can significantly reduce the download time.

**Keywords:** Peer-to-Peer Networking; BitTorrent

## 1. Introduction

According to the recent study [1-3], the majority of file sharing through peer-to-peer networks is adopting the approach called as content bundling. With the content bundling method, publishers can combine multiple related files such as movies of the same genre, episodes of one TV series, and disc images of different flavors of Linux OS and distribute the bundled content in a single large swarm. A user in the swarm may choose one or more files of interest to her in the bundled content to download [3]. In comparison with distributing one single file through a much smaller swarm, the content bundling method can greatly increase the content availability especially for unpopular content [1,2]. In the meantime, the size of the content distributed through peer-to-peer networks is increasing significantly due to the popularity of high-definition multimedia content. Because of the increase of the content size in coupling with the content bundling method, the content distribution performance of peer-to-peer networks has become a research focus again.

In this paper we propose a new approach to improve the download performance of the content distribution with the content bundling method through peer-to-peer networks. The main idea of the approach originates from the classical space-time trade-off [4]. In algorithm researches, the tradeoff can be used to reduce the time to solve a problem at the cost of the space efficiency, *i.e.*, more memory or more storage consumption.

As the storage becomes cheaper, thanks to Moore's Law, the peers in peer-to-peer networks have more disk space that can be used to trade off the time of downloading content. In the new approach, a peer in a swarm contributes a portion of local hard disk for the content distribution even when the peer is not downloading any file. The peer can benefit from the blocks in the contributed storage since these file blocks can be used to exchange back file blocks needed by the peer when the peer is in the downloading phase.

Our major contributions are summarized as follows:

- We propose a novel approach to improve the performance of content distribution through peer-to-peer networks. The approach is based on the classical space-time trade-off.
- We are particularly interested in improving the performance of the BitTorrent protocol due to its dominance in peer-to-peer networks. We develop BitTorrent based on space-time trade-off (BISTRO) with the new approach.
- We verify the new approach with extensive experiments on BISTRO. Our experiments show that the new approach can significantly speed up content distribution, *i.e.*, and reduce the download time.

The rest of the paper is organized as follows: Section 2 defines the problem and briefly introduces the BitTorrent protocols. The detail of the new approach is described in Section 3. We evaluate the performance of BISTRO with extensive experiments in Section 4. Section 5 reviews

related works. We discuss the approach and outline the future work in Section 6. We conclude the paper in Section 7.

## 2. Problem Definition

The goal of this paper is to improve the download performance for peers joining content distribution with the content bundling method. We are particularly interested in BitTorrent due to its dominance in content distribution through peer-to-peer networks: 1) More than 79% of Internet peer-to-peer bandwidth is used by BitTorrent [5]. 2) The estimated number of monthly BitTorrent users is more than a quarter billion, which is more than the number of active users of YouTube and Facebook combined [6].

First we would like to review fundamentals of BitTorrent related to our approach. BitTorrent allows Internet users (*peers*) who are interested in downloading the same content to form a network (*swarm*). In a swarm, peers can download and upload from each other simultaneously. To share a file or bundled files, a torrent file that contains metadata about the shared files must be created. The content distribution is coordinated by a tracker specified in the torrent file. To join the swarm, a peer first obtains the tracker's URL in the torrent file, and then connects to the tracker to ask for other peers' information. Periodically, peers connect to the tracker to provide their downloading status. The tracker also lets peers share information.

In BitTorrent, a large file is divided into smaller file blocks of a fixed size for distribution. The typical size of a file block is 256 KB. Once a peer completely receives a new block of the file, it becomes a possible provider of that block for other peers immediately. Each peer maintains the status of each file block which indicates whether the file block is available for uploading or not. After a file block is completely downloaded, the status of that block will be marked as available for uploading to other peers. The file block status is communicated among peers through BITFIELD message [7].

One of the most important policies in the BitTorrent protocol is the tit-for-tat incentive policy, which is designed to prevent free riding. By the tit-for-tat policy, a peer with a low ratio of the upload rate versus the download rate is choked by data providers. Once the upload activities increased, meaning that the ratio is going up, the peer gets unchoked. Then the peer can continue to download desired file blocks. In other words, more upload contribution made by a peer can bring the peer more file blocks to download.

We believe a new approach to improve the download performance should satisfy the following requirements: 1) The new approach should be compatible with current BitTorrent protocols because of the huge user base. In other

words, peers adopting the new approach should be able to exchange file blocks with vanilla BitTorrent peers according to the *tit-for-tat* policy and help content distribution in a swarm consisting of both vanilla BitTorrent peers and the peers adopting the new approach; 2) The new approach should bring in extra incentives to motivate peers to adopt the new approach. Otherwise no peers will be interested in adopting the new approach.

## 3. BISTRO: BitTorrent Based on Space-Time Trade-Off

In this section, we describe the approach based on the space-time trade-off and present BISTRO. We begin the section with an overview of the approach and then proceed with the details of BISTRO. The rationale behind the new approach and the incentives for peers to use the new approach are introduced at the end of the section.

### 3.1. Overview

The novel approach to improve the download performance in peer-to-peer networks is based on the classical space-time trade-off, which is often used to speed up program execution at the cost of increased memory usage or storage usage in algorithm researches. We extend the idea to the peer-to-peer networks. To speed up local downloading, a peer can contribute a portion of the local hard disk to the content distribution. The storage contributed by the peer can increase the local download speed since more file blocks, including file blocks in the contributed storage can be used for exchanging desired file blocks.

A BISTRO peer can be in one of the two phases: the dormant phase and the download phase. In the dormant phase, a peer is not downloading any desired files. Instead the peer is simply participating in the content distribution and filling the local storage contributed to the content distribution. In the dormant phase, when the contributed storage is full, some replacement methods are used to optimize the storage. When the peer becomes interested in downloading some files in the bundled content, such as the next TV episode in a bundle of TV series, it changes into the download phase. During the download phase, the file blocks in the contributed storage can be used to exchange desired file blocks with other peers.

The details of the BISTRO are described below.

### 3.2. Design of BISTRO

In this section, we first describe the preliminaries including storage configurations and then describe the details of the dormant phase and the download phase. To avoid unnecessary repeating of the original BitTorrent protocol, we focus on the difference between BISTRO and the

vanilla BitTorrent.

**1) Preliminaries:** In this paper, we assume the content files are bundled. A peer joining the distribution of the bundled content may select one or more files in the content bundle to download [3].

Each peer has a portion of local disk space allocated for use by BISTRO to speed up local downloads. In this paper, we call the portion of the local disk space as the contributed storage. The storage is divided into two parts: the content part and the management part. The content part is used to store file blocks and it takes the majority of the space allocated as the contributed storage. The content part is divided into blocks and the size of each block in the content part is the same as the size of file blocks exchanged among BitTorrent peers so that each block of the content part can be used to store one file block. Since most BitTorrent swarms use 256 KB as the block size so the size of the block in the contributed storage is set to 256 KB by default<sup>1</sup>. The management part is used to store information required to manage the storage. For example, the information needed by the block replacement method is kept in the management part.

**2) Dormant Phase:** In the dormant phase, a peer has no desire to download a file from a swarm yet. The peer joins the swarm simply to participate in the content distribution to fill and optimize its contributed storage. So that later, the file blocks in the storage can be used by the peer to maximize the return, *i.e.*, exchange back its desired file blocks according to the *tit-for-tat* policy used by BitTorrent. The pre-filled storage is equivalent to the pre-calculated results stored in memory. In the classical time-space trade-off, the pre-calculated results are used to speed program execution. In BISTRO, the pre-filled storage is used to speed up both local downloads and the content distribution to other peers.

For the dormant phase we extend the vanilla BitTorrent as follows:

**BITFIELD Message:** A vanilla BitTorrent peer exchanges the BITFIELD messages with other peers to announce the file blocks that have already downloaded. A BISTRO peer will also announce the file blocks in its contributed storage so that other peers can download these file blocks in the contributed storage.

**Storage Management:** When the storage is not full, a BISTRO peer simply stores downloaded file blocks in the contributed storage since these file blocks are not of interest by the peer in the dormant phase. But when the storage is full and new file blocks are downloaded, certain blocks in the contributed storage may have to be replaced. Obviously the block replacement method is important to the performance: 1) Because of the *tit-for-tat* policy, file blocks in the contributed storage are essen-

tially bargain chips used to exchange back desired file blocks when the peer is in the download phase. So it is important for a peer to keep most useful bargain chips in the contributed storage to speed up the file downloading in the download phase; 2) In the dormant phase, the peer has no interest in the blocks within the contributed storage. But the file blocks held by the dormant peers affect the content distribution to other peers in the swarm. In this paper, we propose two storage replacement methods:

- **Random Replacement:** The peer randomly selects a file block in the contributed storage and replaces the file block with a newly downloaded file block.
- **Least Frequently Requested (LFR):** In the LFR method, the least frequently requested, *i.e.*, the most unpopular file blocks are replaced. To keep track of requests on each file block in the contributed storage, a request count associated with each file block is kept in the management part of the storage. The heuristic behind the method is that the most popular file blocks are the most useful bargain chips to exchange back desired file blocks in the download phase.

For different content distribution networks, the actual implementation of the dormant phase may vary in terms of participating in the content distribution without the desire to download a file. In some content distribution networks, it is possible for a peer to participate in the distribution passively as a cache node [8]. In other type of content distribution networks, a peer in the dormant phase has to send download requests to participate in the content distribution. For the second type of content distribution networks, the peer in the dormant phase requests file blocks randomly or simple select a file randomly from the bundled content to download. In BISTRO, the second implementation method is used.

The length of the dormant phase depends on when the peer becomes interested in downloading files. Some peers may want to simply contributing the storage for content distribution so that they can use the file blocks in the storage to speed up their future download. Some peers may want to download desired files immediately after joining a BitTorrent swarm.

**3) Download Phase:** Whenever a BISTRO peer becomes interested in downloading files in the swarm, the peer is changed into the download phase.

A BISTRO peer in the download phase acts largely the same as a regular peer in the vanilla BitTorrent. The major difference is that from the beginning of the download phase, the peer has bargain chips, *i.e.*, file blocks in the contributed storage. So when a BISTRO peer advertises the file blocks available for sharing, the BITFIELD message also contains the availability of the file blocks in the contributed storage. So that other peers can request and download the file blocks in the contributed storage.

In the download phase, the contributed storage is not

<sup>1</sup>The block size in the contributed storage can be changed according to the change of the block size used by BitTorrent swarms.

updated. In other words, a peer in the download phase only requests and downloads file blocks in its desired files. In this way, the download bandwidth is only used for downloading the desired files. Since the contributed storage is filled and optimized only in the dormant phase, the benefits brought by the contributed storage are at no cost to the actual downloading of the desired files.

### 3.3. Design Rationales

In the rest of this section, we introduce our design considerations and analyze the benefits of BISTRO.

One of the major design considerations is to make BISTRO compatible with the current BitTorrent protocols because: 1) BitTorrent is the dominant protocol used in the content distribution with a huge user base [9]; 2) Some devices running BitTorrent may not have large amount of disk space available for BISTRO. For example, home routers with the open source firmware such as DD-WRT [10] can run BitTorrent for content distribution. But the storage space in the home routers is relatively small.

BISTRO is compatible with BitTorrent protocols since a vanilla BitTorrent peer is equivalent to a BISTRO peer with an empty contributed storage. The messages exchanged among BISTRO peers are in the same format as the messages exchanged among the vanilla BitTorrent peers.

The second major design consideration is on the incentive to encourage peers to contribute a part of the disk space for the content distribution. Obviously without incentives, most of the peers will not make the contribution. The incentive for a peer to use BISTRO is the less download time for future downloads. According to the *tit-for-tat* policy, the incentive policy in BitTorrent to prevent free-riding, a higher upload rate can exchange for a higher download rate. If the contributed storage is filled with useful bargain chips, the peer holding the storage will benefit from a much faster download speed. So essentially BISTRO amplifies the benefits brought by following the *tit-for-tat* policy. In other words, BISTRO is more encouraging in motivating peers to contribute the local storage in content distribution.

Because of the contributed storage, BISTRO can greatly increase the content availability and in turn speed up the content distribution in BitTorrent swarms. The benefit on the content availability is obvious since the contributed storage can greatly increase the number of file blocks in swarms. We analyze the benefit on the content distribution with the model established for content bundling. According to the content bundling model in [2], the mean download time of a file, denoted as  $[T]$ , can be derived as:

$$E[T] = \frac{s}{\mu} + \frac{1}{r} \cdot P \quad (1)$$

Where  $s$  donates the file size,  $\mu$  donates the mean download rate of peers,  $r$  donates the arrive rate of file content publishers, *i.e.*, seeds in BitTorrent, and  $P$  donates the unavailability of file content. As shown in the equation,  $E [T]$  decreases with  $P$ . In other words, the download time decreases when the file content unavailability decreases. Hence as more file blocks are available at any given time in a BitTorrent swarm because of the contributed storage, the download time decreases. So the download performance is improved. The analysis theoretically proves that BISTRO can speed up the content distribution.

We evaluate the performance of BISTRO with extensive experiments described below.

## 4. Performance Evaluation

In this section we first introduce the experiment setup and then present results of extensive experiments on BISTRO. Due to the space limit, we leave the experiments on size of the contributed storage and length of the dormant phase in the companion technical report [11].

### 4.1. Experimental Settings

We evaluate BISTRO with the ns2 simulator [12]. Since BISTRO is based on BitTorrent and it is compatible with BitTorrent, we implement BISTRO by adding the following key components to the vanilla BitTorrent patch developed by Kolja Eger [7]: 1) A storage class is added and it includes the storage block structure and the functions to manage the storage. The basic information about a storage block, such as the block index, the file ID, the request count, and its status, is defined in this class. In addition, this class defines the implementation of block replacement methods; 2) We add functions necessary to allow a peer in the dormant phase to randomly request file blocks for building up the contributed storage and to exchange blocks in the contributed storage with other peers; 3) An initialization function is added to set the file block status based on pre-filled storage block information. If a file block is found in the contributed storage, then the status of that file block is set to be available, otherwise it is set to unavailable; 4) Functions are added or modified to allow a peer in the download phase to share the file blocks in the contributed storage in exchange for desired file blocks.

In the following experiments, we assume the download capacity and the upload capacity of each peer are 16 Mbps and 2 Mbps respectively.

### 4.2. Performance Metrics

We evaluate the download performance with the download time defined as the time to complete downloading a

file or files. The performance metric used to benchmark one network is the average download time of all peers participating in the BitTorrent network. Since we do not study effectiveness of any incentive methods in BitTorrent in this paper, we assume peers are not required to stay in network after completing downloading all data.

As described in Section 3, a BISTRO peer has no desire to download any file and the peer is simply contributing a portion of local disk to participate in the content distribution passively. So the length of the dormant phase is not a part of download time. In the download phase, a BISTRO peer is actively downloading a desired file so the download time is the length of the download phase for a BISTRO peer.

### 4.3. Size of the Contributed Storage

In this set of experiments, we investigate the download performance with different content file sizes. Among the 100 peers included in this set of experiments, half of the peers are BISTRO peers with 50 MB or 500 MB contributed storage and the other half are vanilla BitTorrent peers. We vary the file size from 100 MB to 1 GB. **Figure 1** shows the experiment results with 50 MB and 500 MB contributed storage. From **Figure 1**, we can observe: 1) The average download time of BISTRO peers in a swarm consisting of half BISTRO peers and half vanilla

peers is consistently 30% less than the download time of peers in the BitTorrent swarm consisting of only vanilla BitTorrent peers. It means BISTRO can greatly reduce the download time; 2) Even the download time taken by the vanilla peers in the swarm with half BISTRO peers and half vanilla peers is 20% less than the download time taken by the peers in the BitTorrent swarm consisting of only vanilla BitTorrent peers. This observation means BISTRO can also help content distribution to vanilla peers; 3) The difference between the download time of BISTRO peers using any of the two block replacement methods and the download time of the vanilla BitTorrent peers is increasing with the file size. In other words, the performance improvement over the vanilla BitTorrent increases with the file size; 4) The performance of BISTRO peers with the LFR replacement method is close to the performance of BISTRO peers with the random replacement method for small file size. When the download file is larger than 700 MB, the LFR replacement methods outperform the random replacement methods.

### 4.4. Number of Peers

In this set of experiments, we study the performance of BISTRO with different swarm sizes, *i.e.*, different number of peers. **Figure 2** shows the results on the random replacement method and the LFR replacement method.

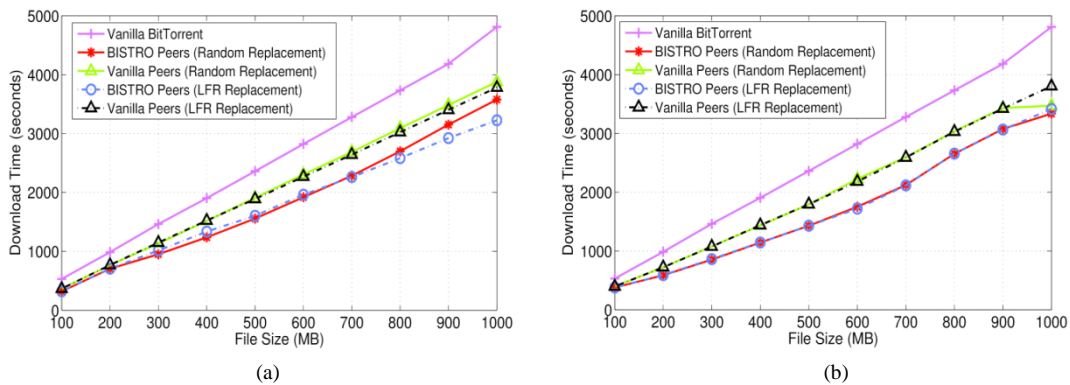


Figure 1. Download Time vs. Download File Size.

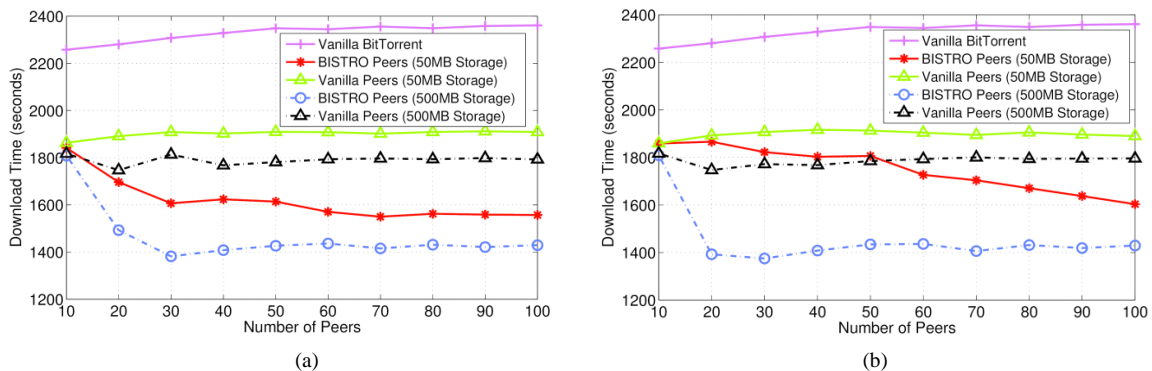


Figure 2. Download Time vs. Number of Peers.

From these two figures, we again observe the large performance improvement of BISTRO over the vanilla BitTorrent. **Figure 2** also shows that the performance improvement increases with the number of peers.

In summary, the experiment results described above indicate that BISTRO can significantly improve the download performance. The performance improvement brought by BISTRO is consistently more than 30%. Our experiments also show that a relatively small contributed storage can bring significant performance improvement.

## 5. Related Work

Since its debut in 2001, BitTorrent, the most popular peer-to-peer file sharing protocol, has attracted a significant amount of researches on its performance. In [13] the improvement of download performance is achieved by collaborative download mechanism. The mechanisms presented in [14,15] have shown that cooperation between peers can also reduce peer's download time. The approaches of utilizing localization among peers for reducing overlay traffic and improving performance were proposed in [16,17]. BISTRO improves the download performance with the contributed storage to trade the local storage for faster download speed.

Lee *et al.* [8] proposed to increase content availability in multi-swarm BitTorrent networks by caching. The proposed approach requires significant changes to the BitTorrent protocols to support the exchange of file blocks in multiple swarms and it involves a large amount of messages exchanged between multiple trackers.

Caching at the ISP level of peer-to-peer networks is proposed by Lehrieder *et al.* [18] to improve the performance of content distribution. The ISP-level approach decreases the inter-ISP traffic by storing popular contents at the ISP level so that the remote peers do not have to download them from peers within the ISP. BISTRO improves the download performance at the peer level and it is compatible with the vanilla BitTorrent protocols.

To the best of our knowledge, BISTRO is the first attempt to speed up downloads in peer-to-peer networks by contributing the local storage space based on the space-time trade-off.

## 6. Discussion and Future Work

The experiments show that BISTRO can greatly reduce the download time when in comparison with the vanilla BitTorrent. But the LRF replacement method used in BISTRO is not significantly better than the random replacement method. Similar results have been reported on cache design in computer architecture [19]: The Least-Recently Used (LRU) method used for cache block replacement is only about 1% better than random replacement method.

Although the performance results from our experiments are consistent with the benchmark results reported in [19] in terms of the difference between the LRF/LRU replacement methods and the random replacement methods, we believe it is possible to further improve BISTRO's performance by improving the replacement method. We plan to make more information such as request trend available to replacement methods. We will compare different replacement methods in terms of performance gain and processing overhead on the additional information.

We also plan to establish a theoretical model to predict the performance of the approach based on the space-time trade-off. The model can be used by peer-to-peer network users to determine the tradeoff between the download performance and the amount of local storage contributed for content distribution.

## 7. Conclusion

In this paper we present a novel approach to improve peers' download performance based on the classical space-time trade-off. The experiment results have shown that the performance improvement over the vanilla BitTorrent is average 30% consistently with different tests. Given the popularity of the low-cost and high-capacity hard disks, we believe the new approach is very promising in practice.

## 8. Acknowledgements

This work was supported in part by the US National Science Foundation (NSF) under grant No. 1144644. Any opinions, findings, conclusions, and recommendations in this paper are those of the authors and do not necessarily reflect the views of the funding agencies.

## REFERENCES

- [1] D. Menasche, G. Neglia, D. Towsley and S. Zilberstein, "Strategic reasoning about Bundling in Swarming Systems," *International Conference on Game Theory for Networks*, 2009, pp. 611-620.
- [2] D. S. Menasche, A. A. de Aragao Rocha, B. Li, D. F. Towsley and A. Venkataramani, "Content Availability and Bundling in Swarming Systems," *Proceedings of the ACM Conference on Emerging Networking Experiments and Technology*, 2009, pp. 121-132.
- [3] J. Han, S. Kim, T. Chung, T. T. Kwon, H.-C. Kim and Y. Choi, "Bundling Practice in Bittorrent: What, How, and Why," *Proceedings of the 12th ACM SIGMETRICS/PERFORMANCE Joint International Conference on Measurement and Modeling of Computer Systems*, 2012, pp. 77-88.
- [4] A. Levitin, "Introduction to the Design and Analysis of Algorithms, 3/E," Addison-Wesley, 2012.
- [5] Internetobservatory.net, "Internet Status Report".

- <http://www.internetobservatory.net>
- [6] A. Carr, "Bittorrent Has More Users Netflix and Hulu Combined-and Doubled".  
<http://www.fastcompany.com/1714001>
- [7] K. Eger, "Bittorrent in ns-2, Patch for NS-2.29".  
<https://sites.google.com/site/koljaeger/bittorrent-simulation-in-ns-2>
- [8] H. Lee, A. Nakao and J. Kim, "Multi-Swarm Collaboration for Improved Content Availability in Bittorrent-Like Systems," *Proceedings of Consumer Communications and Networking Conference (CCNC), IEEE*, 2011, pp. 565-569.
- [9] A. M. Mateus and J. M. Peha, "Quantifying Global Transfers of Copyrighted Content Using Bittorrent".  
<http://papers.ssrn.com/abstract=1985737>
- [10] DD-WRT.com, "Unleash Your Router".  
<http://www.dd-wrt.com>
- [11] R. Lun, Y. Zhu and H. Fu, "Bistro: Bittorrent Based on Space-Time Trade off," Department of Electrical and Computer Engineering, Cleveland State University, Tech. Rep. 20130423a, 2013.  
[http://academic.csuohio.edu/zhu\\_y/techreport/20130423a.pdf](http://academic.csuohio.edu/zhu_y/techreport/20130423a.pdf)
- [12] ISI.edu, "The Network Simulator NS2".  
<http://www.isi.edu/nsnam/ns>
- [13] N. Zeilemaker, M. Capota, A. Bakker and J. Pouwelse, "Tribler: Searchand Stream," *IEEE International Conference on Peer-to-Peer Computing (P2P)*, 2011, pp. 164-165.
- [14] C.-J. Wu, C.-Y. Li and J.-M. Ho, "Improving the Download Time of BitTorrent-Like Systems," *IEEE International Conference on Communications*, 2007, pp. 1125-1129.
- [15] M. Lingjun, P.-S. Tsang and K.-S. Lui, "Improving File Distribution Performance by Grouping in Peer-to-Peer Networks," *IEEE Transactions on Network and Service Management*, 2009, pp. 149-162.
- [16] R. Rahman, M. Meulpolder, D. Hales, J. Pouwelse, D. Epema and H. Sips, "Improving Efficiency and Fairness in p2p Systems with Effort-Based Incentives," *IEEE International Conference on Communications (ICC)*, 2010, pp. 1-5.
- [17] C. Tian, X. Liu, H. Jiang, W. Liu and Y. Wang, "IMPROVING BitTorrent Traffic Performance by Exploiting Geographic Locality," *Global Telecommunications Conference*, 2008, pp. 1-5.
- [18] F. Lehrieder, G. Dan, T. Hossfeld, S. Oechsner and V. Singeorzan, "Caching for Bittorrent-Like p2p Systems: A Simple Fluid Model and Its Implications," *IEEE/ACM Transactions on Networking*, 2011.
- [19] J. L. Hennessy and D. A. Patterson, "Computer Architecture: A Quantitative Approach," 3rd Edition, Morgan Kaufmann Publishers Inc., San Francisco, 2002.