Proceedings of APCC2008 copyright (c) 2008 IEICE 08 SB 0083

Optimizing Shared Upload Bandwidth in BitTorrent-like Peer-to-Peer Networks

Fouzhan Hosseini*†, Ahmad Khonsari*†, Mohammad Sadegh Talebi† and Afshin Moraveji‡
* ECE Department, University of Tehran, Tehran, Iran
† School of Computer Science, IPM, Tehran, Iran
‡ Islamic Azad University, Tehran Gharb Branch, Tehran, Iran
Email: {f.hosseini, mstalebi, ak}@ipm.ir, a_moraveji@wtiau.ac.ir

***Abstract*—Of all applications of peer-to-peer systems BitTorrent-like file sharing systems becomes increasingly popular, and form a large portion of traffic on Internet. In such systems, peers should share their upload bandwidth in order to increase their download rates. Consequently, tuning the upload bandwidth, a costly resource, such that provides a desirable download time is one of the main concerns of peers. In this paper, we model this as an optimization problem which minimize upload bandwidth and is constrained with a maximum acceptable download time. Solving this problem, we propose an iterative algorithm which has the benefit of low complexity and can be implemented with low computation and communication overhead.**

I. Introduction

Peer-to-peer systems brought a revolution in computer networking and opened new ways of providing network services such as content distribution, video streaming, video conferences, and application level multi-cast. In recent years, there has been a growing trend towards peer-to-peer usages such that peer-to-peer systems especially file sharing applications form a significant part of the Internet traffic [1]. Forming 53% of total peer-to-peer traffic [2], Of all peer-to-peer file sharing applications, BitTorrent is presumably the most glorious one. Practically successfulness of BitTorrent, we focus on BitTorrent-like networks.

Since advent of Napester, the first peer-to-peer file sharing applications, numerous research have been conducted on designing new distributed file sharing applications or improving the existing ones, but with explosive increasing demand for peer-to-peer networks, performance analysis of prevailing peer-to-peer applications which opens way of comparison and improving these systems becomes more sophisticated. Although some substantial works have been done to obtain an analytical model for peer-to-peer file sharing applications, to the best of our knowledge a lot of simplifications have been made to existing model, which may harm our understanding of real behavior of such systems. Preparing peer-to-peer networks to deal with dramatic growth of service request in future, we should consider more realistic scenarios in analyzing peer-to-peer systems.

In this paper, we examine the average download time, one of the most important performance metrics of file sharing application. Although performance of peer-to-peer file sharing applications is highly dependent to shared upload bandwidth of selfish peers, to the best of our knowledge, a few works have dealt with the average download time as function of peers collaboration. Using optimization approach, we try to answer how peers can adjust their upload bandwidth in order to achieve a desirable download time. Our work is inspired by the noteworthy model of [3] which has described the BitTorrent behavior in a homogeneous environment while we consider more realistic heterogenous environment.

This paper is organized as follows. We review some related works and give a brief description of BitTorrent in Sections (II) and (III), respectively. In Section (IV), we present the system model and formulate the optimization problem for adjusting upload bandwidth in BitTorrent-like networks. In Section(V), we solve the underlying optimization problem in two cases: when peers know exact value of network's global parameters or just estimate it. Section (VI) presents the simulation results. Finally, the Section (VII) concludes the paper and states some future work directions.

II. Related Work

Due to the success and popularity of BitTorrent, numerous either measurement or analytical based studies have been conducted to clarify the behavior of bitTorrent-like file sharing applications. Qui and Srikant[3] have presented a simple fluid model of peer evolution in BitTorrent and study the performance of system in steady-state. Their results have shown that BitTorrent file sharing mechanisms are very effective, but they did not consider effect of link heterogeneity and peers' selfish behavior. Considering two general categories of free-riders and non-free-riders, Yu et al. [4] have used an analytical model based on [3] to study free-riding in BitTorrent. To investigate performance of BitTorrent-like peer-to-peer networks, Guo et al. [5] have performed widespread measurements and trace analysis and presented a model like [3]. To study the behavior of peer-to-peer streaming systems, Kumar et al. [6] have developed a stochastic fluid model to examine the effect of peer's real-time demand, churn, limited capacities and peer buffering and playback delay.

III. BitTorrent Overview

Providing fast download of popular files makes BitTorrent [7] the most popular peer-to-peer file sharing application. The basic idea proposed by BitTorrent is to divide large files into smaller pieces, called chunks, and peers which begin downloading a new file are able to upload to other peers as soon as they receive the first chunk. Therefore, peers interested in the same file not only download chunks of file from the source but also serve it to each other. This file sharing mechanism distributes uploading cost among all peers employing the overlay.

Two type of peers, seeds and downloaders, are differentiated in BitTorrent. Downloaders are the peers who have not yet finished their Downloads. Such peers, consequently, contribute to both uploading and downloading, while seeds are peers who have all chunks of the file, and just contribute to uploading.

BitTorrent organizes the peers sharing the same file into one peer-to-peer overlay network, and assigns a centralized server called tracker to each overlay. When a peer wants to download a file, first it should connect to the tracker of that file, and then the tracker will return a list of peers participating in sharing the file. Indeed, the only task of tracker is to help peers find each other. To manage file sharing in an effective manner, BitTorrent introduces its piece selection and choking algorithms which helps peer to find the best connection for sending and receiving data in order to minimize the download time. Piece selection algorithms specify the next chunk that peers should request, while choking algorithms help one peer decide which peers exchange data with in order that it can maximize its download rate.

IV. Network Model and Problem Formulation

A. Network Model

1) Background: Peer-to-peer networks are consisting of dynamic nodes which frequently join and leave the system. To model the dynamics of such a peer-to-peer network, we adopt the simple fluid model proposed by Qui *et al* in [3]. In what follows, we briefly describe their fluid model. $x(t)$ and $y(t)$, respectively, denote the number of seeds and downloaders which exist in network at time t. Suppose that the arrival rate of new peers follows a Poisson process with parameter λ, and the time which seeds and downloaders remain in system obey an exponentially distributed stochastic process with mean $1/\gamma$ and $1/\theta$, respectively. They have assumed all peers have the same upload and download bandwidth which are respectively denoted by u and d. Also, the peers whom they have considered do not behave selfishly, i.e peers share their upload bandwidth if they have chunks which other peers need. Let η indicates the effectiveness of file sharing in network, i.e. the probability that the peer possess a chunk that the connected downloaders need. As presented in [3], considering homogeneous peers, the total feasible downloading and uploading bandwidth will be $dx(t)$ and $u(\eta x(t) + y(t))$, respectively. Therefore, at time t the total uploading rate of the network can be expressed as $\min\{dx(t), u(\eta x(t) + y(t))\}$. Dynamics of such a simplified peer-to-peer network, can be modeled using a simple fluid model, as following [3]:

$$\frac{dx}{dt} = \lambda - \theta x(t) - \min\{dx(t), u(\eta x(t) + y(t))\}, \quad (1)$$

$$\frac{dy}{dt} = \min\{dx(t), u(\eta x(t) + y(t))\} - \gamma y(t) \quad (2)$$

This fluid model describes peer evolution in BitTorrent-like peer-to-peer networks.

TABLE I
NOTATION USED IN NETWORK MODEL.

$x(t)$	Number of downloaders at time t.
$y(t)$	Number of seeds at time t.
u_n	The upload bandwidth of peer n.
d_n	The download bandwidth of peer n.
η_n	Shared upload ratio of peer n.
T_n	The download time of peer n.
$\hat{\lambda}_n$	the estimated arrival rate of new requests.
$\hat{\gamma}_n$	the estimated departure rate of seeds.
$\hat{\theta}_n$	the estimated abort rate of downloaders.

2) Our Model: We focus on a BitTorrent-like peer-to-peer network, which is governed by (1) and (2). However, one can justify that for many realistic scenarios, such a simplistic assumptions might not hold any longer, especially for those which have selfish and heterogeneous peers. In this regard, we do not restrict ourselves to such an oversimplified scenario; as mentioned earlier, we assume that there are selfish peers which do not share their total upload bandwidth. In addition peers are heterogeneous, i.e. upload and download bandwidth of each peer differs from other peers in general. We also assume that peers do not have complete knowledge of global parameters of the network, e.g. peer arrival rate, departure rate of seeds, etc. In fact, we assume that each peer only has an estimation of such parameter. In this respect, we think of each peer n as having an estimation of global parameters of the network, i.e. λ, θ and γ, denoted by $\hat{\lambda}_n$, $\hat{\theta}_n$ and $\hat{\gamma}_n$, respectively.

We consider a BitTorrent-like peer-to-peer network consisting of a set $\mathcal{N} = \{1, \ldots, N\}$ of peers. For peer $n \in \mathcal{N}$ we associate an upload and download bandwidth, denoted by u_n and d_n, respectively. To model bandwidth heterogeneity, the bandwidth of peers can take different values. As upload bandwidth is usually costly, the peers prefer not to share their upload bandwidth. Let $\eta_n \in [0, 1]$, be the portion of upload bandwidth u_n which is shared by peer n. In [3], η indicates effectiveness of file sharing, i.e. the portion of content that each downloader can gain to share with others. But, in a more general model, we can assume that η_n denotes the proportion of shared upload bandwidth of peer n. It is obvious that this assumption can be adapted to the model of [3]. The parameters which have been used to describe the network model are summarized in table I.

B. Problem Formulation

In peer-to-peer file sharing applications, download time is one of the main factors of system performance. Thus, Having analyzed the above fluid model, Qui *et al.*[3] derived the average download time in steady state regime as following:

$$T = \frac{1}{\theta + \beta} \quad (3)$$

where

$$\frac{1}{\beta} = \max\left(\frac{1}{d}, \frac{1}{\eta}\left(\frac{1}{u} - \frac{1}{\gamma}\right)\right). \quad (4)$$

As (3) offers, the average download time is a non-increasing function of η; hence it's trivial that in such a simplified model the least average download time is achieved when η approaches one. However, for the network model defined in the previous subsection, such a trivial solution dose not hold any longer and according to parameters of each peer, the best choice for η varies over the network.

To overcome selfish behaviors, BitTorrent utilizes tit-for-tat mechanism. Consequently, each peer must share its upload bandwidth to decrease its download time. Indeed, there is trade-off between the shared upload bandwidth and download time for each peer. Thus, in BitTorrent-like network, the objective is choosing appropriate η_n for all $n \in \mathcal{N}$ so as to minimize the shared upload bandwidth of each peer while preserving the average download time below a threshold. Hence the optimization problem can be formulated as:

$$\min_{\eta_1,\dots,\eta_n} \sum_{n=1}^{N} \eta_n u_n \tag{5}$$

subject to:

$$\mathbb{E}(T) < Z \tag{6}$$

$$0 \leq \eta_n \leq 1; \quad n = 1, \dots, N \tag{7}$$

where the expectation in (6) is taken over all peers. For sufficiently large n, it can be shown that the download time of all peers are independent and identically distributed random variables, and therefore, $\mathbb{E}(T)$ is given by:

$$\mathbb{E}(T) = \lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} T_n \tag{8}$$

In more detail, $\mathbb{E}(T)$ can be approximated by:

$$\mathbb{E}(T) \approx \frac{1}{N} \sum_{n=1}^{N} T_n \tag{9}$$

Assuming peer n chooses η_n to balance its upload and download, download time of peer n in the steady state regime can be estimated according to the average download time given by (3). Thus, for peer n, T_n is given by:

$$\begin{aligned} T_n &= \frac{1}{\hat{\theta}_n + \beta_n} && (10)\\ &= \frac{1}{\hat{\theta}_n + 1/\max\{\frac{1}{d_n}, \frac{1}{\eta_n}(\frac{1}{u_n} - \frac{1}{\hat{\gamma}_n})\}} && (11)\end{aligned}$$

We assume that download bandwidth for all peers dominates the upload, so that we have

$$\frac{1}{d_n} > \frac{1}{\eta_n}\left(\frac{1}{u_n} - \frac{1}{\hat{\gamma}_n}\right) \tag{12}$$

which yields

$$T_n = \frac{1}{\hat{\theta}_n + \frac{\eta_n}{K_n}} \tag{13}$$

where

$$K_n = \frac{1}{u_n} - \frac{1}{\hat{\gamma}_n} \tag{14}$$

Therefore, (5), can be expressed in more detail as:

$$\min_{\eta_1,\dots,\eta_n} \sum_{n=1}^{N} \eta_n u_n \tag{15}$$

subject to:

$$\frac{1}{N} \sum_{n=1}^{N} \frac{1}{\hat{\theta}_n + \frac{\eta_n}{K_n}} \quad < \quad Z \tag{16}$$

$$0 \leq \eta_n \leq \min\{1, d_n K_n\} \quad ; \qquad n \in \mathcal{N} \tag{17}$$

We would like to decompose problem (15) to N subproblems so that each subproblm n only comprises the parameters of peer n and its neighbors. In this respect, we assign a threshold value to peer n which yields the subproblems of peer n as follows:

$$\min_{\eta_n} \eta_n u_n \tag{18}$$

subject to:

$$\frac{1}{\hat{\theta}_n + \frac{\eta_n}{K_n}} < Z_n \tag{19}$$

$$0 \leq \eta_n \leq \min\{1, d_n K_n\} \tag{20}$$

In this respect, we would like that each user be able to estimate (9), only using local information of the network, i.e. with its own parameters and those of its neighbors. More precisely, peer n might acquire such a threshold by simply obtaining the average of thresholds of its neighbors. Let $\mathcal{N}_n \subset \mathcal{N}$ represent the set of neighbors of peer n. Peer n calculates its threshold, Z_n, as following

$$Z_n = \frac{1}{|\mathcal{N}_n|} \sum_{i\in\mathcal{N}_n} T_i \tag{21}$$

Therefore, Z_n is a function of $(\eta_i, i \in \mathcal{N}_n)$, $(d_i, i \in \mathcal{N}_n)$ and $(u_i, i \in \mathcal{N}_n)$. For the sake of simplicity, we omit such a dependency in notation.

We defer solving (18) until the next section.

V. Optimal Solution

The objective function of problem (18) is affine and the associated constraints are linear; hence problem (18) is a convex optimization problem and therefore admits a unique optimal point [8][9], which in vector form is denoted by $\eta^* = (\eta_n^*, n \in \mathcal{N})$. In other words, there exist a unique η^*, which maximizes total download of peers while maintaining the average download time below a threshold. Such a constrained problem could be solved using Interior Point Method, which poses great computation complexity into the network, and also cannot be addressed in distributed scenarios.

Such problems might be solved indirectly via their dual problems [8][9], which can be solved using simple iterative methods due to unconstrained nature of them. In this section, we solve (18) using its dual. We start our solution procedure by defining dual function and thereby dual problem of (18) via the definition of the Lagrangian.

Lagrangian of (18) is given by:

$$L(\eta_n, \nu_n) = \eta_n u_n + \nu_n \left(\frac{1}{\hat{\theta}_n + \frac{\eta_n}{K_n}} - Z_n \right) \tag{22}$$

where ν_n is the positive Lagrange multiplier associated with constraint (19) and η_n is assumed to fall within $[0, \min\{1, c_n K_n\}]$. Based on KKT condition, optimality condition necessitates that:

$$\nabla L(\eta_n, \nu_n) \mid_{(\eta_n^*, \nu_n^*)} = 0 \tag{23}$$

Therefore,

$$u_n - \frac{\nu_n^*}{K_n} \frac{1}{(\hat{\theta}_n + \eta_n^*/K_n)^2} = 0 \tag{24}$$

which yields

$$\eta_n^* = \Big[\sqrt{\frac{\nu_n^* K_n}{u_n}} - \hat{\theta}_n K_n\Big]_0^{\min\{1, d_n K_n\}} \tag{25}$$

Dual function of problem (15) is defined as:

$$D(\nu_n) = L(\eta_n^*, \nu_n) \tag{26}$$

Therefore, using (25), we have

$$D(\nu_n) = 2\sqrt{d_n K_n \nu_n} - Z_n \nu_n - d_n K_n \hat{\theta}_n \tag{27}$$

Dual problem of (15) is defined as

$$\max_{\nu_n \geq 0} D(\nu_n) \tag{28}$$

where $D(\nu_n)$ is given by (27). Dual problem is a convex optimization problem, Hence it can be solved using iterative methods. In the sequel, we solve (28) using *Projected Subgradient Method* [8].

A subgradient of function f at x can be any vector g which satisfies

$$f(y) \geq f(x) + g^T(y - x) \tag{29}$$

for all y. It's worthmentioning that for differentiable function f, subgradient reduces to the well-known gradient, i.e. $g = \nabla f$.

Given an objective function, say f, to be minimized over a feasible domain, the projected subgradient method at each iteration, steps toward the opposite direction of its subgradient, denoted by g, as follows:

$$x^{k+1} = \mathcal{P}\left(x^k - \alpha_k g^k\right) \tag{30}$$

where x^k is the value of kth iteration, α_k is the kth step size, and $\mathcal{P}$ is the projection operator, which guarantees that x^{k+1} will remain in the feasible domain. For the projected subgradient method to converge to the optimal point, α_k should admit the following conditions:

$$\alpha_k > 0; \qquad \forall k \tag{31}$$

$$\sum_{k=1}^{\infty} \alpha_k^2 < \infty \tag{32}$$

$$\sum_{k=1}^{\infty} \alpha_k = \infty \tag{33}$$

In the sequel, we consider two different scenarios to solve (28) using the projected subgradient method. In the first one, we assume that all peers have exact knowledge of the global parameters, while in the second, peers have only inexact estimations of them.

A. Exact Global Parameters

In this scenario, peers are assumed to have exact knowledge about the global parameters. In other words, their estimations of global parameters are supposed to be arbitrarily precise; i.e $\forall n \in \mathcal{N}$:

$$\hat{\lambda}_n = \lambda \tag{34}$$

$$\hat{\theta}_n = \theta$$

$$\hat{\gamma}_n = \gamma$$

In this respect, the update equation (30) for ν_n will be

$$\nu_n^{k+1} = \left[\nu_n^k + \alpha_k \left(\sqrt{\frac{u_n K_n}{\nu_n}} - Z_n\right)\right]^+ \tag{35}$$

where $K_n = 1/u_n - 1/\gamma_n$ and $[z]^+ = \max\{z, 0\}$.

B. Estimated Global Parameters

In this scenario, peers are assumed to have inexact knowledge about the global parameters. In this case, the update equation for η_n will be

$$\nu_n^{k+1} = \left[\nu_n^k + \alpha_k \left(\sqrt{\frac{u_n K_n}{\nu_n}} - Z_n\right)\right]^+ \tag{36}$$

C. The Algorithm

The (25) and (35) (or (36)) together can be used to solve (28), in an iterative fashion. Problem (15) is a convex optimization problem, and therefore the duality gap, i.e. the gap between the dual-optimal and primal-optimal is zero. As a result, solving (28), or equivalently obtaining ν_n^*, through (25) leads to η_n^* which solves the primal problem.

The final result of the above discussion for the second case, is listed below as Algorithm 1.

Algorithm 1 Adjusting Shared Upload Bandwith

Initialization
Initialize the following items:
1. Sets of neighboring peers.
2. d_n and u_n for $n \in \mathcal{N}$.

Main Loop
While $t <$ `max_iteration` AND `status`$\neq$`converged`

Step 1
$K_n = 1/u_n - 1/\hat{\gamma}_n$
$T_n = 1/(\hat{\theta} + \eta_n/K_n)$
$Z_n = \frac{1}{|\mathcal{N}_n|}\sum_{i \in \mathcal{N}_n} T_i$

Step 2
$\nu_n^{t+1} = \left[\nu_n^t + \alpha_t \left(\sqrt{\frac{d_n K_n}{\nu_n}} - Z_n\right)\right]^+$

Step 3
$\eta_n^t = \Big[\sqrt{\frac{\nu_n^t K_n}{d_n}} - \hat{\theta}_n K_n\Big]_0^{\min\{1, d_n K_n\}}$

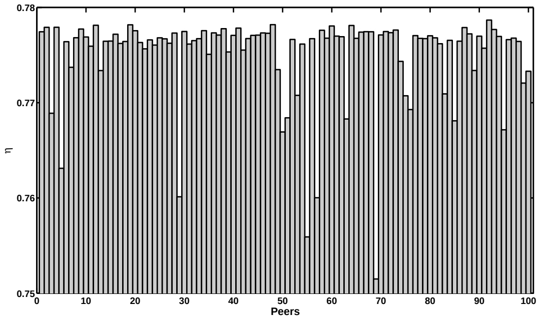
end while
end

Algorithm 1. Adjusting Shared Upload Bandwith

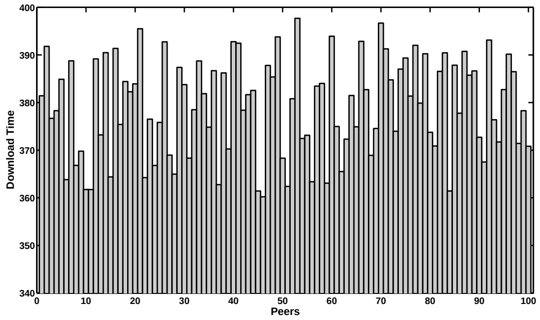
VI. Simulation Experiments

We conducted a series of experiments to examine the proposed tuning upload bandwidth algorithm for a typical BitTorrent-like peer-to-peer network. First, we examine download time and shared upload bandwidth of peers in a typical peer-to-peer file sharing network, the peers of which follow the proposed algorithm. Then, we illustrate the effect of peer's having either exact global parameter of the network or just an estimation of it.

In our scenarios for the experiments, we consider networks which consist of peers with various upload and download bandwidth. To simulate this variety, the upload and download bandwidth of each peer , i.e. d_n and u_n set to a random variable which follows Unique distribution on $[0.0005, 0.0007]$ and $[0.0004, 0.0005]$, respectively. Also, we set parameters $\theta = 0.002$ and $\gamma = 0.001$ and assume that all peers know the exact value of these parameters at first. We assume the network consists of 100 peers which is the average population of torrents in BitTorrent [5]. The peers' download time and shared upload ratio, η_n, in a network, the peers of which follows Algorithm 1, are shown in Fig. 1. Each peer, receiving just d_n and

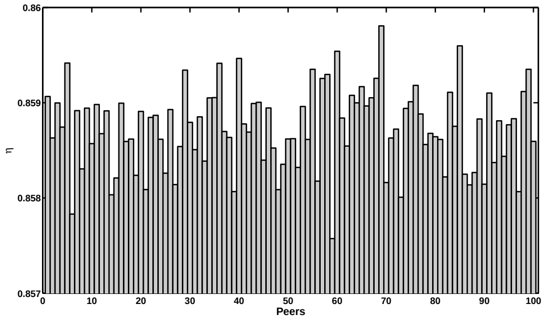


(a) Shared Upload Bandwidth of Each Peer

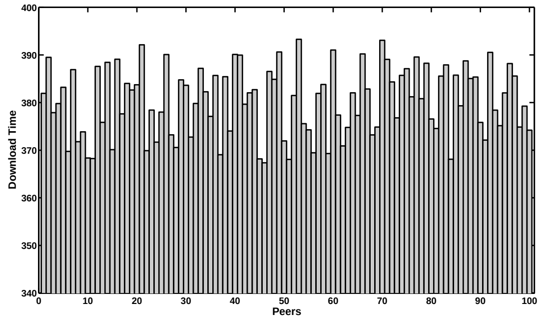


(b) Download Time of Each Peer

Fig. 1. Global network parameters are known.



(a) Shared Upload Bandwidth of Each Peer



(b) Download Time of Each Peer

Fig. 2. Global network parameters are estimated.

u_n of its neighbors and knowing the global parameters of the network, adjust its η_n such that its download time approximate the average download time of all peers.

Obviously, knowing the exact value of global parameters of the network helps peers to adjust their shared upload bandwidth to a better value, although Algorithm 1 is effective when peers just know the estimated parameters. To clarify the effect of estimating the global parameters of the network by each peer, in another scenario, we assume that peers estimate the value of θ and γ, i.e, each peer knows an $\hat{\theta}$ and $\hat{\gamma}$, and by receiving the estimation of its neighbors, each peer update its estimation of these values. The download time and shared upload ratio, η_n, of all peers, in this conditions are shown in Fig. 2. As shown in Fig 2, both shared upload ratio and the average download time of peers increase. However, the difference between download time, in two scenarios, are not significant, while the η_n of all peers are set to a higher value.

VII. Conclusion

In this paper, we addressed the problem of tuning shared upload bandwidth of peers in BitTorrent-like networks as a solution of the optimization problem which maximized the overall download rate of peers, i.e, minimized their upload rate, while the average download time could not exceed one threshold. Decomposing these problem and using projected subgradient method, the solution was led to an distributed iterative algorithm which can be used to determine optimal value of shared upload bandwidth. The algorithm can be implemented with light communications between each peer and its neighbors, and communication overhead will not be considerable to the system. Further investigation into the convergence speed and the fairness property of the proposed algorithm are the main direction of our future studies.

References

[1] T. Karagiannis, A. Broido, N. Brownlee, kc claffy, and M. Faloutsos, "Is p2p dying or just hiding," in *Proceedings of IEEE Global Communications Conference(GLOBECOM)'04*, November 2004.
[2] A. Parker, "the true picture of p2p file sharing," http://cachelogic.com, June 2004.
[3] D. Qui and R. Srikant, "Modeling and Performance Analysis of BitTorrent-like Peer-to-Peer Networks," in *Proceedings of the ACM SIGCOMM '04 Symposium*, August 2004.
[4] J. Yu, M. Li, and J. Wu, "Modeling Analysis and Improvement for Free-Riding on Bittorrent-like File Sharing Systems," in *International Conference on Paralleled Processing Workshops(ICPPW)*. IEEE Computer Society, 2007.
[5] L. Guo, S. Chen, and Z. Xiao, "A Performance Study of BitTorrent-like Peer-toPeer Systems," *IEEE Journal on Selected Areas in Communications (JSAC)*, vol. 25, no. 1, pp. 155–170, January 2007.
[6] R. Kumar, Y. Liu, and K. Ross, "Stochastic Fluid Theory for P2P Streaming Systems," in *Proceeding of IEEE infocom'07*, 2007.
[7] B. Cohen, "Incentives build robustness in bittorrent," http://www.bittorrent.org/bittorrentecon.pdf, May 2003.
[8] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2003.
[9] D. Bertsekas, *Nonlinear Programming*. Athena Scientific, 1999.