SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

October 2015, Vol. 58 102303:1–102303:15
doi: 10.1007/s11432-014-5268-4

A bottleneck-free model for P4P

LIU Ying[1], ZHANG ShengLin[1]* & LIU HongYing[2]

[1]*Institute for Network Sciences and Cyberspace, Tsinghua University, Beijing* 100084*, China;*
[2]*School of Mathematics and Systems Science, Beihang University, Beijing* 100191*, China*

Received April 23, 2014; accepted October 23, 2014; published online April 23, 2015

Abstract Provider Portal for Applications (P4P) is a model aiming to incorporate (peer to peer) P2P applications with Internet Service Providers (ISPs) and improve the performance of the both ISP and the P2P applications. In this study, we have analyzed the relationship between the link traffic and the P-distance, which is the core interface of P4P. In addition, the limitation of P4P in dealing with network applications having bottleneck links is illustrated. Furthermore, taking link utility function as the optimization objective, we propose a Bottleneck-Free model for P4P (BFP), making the traffic produced by P2P applications more homogeneous, which can reduce the peak link utilization, protect bottleneck links, and thereby improve both the network efficiency and the P2P performance. We have built a simulation platform based on BitTorrent and conducted extensive simulations. The simulation results demonstrate that BFP achieves a lower cost for ISPs and a better performance for P2P applications than P4P and native P2P applications, both in intra-domain and inter-domain settings. BFP performs steadily in different topologies and swarm sizes, proving that it is scalable and easy to deploy.

Keywords subgradient method, P2P, traffic engineering, cooperative model, duality

Citation Liu Y, Zhang S L, Liu H Y. A bottleneck-free model for P4P. Sci China Inf Sci, 2015, 58: 102303(15), doi: 10.1007/s11432-014-5268-4

1 Introduction

Peer-to-peer (P2P) technology has fundamental advantages over the traditional Client/Server(C/S) model and the fixed infrastructure content distribution networks because of its excellent robustness and scalability, and it plays an important role in modern networks. Some researchers have found that more than 50% of the network traffic is introduced by P2P applications [1,2], and the massive traffic generated by P2P brings significant challenges in traffic engineering for Internet Service Providers (ISPs) [3].

Because P2P applications are ignorant of the underlying network topology, most P2P applications apply application-level routing that is based only on the overlay network metrics [4]. Moreover, some P2P applications select the source of download randomly, which may lead a P2P user to download data from far-away users even though the resources are already available in vicinity. This kind of long-distance download may decrease network efficiency and cause the decline in performance of the P2P applications concurrently. P2P applications can avoid this by selecting neighbors with a lower delay or less hops. But purely selecting neighbors with a low delay or less hops may cause a Comcast user to select an AT&T user

* Corresponding author (email: zhangsl12@mails.tsinghua.edu.cn)

as its neighbor, leading to additional inter-domain traffic. In conclusion, the current P2P applications have the following problems:

(1) A P2P system may cause the dispersion of network traffic and make the traffic flow unnecessarily through multiple intra-domain links. By conducting practical tests, [5] found that every bit of P2P traffic in Verizon needs 5.5 hops while passing through 1000 miles on average, and this average number of hops can be reduced to 0.89 without compromising P2P application performance.

(2) A P2P system may generate massive inter-domain traffic or cause massive traffic produced by multiple ISPs through a specific network [6]. In [7], Karagiannis studied the BitTorrent performance in a college network. He found that this low-efficient inter-domain traffic may cause significant financial loss with ISPs. Even in the top-level ISPs (tier-1 ISPs who do not pay other ISPs), the inter-domain traffic caused by a P2P system can cause the traffic between these tier-1 ISPs to lose balance and then violate the P2P protocol.

(3) The dynamic traffic allocation of a P2P system may not coexist with network traffic engineering. ISPs spare little effort in evaluating a traffic model and determining the routing according to the model. However, a P2P system can adjust a network change by altering its traffic, which leads to ISP failures, and causing oscillatory flow pattern and suboptimal routing selection.

Given the aforementioned issues, both ISPs and P2P applications have proposed a series of solutions. On the one hand, ISPs have attempted a diversity of flow control techniques, including charging and choking. However, without the cooperation of P2P applications, these techniques are not ideal in improving network efficiency and may even decrease the P2P performance dramatically. On the other hand, P2P systems have also increased network efficiency with the flexibility of P2P operations. Some popular P2P applications, such as Kontiki, attempt to localize P2P operations in the same autonomous system in order to enhance network efficiency. However, such a localization requires P2P applications to deduce a variety of network information, such as network topology, congestion state, communication costs, and network policy. However, the deduction of such information, particularly the later two is considerably difficult.

In conclusion, a one-sided strategy for P2P applications and ISPs is not ideal. Instead, a cooperative model should be built, making both sides exchange information and control P2P traffic cooperatively, and thus improving network efficiency and P2P performance simultaneously. There have been several approaches for this purpose [5,8–12].

Provider Portal for Applications (P4P) [5] is a remarkable solution trying to incorporate P2P technology with ISPs and get better performance for both P2P applications and ISP. This interactive model is illustrated in Figure 2 of [5]. The iTracker provides a part of the network information, and with this information, P2P applications can optimize their traffic strategy, reduce their blindness in building an application-level network topology, and eventually optimize the co-benefits of P2P applications and ISPs. At the same time, combined with its optimization objective (such as the min-max link utilization), an ISP can better utilize its network resources, reduce congestion, and then provide P2P and other applications better service with the considerable directions received from iTracker.

There are multiple interfaces for ISPs and P2P applications in P4P, of which the P-distance interface is the most important one. In this paper, the relationship between the P-distance and the link traffic is explored, and the limitation of P4P in dealing with network applications having bottleneck links is analyzed theoretically. To make the theory easier to understand, we illustrate the limitation of P4P with an instance in the given scenario.

Further, we follow the methodology introduced in congestion control [13] and make the link utility function as the ISP optimization objective. We propose a Bottleneck-Free model for P4P (BFP), aiming to make the traffic produced by P2P applications more homogeneous. Theoretically, we prove that BFP can reduce the peak link utilization and protect bottleneck links, and then improve both network efficiency and P2P performance.

We then build a BitTorrent-based simulation platform and conduct abundant simulations. These simulations demonstrate that compared with native P2P applications and P4P, BFP reduces traffic on bottleneck links and inter-domain links, and improves the performance of ISP and P2P applications simultaneously. Furthermore, BFP performs steadily in various topologies and swarm sizes, which proves its scalability and deployability.

Our contribution is three-fold: (1) We explore the limitation of P4P in the context of network applications having bottleneck links theoretically. (2) With link utility function as the ISP optimization objective, we propose BFP to accommodate P4P to the scenario having bottleneck links and demonstrate its validity. (3) We verify the improved performance of BFP with the simulation platform, and prove that it is scalable, feasible, and easy to deploy at the same time.

2 The disadvantages of P4P

2.1 Overview of P4P

P4P [14] is a novel solution to the coordination of ISP and P2P applications proposed by H. Xie et al. It is so remarkable that the Distributed Computing Industry Association (DCIA) has meticulously taken up studying P4P and promote the deployment of P4P mechanisms.

In the P4P model, each ISP maintains an iTracker for its network, that has multiple interfaces for ISPs to communicate with P2P applications with respect to the following: (a) static network policy, (b) P-distance mirroring network policy and status, and (c) network capacity. The main interfaces are shown in Figure 2 in [5], where the P-distance interface is the core of P4P. Through P-distance, an ISP can communicate to the P2P applications the current cost of its intra-domain and inter-domain links. P-distance reflects network preference and status with respect to application cost. The main algorithm of P-distance is the min-max link utilization with a distributed algorithm which is designed to solve the optimization problem as follows:

$$\min_{\forall k: t^k \in T^k} \max_{e \in E} \left(b_e + \sum_k \sum_i \sum_j t^k_{ij} I_e(i,j) \right) \Big/ c_e, \tag{1}$$

where b_e is the background traffic (i.e. traffic that P4P can't control), c_e is the capacity of link e, $I_e(i,j)$ is the indicator link e on the route from PID$-i$ to PID$-j$, and T^k denotes the set of all feasible traffic solutions on the basis of the demand and the property of the P2P applications session k. Also, $t^k = (t^k_{ij})$, where t^k_{ij} denotes the traffic from PID$-i$ to PID$-j$ in the P2P session k.

The above-mentioned PID (opaque ID) is a virtual network node. There are many types of PIDs, one of which is an aggregation node, i.e. a PID represents a set of nodes. In fact, a PID can also represent Point of Presence(PoP), or a set of nodes with the same congestion state. In this study, each PID represents an aggregation node.

In particular, T^k is made up of t^k that satisfies the constraint condition as follows,

$$\sum_{j:j\neq i} t^k_{ij} \leqslant u^k_i, \quad \forall i, \tag{2a}$$

$$\sum_{j:j\neq i} t^k_{ji} \leqslant d^k_i, \quad \forall i, \tag{2b}$$

$$t^k_{ij} \geqslant 0, \quad \forall i \neq j, \tag{2c}$$

$$t^k_{ij} \geqslant \rho^k_{ij} \sum_{j'\neq i} t^k_{ij'}, \quad \forall i,\ j \neq i, \tag{2d}$$

$$\sum_i \sum_{j\neq i} t^k_{ij} \geqslant \beta * \mathrm{OPT}, \tag{2e}$$

where u^k_i denotes the aggregation upload capacity from PID$-i$ to other PIDs in session k, and d^k_i denotes the aggregation download capacity from other PIDs to PID$-i$ in session k. ρ^k_{ij} is the lower bound on the percentage of traffic from PID$-i$ to PID$-j$ among all the traffic from PID$-i$ to other PIDs. Note that $0 < \rho^k_{ij} < 1$ and $\sum_{j\neq i} \rho^k_{ij} < 1, \forall i$. β is the efficiency factor that can be configured particularly to

P2P applications in engineering. The OPT in (2e) is the lower bound of P2P performance. Because the cooperation of ISPs and P2P applications aims at improving the dual performance such that cooperation does not compromise the performance of P2P applications. Hence, in general, OPT can be set as the optimal value in the independent optimization of P2P applications. Typically, it can be set as follows:

$$\mathrm{OPT} = \underset{t\in T^k}{\text{maximize}} \sum_i \sum_{j\neq i} t_{ij}^k, \tag{3}$$

i.e. P2P aims at matching download and upload.

Suppose that $t_e^k = \sum_i \sum_j t_{ij}^k I_e(i,j)$, i.e. the total traffic produced by P2P in link e, then the optimization problem (1) is equivalent to:

$$\underset{\alpha, t^k\in T^k,\forall k}{\text{minimize}} \ \ \alpha, \tag{4a}$$

$$\text{subject to} \quad b_e + \sum_k t_e^k \leqslant \alpha c_e, \forall e \in E. \tag{4b}$$

The Lagrange dual function of (4) is as follow:

$$D(p) = \min_{\alpha,\forall t^k\in T^k,k} \sum_e p_e(b_e + \sum_k t_e^k) + (1 - \sum_e p_e c_e)\alpha.$$

To make $D(p)$ finite, the coefficient of α should be zero. i.e.

$$\sum_e p_e c_e = 1.$$

Then

$$D(p) = \sum_e p_e b_e + \sum_k \min_{t^k\in T^k} \sum_e p_e t_e^k. \tag{5}$$

Its dual problem is

$$\underset{p\geqslant 0}{\text{maximize}} \ \ D(p) \text{ subject to } \sum_e p_e c_e = 1. \tag{6}$$

This dual problem can be decomposed into independent subproblems on different sessions of applications with a distributed algorithm, i.e.

$$\underset{t^k\in T^k}{\text{minimize}} \sum_i \sum_{j\neq i} p_{ij} t_{ij}^k. \tag{7}$$

The aforementioned solution is the interactive optimization algorithm between ISPs and a P2P applications, i.e. the P2P application solves the subproblem (7) independently and delivers the optimal result $\overline{t}^k$ to iTracker, after which iTracker solves the master problem (6) to update p_e.

Assumption 1. In the following analysis, we suppose that there exists $\tilde{t}^k \in \tilde{T}^k$ that makes $b_e + \sum_k \tilde{t}_e^k < c_e, \forall e \in E$, i.e. there exists feasible flow solution $\tilde{t}^k$ that makes the constraint on the link capacity strictly feasible.

2.2 Properties of P-distance in P4P

Theorem 1. Suppose that $\{\tilde{t}_e^k\}$ is the solution to (4), and $\{\tilde{p}_e\}$ is the solution to (6). Then there exists at least one link e whose link utilization is maximal and its corresponding $\tilde{p}_e > 0$. The $\tilde{p}_e$ whose corresponding link utilization does not achieve maximum is 0.

Proof. Eq. (4) is an instance of convex programming. And according to assumption A, we know that the Slater constraint quality is true; hence, the strong dual theory holds. As a result, the solution of (4) and of its dual problem (6) satisfy the following:

$$b_e + \sum_k \tilde{t}_e^k \leqslant \tilde{\alpha} c_e, \quad \forall e \in E, \tag{8a}$$

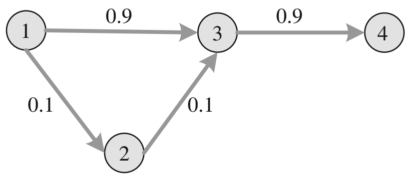

Figure 1 Topology of network having bottleneck links.

$$1-\sum_e \tilde{p}_e c_e = 0, \tag{8b}$$

$$\tilde{p}_e \geqslant 0, \quad \forall e \in E, \tag{8c}$$

$$\tilde{p}_e \left(b_e + \sum_k \tilde{t}_e^k - \tilde{\alpha} c_e \right) = 0, \quad \forall e \in E, \tag{8d}$$

where (8a) is the original feasible condition, (8b) and (8c) are the dual feasible conditions, and (8d) is the complementary condition. Furthermore, because of the optimality of $\{\tilde{t}_e^k\}$ to the problem (4), it holds that

$$\tilde{\alpha} = \max_{e\in E} \left(b_e + \sum_k \tilde{t}_e^k \right) \bigg/ c_e.$$

Based on (8d), we know that all $\tilde{p}_e$s whose corresponding links do not achieve the maximum utilization equal to 0, and $\tilde{p}_e$s whose corresponding links achieve the maximum are equal to or greater than 0. Moreover, by (8b), we know that there exists at least one $\tilde{p}_e$ that is not equal to 0.

As we can see from the above theorem, all prices of links that do not achieve the most congested state are equal to 0, i.e. for a flow, the link price of each non-most-congested link is equal to that of another such link. This property of the multiplier makes Maximum Link Utilization (MLU) is invalid when dealing with networks having bottleneck links.

To make the above-mentioned theory easier to understand, we illustrate the limitation as an toy example in Figure 1. Suppose that the capacity of each link is 1. The traffic between node 1 and node 3 is 1, and the traffic between node 3 and node 4 is 0.9. If we set MLU as the optimization objective of ISP, to achieve the optimal value, the traffic on link (1, 3) will be 0.9 and the traffic on path (1, 2) will be 0.1. This leads to a situation where of the two paths between node 1 and node 3, path (1, 3) is very congested and the other path (1, 2, 3) is quite idle.

3 Improved-P4P: an improved cooperation algorithm

We introduce a link utility function as the ISP optimization objective and verify that the new objective can utilize the network resource better by carrying out a theoretical analysis and experimental simulations. In this section, we propose a cooperative algorithm of ISPs and P2P applications with a new link utility function, and analyze some issues with this objective function.

We follow the methods introduced in congestion control [13] to adopt new link utility function. Consider that

$$\underset{\{s_e\},t^k\in T^k,\forall k}{\text{maximize}} \quad \sum_e v_\beta(s_e), \tag{9a}$$

$$\text{subject to} \quad s_e \leqslant c_e - \sum_k t_e^k - b_e, \quad \forall e \in E, \tag{9b}$$

where s_e is the free capacity of link e and $v_\beta(s_e)$ is an increasing concave function. In this paper, we adopt the form as stated in [13], i.e.

$$v_\beta(s_e) = \begin{cases} \log(s_e), & \beta = 1 \\ \frac{s_e^{1-\beta}}{1-\beta}, & \beta \neq 1. \end{cases}$$

The Lagrange dual function of (9) is as follows:

$$D(p) = \max_{\{s_e\},t^k\in T^k,\forall k} \sum_e \left(v_\beta(s_e) - \sum_e p_e\left(s_e - c_e + \sum_k t_e^k + b_e\right)\right)$$
$$= \max_{s_e} \sum_e \big(v_\beta(s_e) - p_e s_e\big) + \sum_k \min_{t^k\in T^k} \sum_e p_e t_e^k + \sum_e p_e\big(c_e - b_e\big). \quad (10)$$

The dual problem of (9) is as follows:

$$\min_{p\geqslant 0} \ D(p). \quad (11)$$

Considering that $D(p)$ is undifferentiable and (11) cannot be solved with the gradient method directly, we solve the problem by the subgradient method. We can obtain the subgradient of $D(p)$ from [15],

$$\zeta_e = c_e - b_e - \tilde{s}_e - \sum_k \tilde{t}_e^k, \quad \forall e\in E, \quad (12)$$

where $\tilde{s}_e$ is the solution of

$$\underset{c_e-b_e\geqslant s_e}{\text{maximize}} \big(v_\beta(s_e) - p_e s_e\big), \quad \forall e\in E, \quad (13)$$

and $\{\tilde{t}_e^k\}$ is the solution of

$$\underset{t^k\in T^k}{\text{minimize}} \sum_e p_e t_e^k, \ \ \forall k \qquad \sum_e p_e t_e^k, \ \ \forall k. \quad (14)$$

On the basis of the subgradient projection method, p_e can be updated as follows,

$$p_e(\tau+1) = \begin{cases} p_e(\tau) - \mu(\tau)\zeta_e(\tau), & p_e(\tau) > \mu(\tau)\zeta_e(\tau), \\ 0, & p_e(\tau) \leqslant \mu(\tau)\zeta_e(\tau), \end{cases} \quad (15)$$

where ζ_e is the subgradient and $\mu(\tau)$ is the step-size parameter. Theoretically, the step-size parameter $\mu(\tau)$ is of vital importance to the convergence of this algorithm. However, practically, owing to the continuous evolving of the network and the P2P applications, we can set the step-size parameter as a constant.

After solving (11) by the subgradient method, we obtain the distributed algorithm to solve (9), which is the interactive optimization algorithm of ISPs and P2P applications. In other words, P2P applications and ISPs can solve the subproblems (14) and (13) independently at first and then transfer the optimal solution $\tilde{t}^k$ and $\tilde{s}_e$ to iTracker. Next, iTracker will update p_e according to (15).

With respect to $v_\beta(s_e) = \log(s_e)$, we can provide the explicit solution of (13) and the engineering significance of its dual variable which is also the link price as follows:

Theorem 2. When $\beta = 1$, the explicit solution to the subproblem (13) is

$$\tilde{s}_e = \begin{cases} \frac{1}{p_e}, & p_e \geqslant \frac{1}{c_e-b_e}, \\ c_e - b_e, & 0 \leqslant p_e < \frac{1}{c_e-b_e}. \end{cases}$$

Proof. Suppose that the derivative of objective function of problem (13) is $U'(s_e)$, then $U'(s_e) = \frac{1}{s_e} - p_e$, and the stationary point of the derived function is $s_e = \frac{1}{p_e}$. When $p_e \geqslant \frac{1}{c_e-b_e}$, the stationary point of the objective function's derivative function $U'(s_e)$, i.e. $s_e = \frac{1}{p_e}$ satisfies the constraint condition of (13), i.e. $0 < s_e \leqslant c_e - b_e$. As a result, it is the optimal solution of (13). When $0 < p_e < \frac{1}{c_e-b_e}$, for all feasible solutions of (13), i.e. for all s_e with $0 < s_e \leqslant c_e - b_e$, it can be checked that $U'(s_e) > 0$. So $U(s_e)$ is increasing in $[0, c_e - b_e]$. Therefore, $s_e = c_e - b_e$ is the optimal solution.

Theorem 3. Suppose that $\{\tilde{t}_e^k\}$ is the solution of (9) and $\{\tilde{p}_e\}$ is the solution of the dual problem (11). Then $\forall e \in E$, and when $\tilde{s}_e = c_e - b_e$,

$$\tilde{p}_e \in \left[0, \frac{1}{c_e - b_e}\right),$$

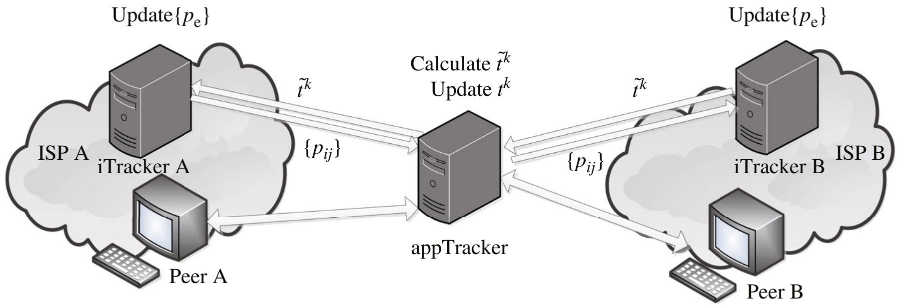

Figure 2 Interaction between iTracker and applications.

when $\tilde{s}_e < c_e - b_e$,

$$\tilde{p}_e = \frac{1}{c_e - \sum_k \tilde{t}_e^k - b_e}.$$

Proof. We know that (9) is a convex programming problem, and with assumption A, we can get that the Slater constraint specification is true. Hence, the strong dual theorem holds, and there exist the following facts,

$$c_e - b_e - \sum_k \tilde{t}_e^k \geqslant \tilde{s}_e, \quad \forall e \in E, \tag{16a}$$

$$\tilde{p}_e \left(\tilde{s}_e - c_e + \sum_k \tilde{t}_e^k + b_e \right) = 0, \quad \forall e \in E, \tag{16b}$$

where (16a) is the original feasible condition and (16b) is the complementary condition. From the fact that $\log(s_e)$ is an increasing function and $\tilde{s}_e$ is the optimal solution, we can get (16a) holds equally, i.e. $\tilde{s}_e = c_e - b_e - \sum_k \tilde{t}_e^k, \forall e \in E$. When a P2P flow does not use link e, i.e. $\tilde{s}_e = c_e - b_e$, we know that the link price is as follow:

$$p_e \in \left[0, \frac{1}{c_e - b_e}\right).$$

When P2P flows use link e, i.e. $\tilde{s}_e < c_e - b_e$, based on theorem 3, we can get,

$$\tilde{s}_e = \frac{1}{\tilde{p}_e}. \tag{17}$$

After substituting $\tilde{s}_e = c_e - b_e - \sum_k \tilde{t}_e^k, \forall e \in E$ into (17), we can get

$$\tilde{p}_e = \frac{1}{c_e - \sum_k \tilde{t}_e^k - b_e}, \quad \forall e \in E. \tag{18}$$

As we can see from Theorem 3, for any link e, the larger its free link capacity is, the lower its link price ($\tilde{p}_e$) will be; the smaller its free link capacity is, the higher its link price ($\tilde{p}_e$) will be. This makes ISPs to control traffic on the non-most-congested links more efficiently when dealing with networks applications having bottleneck links.

Figure 2 shows an example of the interaction between iTracker and P2P applications. In the example, all P2P peers are divided into two networks of which the network providers are ISP A and ISP B. Each ISP runs an iTracker for its own network. Peer A first registers with the appTracker, and the appTracker inquires iTracker A through the interface for p_e and sends $\tilde{t}^k$ to the iTracker A. After obtaining $\tilde{t}^k$, iTracker A gets s_e with Theorem 2, then obtains ξ_e in (12), and updates p_e according to (15) and sends p_e back to the appTracker. As soon as the appTracker obtains p_e, it will decide peer-selection for Peer A and collect the information of peer A for calculating $\tilde{t}^k$ based on (14). As for iTracker B and peer B, they interact with the appTracker in the same way.

4 Simulation methodology

We have built a discrete-event package based on BitTorrent for simulation. We followed the method described in [16] and performed the simulations by implementing the native BitTorrent protocol. We also calculated the traffic on every link in order to estimate the link utilization. Our simulation includes keeping traffic statistics on bottleneck links, inter-domain links, and intra-domain links in P2P, P4P, and BFP applications. Further, we have studied the benefits of P2P applications in P4P and BFP by varying the swarm size.

4.1 Simulator overview

Considering the popularity of BitTorrent in P2P applications, we study the performance of P2P, P4P and BFP applications with BitTorrent.

BitTorrent is a P2P application that enables fast and efficient large-scale file sharing by balancing the upload and download bandwidth of peers. Basically, BitTorrent divides the file into small blocks (32–256 KB typically) and make nodes download the blocks simultaneously. There is at least one seed in a BitTorrent system which has a complete copy of the file and is ready to upload it to other nodes. Once a new node joins the BitTorrent system, it will try to obtain a list of random nodes that are currently in the system from the tracker. Then the new node will try to choose 40 nodes as its neighbors and establish contacts with them. When the number of neighbors is below 20, the node will request the tracker a list of additional nodes that can serve as its neighbors.

It is difficult to study using traces of real torrents [17,18] or just analysis [19,20]. Therefore, we use a simulation-based method to test the performance of P2P, P4P and BFP applications. Such an approach provides the flexibility of carefully managing the configuration parameters, which is quite difficult or even impossible for live Internet measurement techniques. Although certain interactions specific to a real deployment will be missed, we still believe that the abstraction is rich enough to explore the performance of P2P, P4P and BFP applications.

We follow the method described in [16], and verify the performance of BFP on the basis of the software[1)] originally developed by Bharambe et al. The software was examined under a realistic workload which is derived from the Redhat 9 distribution torrent, and proves its feasibility and validity.

Based on discrete-event simulation (DES), our simulator models various peer activities (joins, leaves, block exchanges, etc). In the model, each node associates a download link and an upload link bandwidth, with which the simulator can appropriately schedule the process of exchanging blocks. The calculation of the delay takes into consideration the number of flows which are sharing the upload and download link at either end.

4.2 Assumptions

(1) We have ignored the propagation delay because it relates only to small control packets. We believe that this simplification has negligible impact on the conclusion because of the following: (a) The download time depends on the transmission time of the packets. (b) In practice, the pipelined processing mechanism of BitTorrent reduces most propagation delay of packets.

(2) We have followed the method proposed in [12] and assumed that all TCP sessions share link capacity equally in the stable state. We do not simulate at the packet level, because it is very difficult to simulate a large number of packets when hundreds of peers share a very large file. When TCP sessions join or leave the network, we recalculate the throughput of the sessions being influenced. Although this simplification means that we cannot simulate TCP exceptions, we believe that the relatively long connection reduces the impact. Notice that though BitTorrent has short single packet, the packets can be as long as possible after pipeline process.

(3) All nodes join the network in a very short time, and this is a critical condition, with which we can test the performance of P2P, P4P, and BFP applications effectively.

1) Bharambe A, Herley C, Padmanabhan V N. Microsoft research simulator for the BitTorrent protocol.

Table 1 Network topology

Network	Region	Aggregation level	♯Nodes	♯Links	Link capacity
Abilene	US	PoP-level	11	28	10 Gbps
CERNET2	China	PoP-level	20	44	2.5 Gbps or 10 Gbps

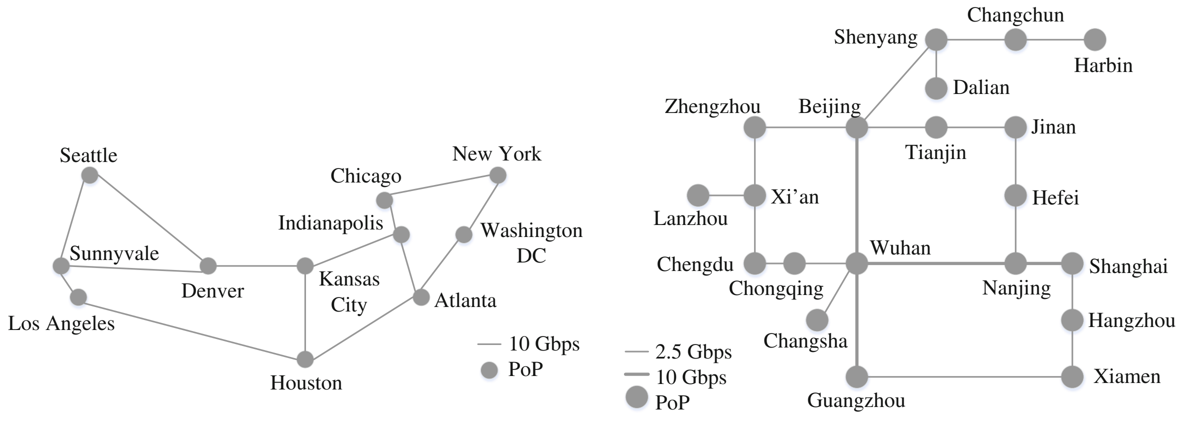

Figure 3 Network topology. (a) Abilene; (b) CERNET2.

(4) After finishing download, all peers leave the network immediately. Considering that some ISPs may charge the upload traffic, most peers choose to leave the network after finishing download. In addition, the number of seeds is considered to be constant in the simulations. If peers stay in the network after they finish download, the model will become too complicated to simulate.

4.3 Detailed parameters

Some detailed parameters are as follows,
1. Size of block: 256 kB;
2. Number of seed: 1;
3. Upstream access link capacity of a seed: 5 MBps;
4. Upstream access link capacity of peers: uniform distribution between 550 kBps to 1000 kBps;
5. Downstream access link capacity of peers: twice the upstream capacity.

We use PoP-level topologies of Abilene and CERNET2. Figure 3 shows these topologies, and Table 1 shows their detailed parameters. Notice that in Figure 3(a), the points represent PoPs in Abilene, and the lines represent links between PoPs with a capacity of 10 Gbps on both directions. Similarly, in Figure3(b), the points represent PoP in CERNET2, the finer lines represents links between PoPs with a capacity of 2.5 Gbps on both directions, and the thicker lines represent links between PoPs with a capacity of 10 Gbps on both directions.

4.4 Neighbor selection policy

P4P and BFP improve the neighbor selection policy with the interaction of ISPs and P2P applications. For a peer,

1. Neighbor selection within PID: appTracker selects a certain number of neighbors within peer i's PID; the ratio of this number to the number of peer i's neighbors should be limited below a certain percentage.

2. Neighbor selection between PIDs: Suppose that the link price between PID i and PID j is p_{ij}. For each $i \neq j$, if $p_{ij} \neq 0$, $w_{ij} = 1/p_{ij}$, else we can set w_{ij} to be a very large value.

$$W_{ij} = \frac{w_{ij}}{\sum_{i \neq j} w_{ij}}. \tag{19}$$

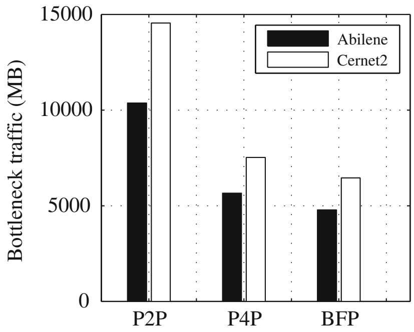

Figure 4 Variance of traffic (swarm size = 600).

For peers in PID i, after finishing neighbor selection within PID, they will select a certain percentage of neighbors from other PIDs according to W_{ij}. For robustness, concave function $f(x)$ can be introduced to enlarge the relative weight of W_{ij}. In this paper, we apply $f(x) = \sqrt[6]{x}$.

4.5 Performance metrics

We consider the following performance metrics:

1. Completion time: It measures the performance of BitTorrent, including various periods that all peers need to finish downloading and that each single peer needs to finish downloading.

2. P2P bottleneck traffic: It is the total P2P traffic on a link that achieves the maximum link utilization, which examines working of P2P, P4P and BFP applications in the context of topologies with bottleneck traffic.

3. Inter-domain traffic: It is the total P2P traffic on links between Autonomous Systems (ASes), which is used only in inter-domain settings. This metric examines the performance of P2P, P4P and BFP in relieving the burden on ISPs.

5 Simulation results

We have detected bottleneck link traffic and traffic on links between PoPs in different topologies (Abilene and CERNET2), and determined statistics of peer's completion time for different swarm sizes in different topologies. Next, we have obtained the statistics of traffic on links among ASes in inter-domain settings.

5.1 Simulation within ASes

In this section, we consider a case in which all peers share a 250-MB file, and discuss the performance of P2P, P4P, and BFP applications for different swarm sizes and network topologies within ASes.

As Figure 4 shows, when the swarm size is 600, compared with the native BitTorrent based on P2P, BitTorrent integrated with BFP can reduce the bottleneck traffic substantially. Moreover, to some degree, BFP can reduce P4P's bottleneck. In particular, in the case of Abilene, the bottleneck traffic produced by the native BitTorrent is 2.17 times that of the BitTorrent integrated with BFP, and in the case of CERNET2, this number is 2.25. In the case of Abilene, BitTorrent integrated with BFP can reduce the bottleneck traffic of BitTorrent integrated with P4P by 15.21%, and in the case of CERNET2, this ratio is 14.23%.

We can conclude from the above result that compared with P2P and P4P, BFP can reduce the bottleneck traffic considerably and better protect the bottleneck.

Figure 5 shows the variation in bottleneck link utilization with time in the cases of Abilene and CERNET2 when the swarm size is 600. Compared with the native BitTorrent or BitTorrent integrated with P4P, BitTorrent integrated with BFP can considerably reduce bottleneck link utilization and decrease the duration of high-level bottleneck traffic. Incidentally, in the case of Abilene, the bottleneck link utilization of BitTorrent integrated with BFP is approximately 60% of the native BitTorrent, and the

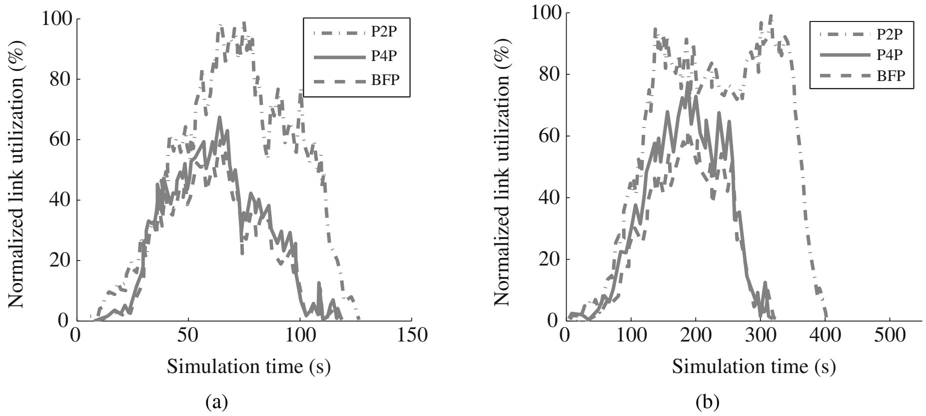

Figure 5 Nomalized bottleneck link utilization (swarm size = 600). (a) Abilene; (b) CERNET2.

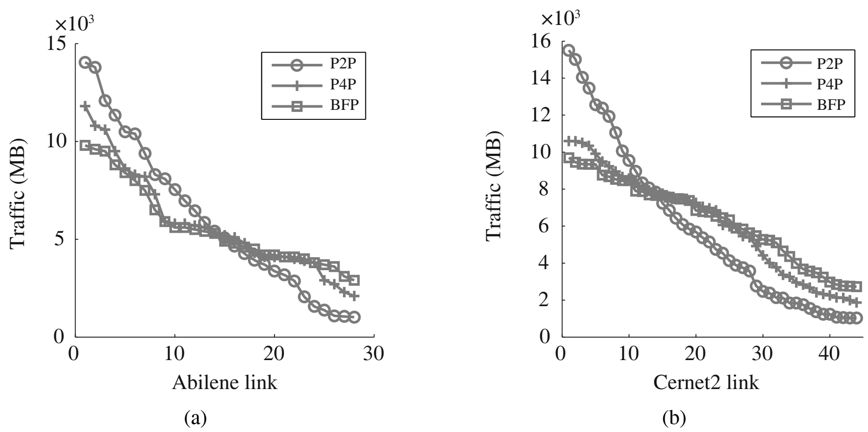

Figure 6 Variance of traffic (swarm size = 600). (a) Abilene; (b) CERNET2.

duration of the high-level bottleneck traffic of BFP is approximately half of P2P. Moreover, BitTorrent integrated with BFP can reduce the bottleneck traffic by 14% as compared to that in the case of the BitTorrent integrated with P4P. In CERNET2, similarly, the bottleneck link utilization of BFP is 65% of that of P2P, and the duration of the high-level bottleneck traffic is approximately 40% of P2P. BFP can reduce the bottleneck traffic by 12% as compared to that in the case of P4P.

In conclusion, BFP can reduce bottleneck link utilization and the duration of high-level bottleneck traffic. Undoubtedly, BFP can relieve the pressure on bottleneck links.

Figure 6 shows the traffic on each link (arranged in decreasing order of the amount of traffic) between the PoP of the BitTorrent system integrated with P2P, P4P and BFP in the case of Abilene when the swarm size is 600. Concretely, in the case of Abilene, the standard deviations of the link traffic in the three different systems are $S_{\rm P2P} = 3968$, $S_{\rm P4P} = 2637$ and $S_{\rm BFP} = 2073$. While in the case of CERNET2, the standard deviations of the link traffic in the three different systems are $S_{\rm P2P} = 4356$, $S_{\rm P4P} = 2792$ and $S_{\rm BFP} = 2165$.

Compared with P2P and P4P, BFP makes the traffic on every link more even, reduces the traffic on the bottleneck links, and performs steadily whether in Abilene or in CERNET2, proving its good scalability.

Figure 7 shows how the average normalized completion time varies with the swarm size. The error bars in the figure represent the standard variations of repeated experiments. The normalized completion time refers to the normalized value based on the maximum average download time of the peers in the native BitTorrent system. In particular, in Abilene, BitTorrent integrated with BFP reduces the completion time by approximately 6% compared with BitTorrent integrated with P4P, and by 18% compared with native BitTorrent. In CERNET2, BFP speeds up by approximately 19% with P2P and 8% with P4P.

Figure 8(a) shows that in Abilene, BitTorrents integrated with BFP enhances the download speed by approximately 20% compared with the native BitTorrent, and 5% compared with BitTorrent integrated

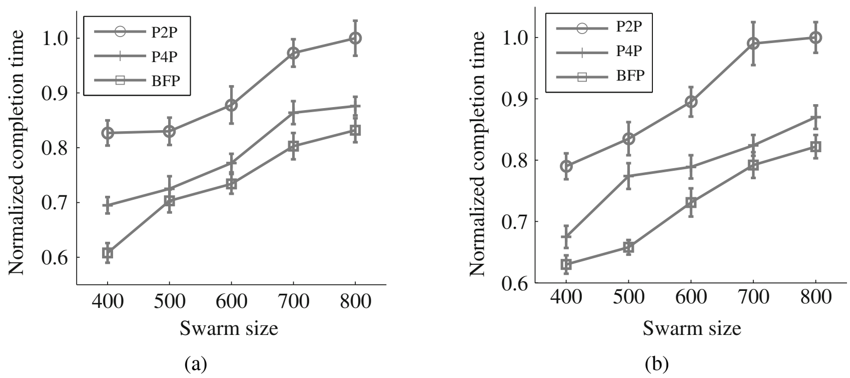

Figure 7 Average completion time. (a) Abilene; (b) CERNET2.

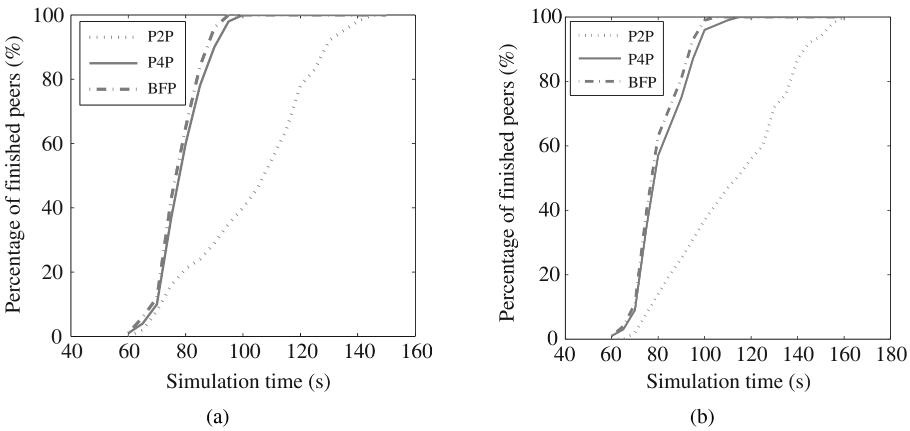

Figure 8 CDFs of completion time (Swarm Size = 600). (a) Abilene; (b) CERNET2.

with P4P. To analyze the performance of BFP in different topologies, we also keep statistics of Cumulative Distribution Functions (CDFs) of completion time in CERNET2, as shown in Figure 8(b). It draws almost the same conclusion with that in Abilene.

In conclusion, in intra-domain settings, in the cases of both Abilene and CERNET2 topologies, compared with P2P and P4P, BFP can improve the performance of ISPs and P2P applications simultaneously. Compared with P4P, BFP can reduce the bottleneck link utilization and therefore relieve the pressure on networks with bottlenecks and improve the performance of the P2P applications. We believe that Apparently, relieving the pressure on topologies with bottlenecks is crucial when it comes to extreme circumstances. The consistency of the simulation results obtained in the cases of Abilene and CERNET2 demonstrates that BFP has a steady performance in different topologies, which then proves that BFP is feasible for deployment.

5.2 Simulation among ASes

In this section, we will study the performance of BFP under the condition that peers are located in different ASes, and we also consider a case in which all 600 peers share a 250-MB file. In our simulation, we select two links in the case of Abilene as the inter-domain links. One link is from Indianapolis to Kansas City, and the other is from Atlanta to Houston. As a result, the Abilene topology is divided into two parts: one with 5 PoPs is close to the east coast, while the other with 6 PoPs is close to the west coast. The two parts can be considered as two "virtual" ISPs, and we suppose that the two parts make a deal that the link price of the both be the same.

Figure 9(a) shows the traffic of the inter-domain links in the BitTorrent system integrated with P2P, P4P, and BFP in Abilene. On the first inter-domain link (Indianapolis to Kansas City), the inter-domain traffic of the BitTorrent integrated with BFP is 58.33% of the native BitTorrent and 87.50% of the BitTorrent integrated with P4P. On the second inter-domain link (Atlanta to Houston), these

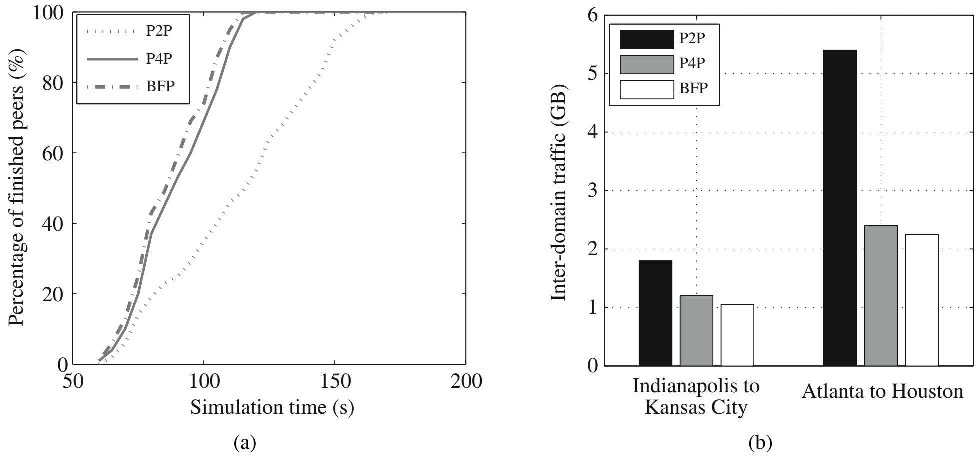

Figure 9 Inter-domain simulation in Abilene. (a) CDFs of completion time; (b) inter-domain traffic.

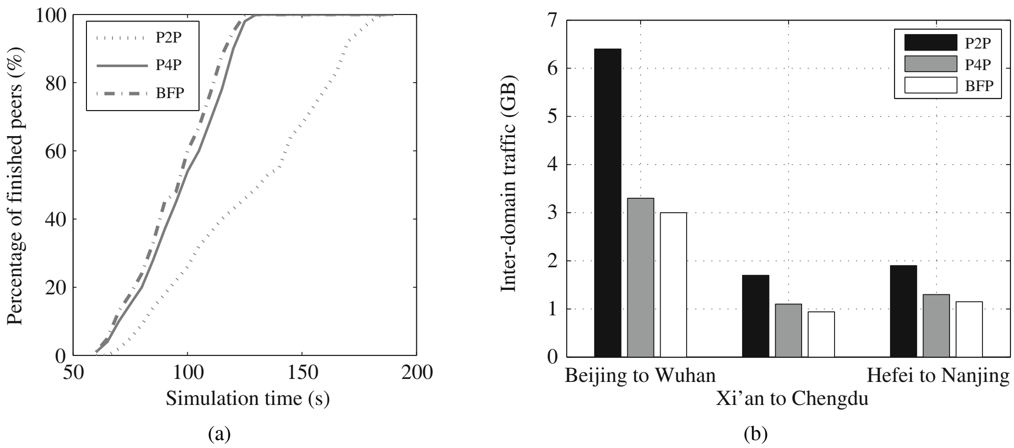

Figure 10 Inter-domain simulation in CERNET2. (a) CDFs of completion time; (b) inter-domain traffic.

two percentages are 41.67% and 93.75% respectively. Hence, we can conclude that BFP can reduce the inter-domain traffic compared with P2P and P4P and decrease the ISP's cost in Abilene.

Figure 9(b) shows the percentage of peers that finish download in Abilene. In the inter-domain settings, the BitTorrent integrated with BFP reduces the download time by approximately 30% compared with the native BitTorrent based on P2P, and by approximately 4% with BitTorrent integrated with P4P.

To study the performance of BFP in different topologies, we also conduct the above simulations in CERNET2. Concretely, we divide the topology of CERNET2 into two parts which are connected with three links. The first link is from Beijing to Wuhan, the second is from Xian to Chengdu, and the third is from Hefei to Nanjing. Therefore, the northern part has 11 PoPs and the southern one has 9 PoPs.

Figure 10(a) shows the traffic of the inter-domain links in the BitTorrent system integrated with P2P, P4P, and BFP in CERNET2. On the first inter-domain link (Beijing to Wuhan), the inter-domain traffic of the BitTorrent integrated with BFP is 46.88% of the native BitTorrent and 90.91% of that the BitTorrent integrated with P4P. On the second inter-domain link, these two percentages become 55.29% and 85.45%, respectively. While on the third inter-domain link, these two percentages are 60.53% and 88.46%, respectively. We can also conclude that BFP can decrease the inter-domain traffic compared with P2P and P4P and decrease the ISP's cost in CERNET2.

Figure 10(b) shows the percentage of peers that finish download. Similarly, BFP improves the download speed by approximately 34% with P2P and 4% by P4P.

In brief, in the inter-domain settings, whether in Abilene or in CERNET2 topology, BFP can improve the performance of both ISP and P2P applications compared with P2P and P4P. In addition, compared to P4P, BFP can reduce the inter-domain traffic and therefore decrease ISP's cost without deteriorating the performance of P2P applications.

6 Related work

Several approaches have been recently proposed to incorporate ISPs with P2P applications.

Saleh et al. [21] analyzed the popularity of P2P objects in different ASes and found that they can be modeled by a Mandelbrot-Zipf distribution. Based on this, they developed a caching algorithm in order to reduce the cost incurred by ISP and alleviate the backbone load. The algorithm is based on object segmentation, partial admission and eviction of objects.

Aggarwal et al. [11] studied the advantages and disadvantages of P2P and proposed a middle server called "oracle", by which ISPs can provide a neighbor selection policy for P2P users. A peer sends its list of potential neighbors to "oracle", which will sort all the possible neighbors according to certain criteria, such as the nearest principle and the link bandwidth. The sorted neighbor list will guide peers to select neighbors and improve the P2P performance. At the same time, the ISPs can effectively manage massive P2P traffic with this mechanism, assuring that the traffic does not pass across them and that it is directed to the right path. With the abovementioned mechanism, ISPs can provide a better network service for their users.

Jiang et al. [22] developed a cooperative model of Content Providers (CPs) and ISP, which aimed at co-optimization by sharing control between ISPs and CPs. This study analyzed the optimality of this model by using the game theory and compared this model with the traditional model by means of simulations. The simulation results showed the advantages of the sharing model under circumstances with different congestion levels, and pointed that under some conditions, if the complete network information is shared without any co-optimization control, the co-benefits of CPs and ISPs may be lower than before the sharing of the complete network information.

7 Conclusion

After analyzing P-distance in the P4P model and the relationship between P-distance and the link traffic, it is concluded that for a network application having bottlenecks, the application of MLU as the optimization objective of P4P cooperative does not achieve optimized performance. We propose a bottleneck-free model for P4P and BFP, by introducing a link utility function as the objective function. The relationship between the link price and the traffic in this model demonstrates that BFP makes the traffic more homogeneous. A considerable number of BitTorrent-based simulations on P2P, P4P, and BFP have been done in order to verify whether BFP achieves the interactive control over the network traffic. In addition, the simulations show that BFP benefits ISPs with a reduction of the bottleneck traffic and the inter-domain traffic, and improves the performance of P2P applications. We believe that easing the pressure on networks with bottleneck-links has great significance especially in the extreme circumstances. Further, BFP performs stably with various swarm sizes and network topologies, proving its feasibility, scalability, and effectiveness.

For future work, we plan to extend the simulation to larger sizes of swarms, seeds, network topologies. With some open platforms like PlanetLab, we also plan to conduct real experiments on the Internet to demonstrate the effectiveness, feasibility, and scalability of BFP.

Acknowledgements

This work was supported by National Natural Science Foundation of China (Grant Nos. 61161140454, 61172060, 60873253).

References

1 Karagiannis T, Broido A, Brownlee A, et al. Is P2P dying or just hiding. In: Proceedings of Global Telecommunications Conference, Dallas, 2004. 1532–1538

2 Steinmetz R, Wehrle K. Peer-to-Peer Systems and Applications, Heidelberg: Springer, 2002

3 Keralapura R, Taft N, Chuah C, et al. Can ISPs take the heat from overlay networks. In: Proceedings of HotNets III, San Diego, 2004. 8–15

4 Aggarwal V, Bender S, Feldmann A, et al. Methodology for estimating network distances of gnutella neighbors. In: Proceedings of the Workshop on Algorithms and Protocols for Efficient Peer-to-Peer Applications, Ulm, 2004. 219–223

5 Xie H, Yang Y R, Krishnamurthy A, et al. P4P: provider portal for applications. In: Proceedings of the ACM SIGCOMM 2008 Conference on Data Communication, Seattle, 2008. 351–362

6 Seetharaman S, Ammar M. Characterizing and mitigating inter-domain policy violations in overlay routes. In: Proceedings of the IEEE International Conference on Network Protocols, Santa Barbara, 2006. 259–268

7 Karagiannis T, Rodriguez P, Papagiannaki K. Should Internet service providers fear peer-assisted content distribution. In: Proceedings of the 5th ACM SIGCOMM Conference on Internet Measurement, Berkeley, 2005. 63–76

8 Zhang G, Tang M, Cheng S, et al. P2P traffic optimization. Sci China Inf Sci, 2012, 55: 1475–1492

9 Seibert J, Torres R, Mellia M, et al. The Internet-wide impact of P2P trafic localization on ISP profitability. IEEE/ACM Trans Netw, 2012, 20: 1910–1923

10 Blond S L, Legout A, Dabbous W. Pushing BitTorrent locality to the limit. Comput Netw, 2011, 55: 541–557

11 Aggarwal V, Feldmann A, Scheideler C. Can ISPs and P2P users cooperate for improved performance. ACM SIGCOMM Comput Commun Rev, 2007, 37: 29–40

12 Bindal R, Cao P, Chan W, et al. Improving traffic locality in BitTorrent via biased neighbor selection. In: Proceedings of IEEE International Conference on Distributed Computing Systems, Lisboa, 2006. 66

13 Mo J, Walrand J. Fair end-to-end window-based congestion control. IEEE/ACM Trans Netw, 2000, 8: 556–567

14 Xie H, Krishnamurthy A, Silberschatz A, et al. P4P: explicit communications for cooperative control between P2P and network providers. P4PWG Whitepaper. 2007

15 Palomar D P, Chiang M. Alternative distributed algorithms for network utility maximization: framework and applications. IEEE Trans Automat Control, 2007, 52: 2254–2269

16 Bharambe A R, Herley C, Padmanabhan V. Analyzing and improving a BitTorrent network's performance mechanisms. In: Proceedings of IEEE International Conference on Computer Communications, Arlington, 2006. 1–12

17 Izal M, Urvoy-Keller G, Biersack E W, et al. Dissecting BitTorrent: five months in a Torrent's lifetime. In: Passive and Active Measurement Workshop, Antibes Juan-les-Pins, 2004. 1–11

18 Pouwelse J A, Garbacki P, Epema D H J, et al. A measurement study of the BitTorrent peer-to-peer file-sharing system. Technical Report PDS-2004-003. Delft University of Technology, The Netherlands, 2004

19 Qiu D, Srikant R. Modeling and performance analysis of Bittorrent-like peer-to-peer networks. In: Proceedings of SIGCOMM, Portland, 2004. 367–378

20 Guo L, Chen S, Xiao Z, et al. Measurements, analysis, and modeling of BitTorrent-like systems. In: Proceedings of the ACM SIGCOMM Conference on Internet Measurement, Berkeley, 2005. 35–48

21 Saleh O, Hefeeda M. Modeling and caching of peer-to-peer traffic. In: Proceedings of the IEEE International Conference on Network Protocols, Santa Barbara, 2006. 249–258

22 Jiang W, Zhang R, Rexford J, et al. Cooperative content distribution and traffic engineering in an ISP network. In: Proceedings of the International Joint Conference on Measurement and Modeling of Computer Systems, Seattle, 2009. 239–250