LBMP: A Logarithm-Barrier-Based Multipath Protocol for Internet Traffic Management

Ke Xu, *Senior Member*, *IEEE*, Hongying Liu, Jiangchuan Liu, *Senior Member*, *IEEE*, and Jixiu Zhang

Abstract—Traffic management is the adaptation of source rates and routing to efficiently utilize network resources. Recently, the complicated interactions between different Internet traffic management modules have been elegantly modeled by distributed primal-dual utility maximization, which sheds new light for developing effective management protocols. For single-path routing with given routes, the dual is a strictly concave network optimization problem. Unfortunately, the general form of multipath utility optimization is not strictly concave, making its solution quite unstable. Decomposition-based techniques like TRaffic-management Using Multipath Protocol (TRUMP) alleviates the instability, but their convergence is not guaranteed, nor is their optimality. They are also inflexible in differentiating the control at different links. In this paper, we address the above issues through a novel logarithm-barrier-based approach. Our approach jointly considers user utility and routing/congestion control. It translates the multipath utility maximization into a sequence of unconstrained optimization problems, with infinite logarithm barriers being deployed at the constraint boundary. We demonstrate that setting up barriers is much simpler than choosing traditional cost functions and, more importantly, it makes optimal solution achievable. We further demonstrate a distributed implementation, together with the design of a practical Logarithm Barrier-based-Multipath Protocol (LBMP). We evaluate the performance of LBMP through both numerical analysis and packet-level simulations. The results show that LBMP achieves high throughput and fast convergence over diverse representative network topologies. Such performance is comparable to TRUMP, and is often better. Moreover, LBMP is flexible in differentiating the control at different links, and its optimality and convergence are theoretically guaranteed.

Index Terms—Traffic management, network utility maximization, multipath routing, logarithm barrier.

1 Introduction

The Internet has evolved into an ultralarge and complex system interconnecting diverse end-hosts and transmission links, with numerous applications running over it. Traffic management thus becomes a critical challenge to the healthy operation of the Internet. To this end, various traffic management tools have been developed, including TCP congestion control at end-hosts, traffic engineering by network operators, and adaptive routing algorithms performed by routers, as shown in Fig. 1. They all target efficient utilization of network resources and quality service to end users. Unfortunately, the collaboration among them is far from being perfect. In particular, network operators regulate traffic through tuning link weights, where the weight-setting problem is indeed NP-hard with no practically optimal solution [6]. The operation is also indirect and is performed over long time scales. The end-hosts, on the other hand, adapt their sending rates in real-time, assuming routing is fixed. This mismatch leads to highly suboptimal outcome with unsatisfactory resource utilization [8], [10].

There have been many efforts toward joint optimizations of these modules. Specifically, the TCP congestion control, together with active queue management, has recently been interpreted as a distributed primal-dual problem that maximizes the aggregate utility, where a user's utility function is (often implicitly) defined by its TCP implementation [12], [16], [4]. For single-path routing with given routes, the dual is a strictly concave network optimization problem. It is, however, known that single-path routing limits the achievable throughput and is vulnerable to link failures. If a flow can be flexibly divided and delivered over multiple paths, higher efficiency and robustness can be expected.

Unfortunately, the general form of the multipath utility optimization is not strictly concave, making its solution quite unstable [22]. Earlier attempts to address the stability issue [14], [22] are mainly theoretical. Recently, decomposition-based practical solutions like TRaffic-management Using Multipath Protocol (TRUMP) [8], [9], [10] have been suggested. Yet, their convergence properties have not been fully proved, nor is their optimality. They are also inflexible in differentiating the control at different links.

The logarithmic barrier method is a powerful optimization method for constrained optimization, in which logarithmic terms are introduced to prevent feasible iterates from moving too close to the boundary of the feasible region [2]. In this paper, we address the above issues through a novel logarithm-barrier-based approach. Our approach jointly considers user utility and routing/congestion control. It

- *K. Xu is with the Department of Computer Science, Tsinghua University, FIT Building 4-104, Beijing 100084, China. E-mail: xuke@tsinghua.edu.cn.*
- *H. Liu is with the School of Mathematics and Systems Science and LMIB, Beijing University of Aeronautics and Astronautics, Beijing 100191, China. E-mail: liuhongying@buaa.edu.cn.*
- *J. Liu is with the School of Computing Science, Simon Fraser University, Burnaby (Metro-Vancouver), British Columbia, Canada. E-mail: jcliu@cs.sfu.ca.*
- *J. Zhang is with the School of Software and Microelectronics, Peking University, Beijing 102600, China. E-mail: zhangjixiu@gmail.com.*

Manuscript received 16 Feb. 2009; revised 1 Sept. 2009; accepted 29 Sept. 2009; published online 30 April 2010.
Recommended for acceptance by J. Lui.
For information on obtaining reprints of this article, please send e-mail to: tpds@computer.org, and reference IEEECS Log Number TPDS-2009-02-0072.
Digital Object Identifier no. 10.1109/TPDS.2010.95.

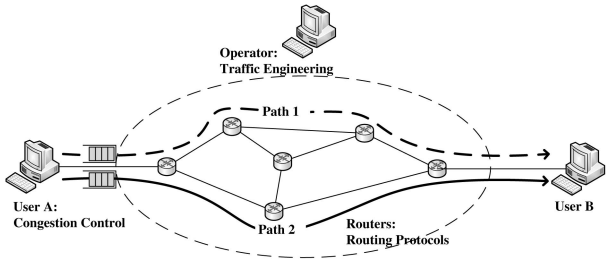

Fig. 1. Illustration of Internet traffic management modules.

translates the multipath utility maximization into a sequence of unconstrained optimization problems, with infinite logarithm barriers being deployed at the constraint boundary. We demonstrate that setting up barriers is much simpler than choosing cost functions and, more importantly, it makes optimal solution achievable. It can be shown that the sequence of the unconstrained optimization approaches to the optimal solution of one of the original problems [2]. We further demonstrate a distributed implementation, together with the design of a practical Logarithm-Barrier-based Multipath Protocol (LBMP). Our LBMP also allows every link to be configured with different control parameters, providing flexibility in dealing with traffic bursts.

We evaluate the performance of LBMP through both numerical analysis and packet-level simulations. The results show that LBMP achieves high throughput and fast convergence over diverse representative network topologies. Such performance is comparable to TRUMP and is often better. In addition, LBMP is flexible in differentiating the control at different links and its optimality and convergence are theoretically guaranteed.

The rest of this paper is organized as follows: Section 2 gives the background and related work. We provide the theoretical foundations of LBMP in Section 3, followed by the distributed implementation in Section 4. In Sections 5 and 6, we investigate the performance of LBMP through numerical and packet-level simulations, respectively. Finally, Section 7 concludes the paper and offers some future directions.

2 Background and Related Work

In this section, we present the network model for traffic management, together with an overview of the recent studies on multipath Internet traffic management.

2.1 Network Model

We focus on a network of directed links, $l \in L$, and origin-destination pairs, $s \in S$. Each origin-destination pair represents a source of traffic (or *source* in short) in the network. Associated with a source is a set of routes, each being a set of links.

We represent the routing by matrix R_{ls} that captures the fraction of source s's flow traversing the links, and we let c_l denote the capacity of link l. As shown in [18], [24], the network utilization maximization problem can be formulated as

$$\max_{\mathbf{R},\mathbf{x}\geq 0} \sum_s U_s(x_s) \text{ subject to } \mathbf{R}\mathbf{x} \leq \mathbf{c}, \tag{1}$$

where both $\mathbf{R}$ and $\mathbf{x}$ are variables. The utility functions U_s are increasing, strictly concave, and twice continuously differentiable.[1]

For single-path routing that is widely used in the current Internet, $\mathbf{R}$ is a 0-1 matrix. Set $R_{ls} = 1$, if link l is in a path of source s, and set $R_{ls} = 0$ otherwise. It is known that single-path routing limits the achievable throughput. If a flow can be flexibly divided and delivered over multiple paths, higher efficiency and robustness can be expected. For multipath routing, the routing matrix's elements are in the range $[0, 1]$. Set $R_{ls} \in (0, 1]$, if link l is in a path of source s, and set $R_{ls} = 0$ otherwise.

2.2 Multipath Utility Maximization: An Overview

Multipath utility maximization appears naturally in many resource allocation problems in communication networks, such as the multipath flow control [7], the optimal quality-of-service(QoS) routing [5], [18] and the optimal network pricing [17].

It is shown in [23] that even the single-path utility maximization is NP-hard and generally has a duality gap. An equilibrium of TCP/IP exists if and only if the single-path utility maximization has no duality gap. In this case, TCP/IP incurs no penalty in not splitting traffic across multiple paths. Such an equilibrium gives a solution to the single-path utility maximization and its Lagrangian dual, but can be quite unstable [24].

For multipath routing, we use z_j^s to represent the sending rate of source s on its jth path. We also represent available paths by a matrix $\mathbf{H}$ where $H_{lj}^s = 1$, if path j of source s uses link l, and $H_{lj}^s = 0$ otherwise. $\mathbf{H}$ does not necessarily represent all possible paths in the physical topology, but only a subset of paths chosen by the network operators or the routing protocol. We can then rewrite the Multipath Utility Maximization as:

$$\begin{array}{ll} \underset{\mathbf{z}\geq\mathbf{0}}{\text{maximize}} & \displaystyle\sum_s U_s\left(\sum_j z_j^s\right) \\ \text{subject to} & \mathbf{H}\mathbf{z} \leq \mathbf{c}. \end{array} \tag{2}$$

This is a convex program with linear constraint, and hence has no duality gap.

In [18], a Dual-based Utility Maximizing Protocol (DUMP) is constructed from a distributed solution of (2) using decomposition. DUMP is similar to the TCP dual algorithm in [16], except that the local maximization is conducted over a vector $\mathbf{z}^s$, as opposed to a scalar x_s only. DUMP, however, has poor convergence behavior with greedy flows [18], because the sources can only reduce their sending rates after packet losses. In addition, its utility is based on throughput only. As such, some links will be operating at close-to-full capacity, resulting in long delays, particularly with traffic bursts.

DUMP was later enhanced by combining performance metrics (users' objective) with network robustness (operator's objective) [9], [10], which leads to the following convex optimization problem:

1. Unless explicitly specified, we use bold upper-case letters to denote matrices, e.g., $\mathbf{H}, \mathbf{R}$, and lower-case letters with subscripts to denote components, e.g., y_l as the lth component of $\mathbf{y}$; superscript is used to denote vectors or matrices pertaining to source s, e.g., $\mathbf{z}^s, \mathbf{p}^s$ and $\mathbf{B}^s$. The notation $[z]^+$ denotes $\max\{0, z\}$.

$$\begin{array}{ll}\underset{\mathbf{z}\geq\mathbf{0}}{\text{maximize}} & \sum_s U_s\left(\sum_j z_j^s\right) - w\sum_l f(y_l/c_l) \\ \text{subject to} & \mathbf{y}\leq\mathbf{c}, \\ & \mathbf{y}=\mathbf{Hz},\end{array} \tag{3}$$

where f is a convex, non-decreasing, and twice-differentiable function that gives heavier penalty as link load increases, e.g., e^{y_l/c_l}. w is a parameter that balances utility and cost. When w is small, the algorithm is very close to DUMP; when w is large, the solution is conservative in avoiding link overload.

A special case of $w=1$, DATE (Distributed Adaptive Traffic Engineering), is examined in [8]. Four distributed algorithms have also been developed based on different decompositions. Combining their best features, TRUMP, a new traffic management protocol, is proposed in [10]. TRUMP is distributed, adaptive, robust, flexible, and easy to manage. Unfortunately, the TRUMP's convergence is not fully proved, nor is its optimality. It is also inflexible in differentiating the control at different links. We will demonstrate that our LBMP well addresses these issues with comparable and often better performance to TRUMP.

3 Logarithm-Barrier-Based Multipath Utility Maximization

Our LBMP is derived using a *Barrier function* technique [2], which translates a constrained optimization problem into a sequence of simpler unconstrained optimization problems; it then constructs infinite barriers at the constraint bounds, and ensures every optimization iteration strictly meet the respective constraints. We will demonstrate three noticeable benefits of applying this barrier function method in the multipath traffic management context. First, setting up barriers is much simpler than choosing cost functions; second, with commonly used logarithm barriers [2], it enables exact solution to the multipath utility maximization; and finally, every link can be allocated with different control parameters, providing flexibility in dealing with traffic bursts.

3.1 Primal Problem

Consider the logarithmic barrier function for (2) with inequality constraint $\mathbf{y}\leq\mathbf{c}$, that is,

$$\begin{array}{ll}\underset{\mathbf{z}\geq\mathbf{0}}{\text{maximize}} & \sum_s U_s\left(\sum_j z_j^s\right) + \sum_l w_l\ln(c_l-y_l) \\ \text{subject to} & \mathbf{y}=\mathbf{Hz}.\end{array} \tag{4}$$

Here, w_l is the barrier parameter associated with $y_l\leq c_l$, and can be viewed as the control parameter set by the operator for link l. Note that the variables in the logarithm function must be positive; the logarithm barrier item thus implicitly includes the constraint $y_l\leq c_l$.

The formulation (4) is equivalent to choosing $f(y_l/c_l)=-\frac{w_l}{w}\ln c_l(1-y_l/c_l)$ in (3), which is not strictly concave. This leads to a discontinuous dual model if we solve (4) directly. As shown in [12], it will be difficult to prove the convergence in this case. Motivated by [7], we introduce a term containing μ to make the objective function being strictly concave, i.e.,

$$\begin{array}{ll}\underset{\mathbf{y}\geq\mathbf{0}}{\text{maximize}} & \sum_s U_s\left(\sum_j z_j^s\right) + \mu\sum_s\sum_j \ln z_j^s \\ & +\sum_l w_l\ln(c_l-y_l) \\ \text{subject to} & \mathbf{y}\geq\mathbf{Hz},\end{array} \tag{5}$$

where μ is a small positive parameter. The new term ensures that there will be a unique maximum to (5), and that also a nonzero rate will be used on any path. The latter is important because otherwise an additional probing protocol would have to be used for probing each high-price path, so as to enable transmission on the path when its price drops significantly. Note that function $\ln(c_l-y_l)$ decreases with y_l. The optimal solution thus will certainly meet the equality bound of y_l; that is, constraint $\mathbf{y}=\mathbf{Hz}$ is equivalent to $\mathbf{y}\geq\mathbf{Hz}$. As such, the objective function becomes *strictly* concave, and hence, the problem has a unique maximizer.

3.2 Dual Problem

We now provide an exact solution to the problem through optimization decomposition. We start from the Lagrangian dual of (5). If link l is in the jth path of source s, we write $l\in(s,j)$. Hence, we can rewrite $\mathbf{y}\geq\mathbf{Hz}$ as $y_l\geq\sum_{s,j:l\in(s,j)} z_j^s, \forall l$. By assigning each link l with a dual variable p_l, the Lagrangian of (5) is given by:

$$\begin{aligned}L(\mathbf{z},\mathbf{y};\mathbf{p}) = &\sum_s U_s\left(\sum_j z_j^s\right) + \mu\sum_s\sum_j \ln z_j^s \\ &+\sum_l w_l\ln(c_l-y_l) + \sum_l p_l\left(y_l - \sum_{s,j:l\in(s,j)} z_j^s\right).\end{aligned}$$

The dual objective function is $D(\mathbf{p}):=\max_{\mathbf{z},\mathbf{y}\geq\mathbf{0}} L(\mathbf{z},\mathbf{y};\mathbf{p})$. Notice that the first term is separable in $\mathbf{z}^s$, and the second is separable in y_l. Hence, we have $D(\mathbf{p}):=\sum_s B_s(\mathbf{p}^s)+\sum_l B_l(p_l)$, where

$$B_s(\mathbf{p}^s) = \max_{\mathbf{z}^s} U_s\left(\sum_j z_j^s\right) + \mu\sum_j \ln z_j^s - \sum_{j\in s} z_j^s p_j^s, \tag{6}$$

$$B_l(p_l) = \max_{y_l\geq 0} w_l\ln(c_l-y_l) + p_l y_l. \tag{7}$$

Here, $\mathbf{p}^s=(p_j^s), p_j^s=\sum_{l:l\in(s,j)} p_l$. The dual problem of (5) then becomes the selection of the dual vector $\mathbf{p}=(p_l, l\in L)$, so as to

$$\min_{\mathbf{p}\geq\mathbf{0}} \sum_s B_s(\mathbf{p}^s) + \sum_l B_l(p_l). \tag{8}$$

We can interpret the dual variable p_l as the price per unit bandwidth at link l; p_j^s then becomes the price per unit bandwidth of the path j of source s. Therefore, $B_s(\mathbf{p}^s)$ gives the maximum benefit s can achieve at the given (vector) price $\mathbf{p}^s$. For a fixed link load y_l, $y_l p_l$ is the income of the network operator from link l and $-\ln(c_l-y_l)$ is the cost. Clearly, y_l will never exceed c_l, and the closer it is, the greater the cost is. w_l can be interpreted as a tradeoff parameter set by the operator for link l. It can be set to zero or a small positive value for a noncritical and nonbottleneck link. For a bottleneck link, w_l should be set to a large positive value, which ensures the link can deal with unexpected traffic bursts. It follows $w_l\ln(c_l-y_l)+p_l y_l$ being the net benefit of

transmitting at traffic y_l, and $B_l(p_l)$ being the maximum benefits l can achieve at the given price p_l.

Lemma 1. *The dual objective function $D(\mathbf{p})$ is convex and continuously differentiable with $\mathbf{p} \geq \mathbf{0}$. The lth component of $\nabla D(\mathbf{p})$ is given by*

$$\frac{\partial D}{\partial p_l}(\mathbf{p}) = y_l(\mathbf{p}) - x^l(\mathbf{p}), \tag{9}$$

where

$$y_l(\mathbf{p}) = \begin{cases} c_l - \dfrac{w_l}{p_l}, & if\ p_l \geq \dfrac{w_l}{c_l}, \\ 0, & otherwise, \end{cases} \tag{10}$$

and for all s and j:

$$U_s'\left(\sum_i z_i^s(\mathbf{p})\right) + \frac{\mu}{z_j^s(\mathbf{p})} - p_j^s = 0, \tag{11}$$

and $x^l(\mathbf{p}) = \sum_{s,j:l\in(s,j)} z_j^s(\mathbf{p}), p_j^s = \sum_{l:l\in(s,j)} p_l$.

Proof. The objective function of (5) is *strictly* concave; hence, $D(\mathbf{p})$ is convex and $\nabla D(\mathbf{p})$ indeed exists and $\nabla D(\mathbf{p}) = \mathbf{y}(\mathbf{p}) - \mathbf{H} * \mathbf{z}(\mathbf{p})$, where $y_l(\mathbf{p})$ is the solution to (7) and $\mathbf{z}^s(\mathbf{p})$ is the solution to (6) [2].

Let $f(y_l) := w_l \ln(c_l - y_l) + p_l y_l$. We need to find the maximizer of $f(y_l)$ in the interval $[0, c_l)$. If $p_l = 0$, it is obvious that $y_l(\mathbf{p}) = 0$ for $w_l \ln(c_l - y_l)$ decreasing with y_l. If $p_l > 0$, note that $f(y_l)$ is strictly concave, so we only need find the stationary point of $f(y_l)$, i.e., $c_l - \frac{w_l}{p_l}$. If $c_l - \frac{w_l}{p_l} < 0$, we have $y_l(\mathbf{p}) = 0$ for $f(y_l)$ decreasing with y_l in the interval $[0, c_l)$; otherwise, we have $y_l(\mathbf{p}) = c_l - \frac{w_l}{p_l}$. $\mathbf{z}^s(\mathbf{p})$ is a solution to (6) if and only if $\mathbf{z}^s(\mathbf{p})$ is the stationary point of the objective function of (6), i.e., the (11) holds for all j. □

It is well known that the dual is a convex problem. Lemma 1 shows that the problem is differentiable, so we can apply a gradient projection method. The explicit expression of gradient is also given by Lemma 1.

3.3 Control Parameters versus Congestion Measures

A sailing feature of our barrier-function-based approach is that the exact relation between its control parameters and the link congestion measure can be derived. Specifically, we have the following theorem:

Theorem 2. *Given $\mathbf{w}$ with $w_l > 0$ for all $l \in L$. Let $\tilde{\mathbf{p}}$ be a solution to problem (8), and $\tilde{\mathbf{z}}$, and $\tilde{\mathbf{y}}$ be solutions to (6) and (7) with $\mathbf{p} = \tilde{\mathbf{p}}$, respectively. We have*

1) *$(\tilde{\mathbf{z}}, \tilde{\mathbf{y}})$ is a solution to (5) with $\tilde{y}_l = \sum_{s,j:l\in(s,j)} \tilde{z}_j^s$. Moreover, we have $\tilde{p}_l = \frac{w_l}{c_l - \tilde{y}_l}$ if $\tilde{y}_l > 0$;*
2) *Let $(\bar{\mathbf{z}}, \bar{\mathbf{p}})$ be a limit point of $(\tilde{\mathbf{z}}, \tilde{\mathbf{p}})$ as $\mathbf{w}$ converges to zero, then $\bar{\mathbf{z}}$ is a solution to problem*

$$\begin{aligned} &maximize && \sum_s U_s\left(\sum_j z_j^s\right) + \mu \sum_s \sum_j \ln z_j^s \\ &subject\ to && \mathbf{Hz} \leq \mathbf{c}, \end{aligned} \tag{12}$$

and $\bar{\mathbf{p}}$ is the corresponding Lagrangian multiplier.

Proof. 1) By the optimality of $\tilde{\mathbf{p}}$, we have $\frac{\partial D}{\partial p_l}(\tilde{\mathbf{p}}) = 0$ if $\tilde{p}_l > 0$, and $\frac{\partial D}{\partial p_l}(\tilde{\mathbf{p}}) \geq 0$ if $\tilde{p}_l = 0$. With (9), we have $\tilde{y}_l \geq \sum_{s,j:l\in(s,j)} \tilde{z}_j^s$ holds for $\forall l$. It can be seen that $(\tilde{\mathbf{z}}, \tilde{\mathbf{y}})$ is feasible for problem (5) and $\sum_s (U_s(\sum_j \tilde{z}_j^s) + \mu \sum_j \ln \tilde{z}_j^s) + \sum_l w_l \ln(c_l - \tilde{y}_l) = L(\tilde{\mathbf{z}}, \tilde{\mathbf{y}}; \tilde{\mathbf{p}}) = D(\tilde{\mathbf{p}})$. By the weak duality, $(\tilde{\mathbf{z}}, \tilde{\mathbf{y}})$ is a solution to problem (5). Since $\ln(c_l - y_l)$ strictly decreases with y_l, we have $\tilde{y}_l = \sum_{s,j:l\in(s,j)} \tilde{z}_j^s$ by the optimality of $\tilde{\mathbf{y}}$. By Lemma 1, we get $\tilde{p}_l = \frac{w_l}{c_l - \tilde{y}_l}$ if $\tilde{y}_l > 0$.

2) Given the optimality (10) of $\tilde{\mathbf{z}}$, we have

$$U_s'\left(\sum_i \tilde{z}_j^s\right) - \tilde{p}_j^s + \frac{\mu}{\tilde{z}_j^s} = 0, \quad \text{for all } s, j, \tag{13}$$

where $\tilde{p}_j^s = \sum_{l:l\in(s,j)} \tilde{p}_l$. Given the optimality condition (10) of $\tilde{\mathbf{y}}$, we have

$$\tilde{p}_l(c_l - \tilde{y}_l) \leq w_l. \tag{14}$$

Let $\mathbf{w} \to \mathbf{0}$, by the continuity of U_s', the (13) becomes

$$U_s'\left(\sum_i \bar{z}_j^s\right) - \bar{p}_j^s + \frac{\mu}{\bar{z}_j^s} = 0, \quad \text{for all } s, j.$$

Let $\mathbf{w} \to \mathbf{0}$, we have $\bar{p}_l(c_l - \bar{y}_l) = 0$ from (14), where $\bar{y}_l = \sum_{(s,j):(s,j)\in l} \bar{z}_j^s$. Hence, $\bar{y}_l = c_l$ when $\bar{p}_l > 0$, and $\bar{p}_l = 0$ when $\bar{y}_l < c_l$. Therefore, $\bar{\mathbf{z}}$ is a KKT (Karush-Kuhn-Tucker) point of (12), and $\bar{\mathbf{p}}$ is the corresponding Lagrangian multiplier. Since (12) is a convex programming problem with a strictly concave objective function, $\bar{\mathbf{z}}$ is the unique solution to (12), and $\bar{\mathbf{p}}$ is a solution to the dual of (12). □

Theorem 2 reveals the relationship between the solutions of (5) and (8). That is, if we solve the dual problem (8), we can as well obtain the solutions of (5), and such relationship as $\tilde{p}_l = \frac{w_l}{c_l - \tilde{y}_l}$ exists between them. $\frac{w_l}{c_l - \tilde{y}_l}$ can be regarded as the approximation for the network congestion measure, or $w_l \approx \bar{p}_l(c_l - \tilde{y}_l)$. Hence, $c_l - \tilde{y}_l$ increases with the increasing of w_l, improving the ability of link l in dealing with the network traffic bursts. In addition, when $\mathbf{w}$ converges to zero, the solution to (5) approximates to that of (12). With sufficiently small μ, when $\mathbf{w}$ converges to zero, the solution to (5) approaches to that of (2) and the network achieves the maximum aggregate utility.

4 Distributed Algorithm and Practical LBMP Implementation

In this section, we present a distributed practical implementation for the optimal solution above. Our solution is based on a gradient projection method [2], because the problem is differentiable as suggested by Lemma 1.

4.1 Distributed Algorithm

We apply the gradient projection method to (8), solving the problem and its dual iteratively, as follows:

$$p_l(t+1) = \left[p_l(t) + \beta\left(\sum_{s,j:l\in(s,j)} z_j^s(t) - y_l(t)\right)\right]^+, \tag{15}$$

where

$$\mathbf{z}^s(t) = \arg\max_{\mathbf{z}^s} U_s\left(\sum_j z_j^s\right) + \mu \sum_j \ln z_j^s - \sum_j \sum_{l:l\in(s,j)} z_j^s p_l(t) \tag{16}$$

and

$$y_l(t) = \begin{cases} c_l - \dfrac{w_l}{p_l(t)}, & \text{if } p_l(t) \geq \dfrac{w_l}{c_l}, \\ 0, & \text{otherwise}, \end{cases} \tag{17}$$

where $\beta > 0$ is a constant stepsize. When the stepsize in the iteration tends to zero, the obtained sequence converges to the solutions of (5) and (8).

Here, y_l can be considered as the effective capacity in (15) and (17), and $\mathbf{p}$ is the link congestion measure, which is determined according to the demand for bandwidth and the effective capacity y_l at time t. Equation (16) denotes the ith source maximizing its net utility according to the congestion level of the paths at time t. Equation (17) gives the effective capacity, which is determined by the network operator according to the information of link l and the network congestion level at time t.

4.2 Convergence Analysis

We now prove that the above algorithm generates a sequence that approaches the optimal rate allocation, provided the following conditions in [1] are satisfied:

C1: On the interval $I_s = [m_s, M_s]$, the utility functions U_s are increasing, strictly concave, and twice continuously differentiable, where $m_s > 0$ and $M_s < \infty$ are the minimum and maximum transmission rates required by source s, respectively.

C2: The curvatures of U_s are bounded above and away from zero on I_s: $1/\underline{\alpha}_s \geq -U_s''(x_s) \geq 1/\bar{\alpha}_s > 0$ for all $x_s \in I_s$.

Lemma 3. *Under the conditions C1 and C2, ∇D is Lipschitz with*

$$\|\nabla D(\mathbf{q}) - \nabla D(\mathbf{p})\|_2 \leq (\bar{c} + \bar{\alpha}\bar{L}\bar{R})\|\mathbf{q} - \mathbf{p}\|_2$$

for all $\mathbf{p}, \mathbf{q} \geq 0$, where

$$\bar{c} = \max_l \frac{c_l^2}{w_l}, \quad \bar{\alpha} = \max_s \frac{M_s(M_s + \mu\bar{\alpha}_s)}{m_s + \mu^2\underline{\alpha}_s},$$
$$\bar{L} = \max_s \sum_l \sum_j H_{lj}^s,$$

and $\bar{R} = \max_l \sum_s \sum_j H_{lj}^s$.

Proof. We first derive the Hessian of D. By (10), we have

$$\frac{\partial y_{l'}}{\partial p_l}(\mathbf{p}) = \begin{cases} \dfrac{w_l}{p_l^2}, & l = l' \text{ and } p_l \geq \dfrac{w_l}{c_l}, \\ 0, & \text{otherwise} \end{cases}.$$

By (11), we have

$$\frac{\partial z_j^s}{\partial p_l}(\mathbf{p}) = -\frac{\frac{z_j^s}{\mu}\left(\frac{\sum_{i:i\neq j} z_i^s}{\mu} - \left[U_s''(\sum_i z_i^s)\right]^{-1}\right)H_{lj}^s - \sum_{i:i\neq j}\frac{z_j^s z_i^s}{\mu^2}H_{li}^s}{\frac{\sum_i z_i^s}{\mu} - \left[U_s''(\sum_i z_i^s)\right]^{-1}}.$$

Let $\mathbf{A}(\mathbf{p}) = \mathrm{diag}(\frac{\partial y_l}{\partial p_l}, l \in L)$ be the $L \times L$ diagonal matrix with diagonal elements $\frac{\partial y_l}{\partial p_l}$ and $\mathbf{B}(\mathbf{p}) = \mathrm{diag}(\mathbf{B}^s(\mathbf{p}))$ be the $S \times S$ diagonal block matrix with diagonal block elements $\mathbf{B}^s(\mathbf{p})$, where $\mathbf{B}^s(\mathbf{p})$ is a square matrix with element

$$[\mathbf{B}^s(\mathbf{p})]_{i,j} = \frac{1}{\frac{\sum_i z_i^s}{\mu} - \left[U_s''(\sum_i z_i^s)\right]^{-1}} b_{ij}$$

and

$$b_{ij} = \begin{cases} \dfrac{z_j^s}{\mu}\left(\dfrac{\sum_{i:i\neq j} z_i^s}{\mu} - \left[U_s''\left(\sum_i z_i^s\right)\right]^{-1}\right), & i = j, \\ -\dfrac{z_j^s}{\mu}\dfrac{z_i^s}{\mu}, & i \neq j. \end{cases}$$

Using (9), we have

$$\nabla^2 D(\mathbf{p}) = \mathbf{A}(\mathbf{p}) + \mathbf{H}\mathbf{B}(\mathbf{p})\mathbf{H}^T.$$

Given any $\mathbf{p}, \mathbf{q} \geq \mathbf{0}$, using Taylor theorem, we have

$$\begin{aligned} \nabla D(\mathbf{q}) - \nabla D(\mathbf{p}) &= \nabla^2 D(\mathbf{w})(\mathbf{q} - \mathbf{p}) \\ &= (\mathbf{A}(\mathbf{p}) + \mathbf{H}\mathbf{B}(\mathbf{p})\mathbf{H}^T)(\mathbf{q} - \mathbf{p}) \end{aligned}$$

for some $\mathbf{w} = t\mathbf{p} + (1-t)\mathbf{q}, t \in (0,1)$. Hence, $\|\nabla D(\mathbf{q}) - \nabla D(\mathbf{p})\|_2 \leq (\|\mathbf{A}(\mathbf{p})\|_2 + \|\mathbf{H}\mathbf{B}(\mathbf{p})\mathbf{H}^T\|_2)\|\mathbf{q} - \mathbf{p}\|_2$.

It is obvious that $\|\mathbf{A}(\mathbf{p})\|_2 = \bar{c}$. With the similar method in [1], it can be shown that $\|\mathbf{H}\mathbf{B}(\mathbf{p})\mathbf{H}^T\|_2 \leq \bar{L}\bar{\alpha}\bar{R}$. □

Theorem 4. *Provided that the stepsize β is sufficiently small, then starting from any prices $\mathbf{p}(0) \geq \mathbf{0}$, every limit point $(\mathbf{z}^*, \mathbf{p}^*)$ of the sequence $(\mathbf{z}(t), \mathbf{p}(t))$ generated by (15)-(17) is primal-dual optimal.*

Proof. Equations (15)-(17) represent the gradient projection method with constant stepsize. Lemma 3 shows that $\nabla D(p)$ is Lipschitz. By Proposition 2.3.2 in [2], it holds that $\mathbf{p}^*$ is stationary, if $0 < \beta < \frac{2}{\bar{c}+\bar{\alpha}L\bar{R}}$, i.e.

$$\nabla D(\mathbf{p}^*)^T(\mathbf{p} - \mathbf{p}^*) \geq 0, \quad \forall \mathbf{p} \geq \mathbf{0}.$$

Since $D(\mathbf{p})$ is convex, we have

$$D(\mathbf{p}) - D(\mathbf{p}^*) \geq \nabla D(\mathbf{p}^*)^T(\mathbf{p} - \mathbf{p}^*) \geq 0, \quad \forall \mathbf{p} \geq \mathbf{0}.$$

That is, $\mathbf{p}^*$ is dual optimal. It follows that $\mathbf{z}^*$ is primal optimal since $\mathbf{z}(t)$ defined by (16) satisfies (11) with $\mathbf{p} = \mathbf{p}(t)$, which is continuous. □

4.3 LBMP: Practical Distributed Implementation

Note that (15)-(17) ignore feedback delay. They also assume fluid traffic flows. This is not true for packet switched Internet. To implement LBMP in the real Internet, the source sending rate update depends on T_j^s, the time it takes for source s to receive feedback along all the links of path j. The link prices are calculated based on the estimated local link load: N_T, the number of bits, which arrived in period $(t, t+T)$ divided by the length of the period. We now detail the update operations as follows:

Source s's jth flow rate update. For each subproblem from (16), we have that $\mathbf{z}^s(t)$ is a solution for the given price vector $\mathbf{p}(t)$ if and only if $\mathbf{z}^s(t)$ satisfies (11), i.e.,

$$\frac{1}{\sum_i z_i^s(t)} + \frac{\mu}{z_j^s(t)} - p_j^s(t) = 0, \quad \forall j.$$

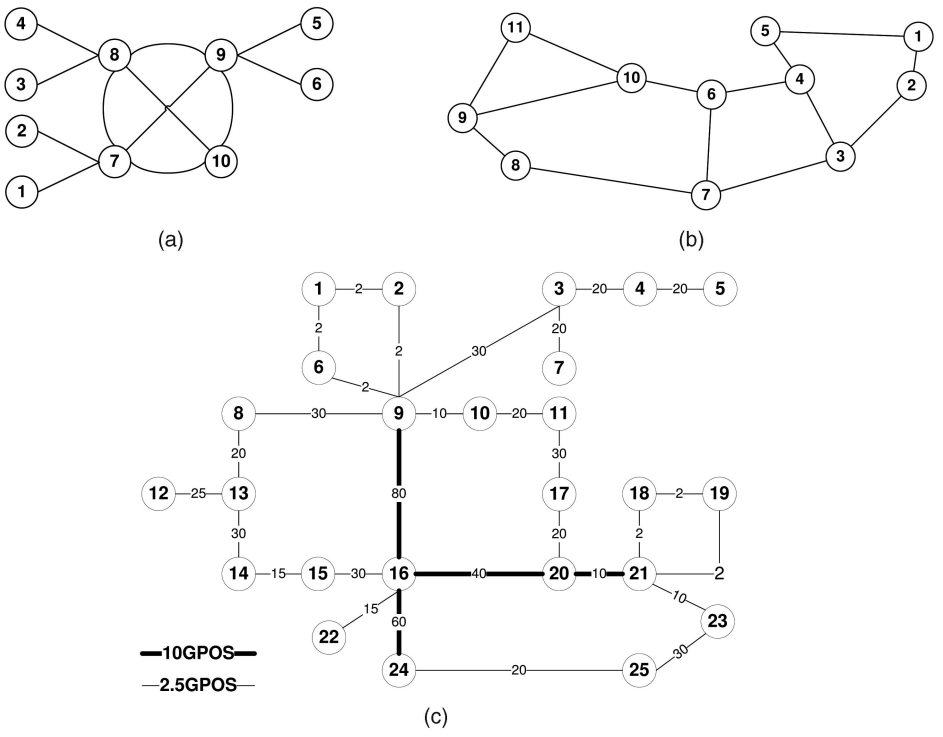

Fig. 2. Three realistic network topologies. (a) Access-core topology. (b) Abilene topology. (c) Cernet topology.

For the jth flow of source s, if

$$p_j^s(t) > \frac{1}{\sum_i z_i^s(t)} + \frac{\mu}{z_j^s(t)},$$

i.e.,

$$\frac{1}{p_j^s(t)} - \frac{z_j^s(t)\sum_i z_i^s(t)}{z_j^s(t) + \mu\sum_i z_i^s(t)} < 0,$$

it is necessary to reduce the sending rate of this flow, given its congestion measure is relatively high. In this case, the difference

$$\frac{1}{p_j^s(t)} - \frac{z_j^s(t)\sum_i z_i^s(t)}{z_j^s(t) + \mu\sum_i z_i^s(t)}$$

can be applied in our update. A weighting factor γ is also introduced to avoid great variation in each iteration, we take $\gamma = 0.1$ in both Section 5 and Section 6. We then have the following source rate update:

$$z_j^s(t + T_j^s) = \left[z_j^s(t) + \gamma\left(\frac{1}{p_j^s(t)} - \frac{z_j^s(t)\sum_i z_i^s(t)}{z_j^s(t) + \mu\sum_i z_i^s(t)} \right) \right]^+. \quad (18)$$

On the other hand, if $p_j^s(t) < \frac{1}{\sum_i z_i^s(t)} + \frac{\mu}{z_j^s(t)}$, i.e.,

$$\frac{1}{p_j^s(t)} - \frac{z_j^s(t)\sum_i z_i^s(t)}{z_j^s(t) + \mu\sum_i z_i^s(t)} > 0,$$

it is necessary to increase the sending rate of this flow, and (18) can be used as well. If

$$p_j^s(t) = \frac{1}{\sum_i z_i^s(t)} + \frac{\mu}{z_j^s(t)},$$

i.e.,

$$\frac{1}{p_j^s(t)} - \frac{z_j^s(t)\sum_i z_i^s(t)}{z_j^s(t) + \mu\sum_i z_i^s(t)} = 0,$$

no update is necessary.

Congestion measure update of link l. Since the congestion measure at time $t + T$ is calculated explicitly according to the status of link l at time t, we only need to approximate the local link load $\sum_{s,j:l\in(s,j)} z_j^s(t)$ of link l at time t with N_T. That is,

$$p_l(t + T) = \left[p_l(t) - \beta\left(y_l(t) - \frac{N_T}{T c_l} \right) \right]^+,$$

where the effective capacity of link l is

$$y_l(t) = \begin{cases} c_l - \dfrac{w_l}{p_l(t)}, & \text{if } p_l(t) \geq \dfrac{w_l}{c_l}, \\ 0, & \text{otherwise.} \end{cases}$$

For the utility function U, we employ a logarithmic function $\ln x_s$, which is commonly used for ensuring proportional fairness [19]. Other functions, however, can be applied in our framework as well.

5 LBMP Performance: Numerical Investigation

To understand the performance of LMBP and to compare it with state-of-the-art solutions, we now present a MATLAB-based numerical study. We will conduct further packet-level simulations and comparison in the next section.

In our MATLAB-based numerical experiments, we use three typical network topologies, as shown in Fig. 2. The

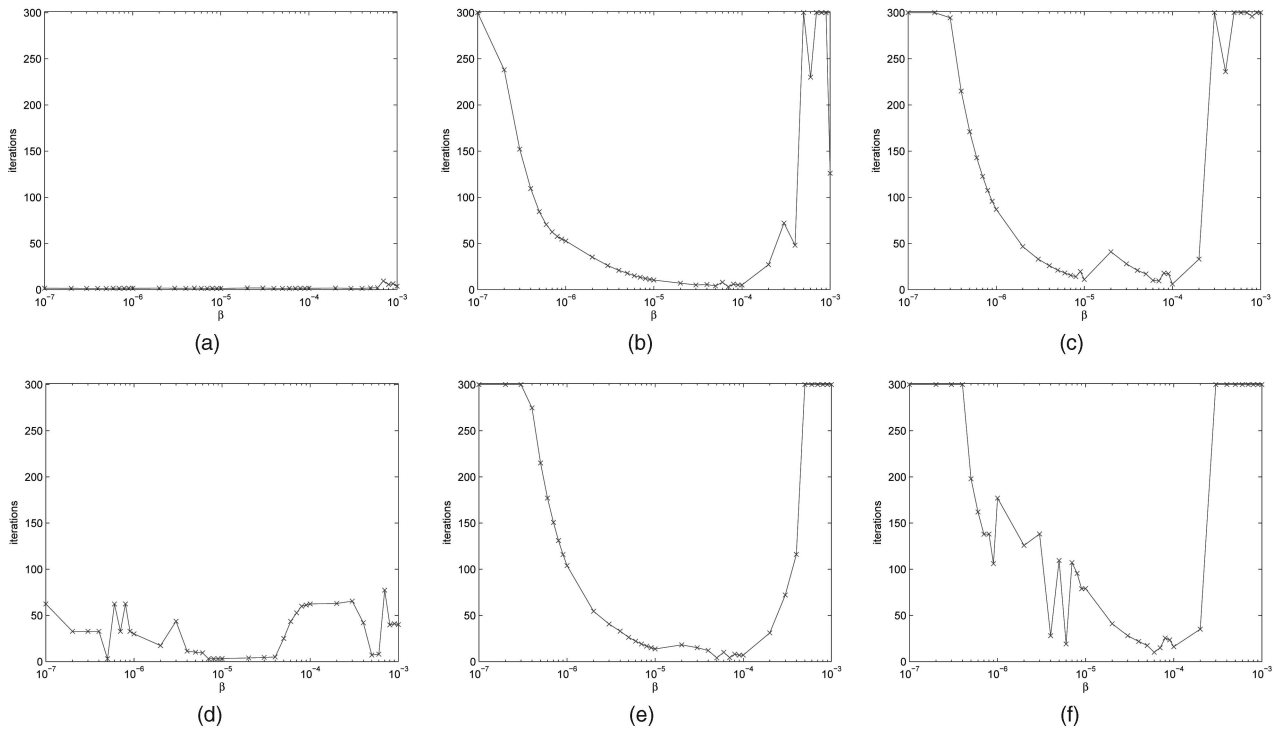

Fig. 3. LBMP's convergence speed against stepsize for the Access-core topology. 'x' represents the average value over 10 runs with different initial points. (a) $w = 1$. (b) $w = 1/6$. (c) $w = 1/36$. (d) $w = 1$. (e) $w = 1/6$. (f) $w = 1/36$.

first is a tree-mesh topology, which models a common Access-core network structure [9]; the second is the Abilene backbone network structure [26]; the third is the backbone of the China Education and Research Network (Cernet) [27], where the realistically measured average delays are displayed along the links.

The settings of source-destination pairs and paths for the Access-core topology and the Abilene backbone topology are the same as that in [9]. That is, we select six source-destination pairs (1-3, 1-5, 2-4, 2-6, 3-5, 4-6) for Access-core and four pairs (1-6, 3-9, 7-11, 1-11) for Abilene. For each source-destination pair, we choose three minimum-hop paths as possible paths for Access-core and four minimum-hop paths as possible paths for Abilene. For Cernet, we select six source-destination pairs (12-2, 22-21, 24-7, 25-9, 21-1, 5-19), for each of which we choose three minimum-hop paths as possible paths.

For the cost function in (3), we use the exponential function $f(y_l/c_l) = e^{y_l/c_l}$ as suggested in [9] for TRUMP. That is to say, we consider the exponential-cost-based TRUMP and the logarithmic-barrier-based LBMP. We set link capacity $c_l = 100$ Mbps for Access-core and Abilene. For Cernet, since the real link capacities are too high to be used for our later packet-level simulations, we proportionally reduce the capacities by 100 times, to between 25 Mbps and 100 Mbps.

In all the MATLAB-based numerical experiments and NS-2 simulations, we define the link utilization for link l as $\mathtt{LU}_l = (\sum_{s,j:(s,j)\in l} z_j^s)/c_l$ and the number of the full link utilization ($\mathtt{LU}_l = 1$) $\mathtt{NF}$. We definite the (maximum) link utilization as $\mathtt{NU} = \max_l \mathtt{LU}_l$ and the aggregate throughput as $\mathtt{NT} = \sum_{s,j} z_j^s$ for the network, respectively.

5.1 Convergence Speed

We first evaluate the convergence speed of LBMP, in particular, the impact of stepsize β and the control parameter $\mathbf{w}$. To fairly compare LBMP with TRUMP, we set the tunable parameters $w_i = w$ for all links and $\mu = 0$, and we will examine heterogeneous w_i later. For this set of experiments, when the relative difference between the objective function value of (5) obtained through LBMP and that obtained directly with the function `fmincon` in MATLAB is less than 1 percent, LBMP with maximum iteration count 300 is terminated. In Figs. 3a, 3b, and 3c, we plot the stepsize versus the number of iterations for $w = 1, 1/6$ and $1/36$, respectively. We use the Access-core topology as a representative and similar results have been observed with the other two topologies. For each stepsize, 10 random initial points $z_j^s(0)$(uniformly from interval [0, 5]) and $p_l(0) = 0.01$ are chosen for the experiment. The presented results are the average of the 10 runs of LBMP with different initial points.

From Fig. 3a, it can be seen that LBMP terminates almost instantly. After checking the iterations, however, we notice that although the value of the objective function changes little over the iterations, there are significant differences in the link utilization and the aggregate throughput. Hence, we add two termination criteria for LBMP in the above experiments, i.e., the relative error of the aggregate throughput being less than 5 percent and the absolute error of the link utilization being less than 0.02. The results are shown in Figs. 3d, 3e, and 3f.

From Fig. 3, we see that, as w shrinks, the number of iterations with the optimal stepsize gradually grows, and the range of stepsizes with fast convergence gradually shrinks. This explains why DUMP ($w = 0$) is difficult to

TABLE 1
The Optimal Link Utilization and Aggregate Throughput

Topology	NT(Mbps)	NU(%)	NF
Access-core	333	100	5
Abilene	300	100	5
Cernet	175	100	15

TABLE 2
MATLAB-Based Results for TRUMP and LBMP

Topology	TRUMP				LBMP			
	w	NT(Mbps)	NU(%)	NF	w	NT(Mbps)	NU(%)	NF
Access-core	0.27	307.4	100	4	0.07	306.8	94.8	0
Abilene	0.15	282.6	100	4	0.02	284.8	98.2	0
Cernet	0.20	140.6	100	2	0.10	140.3	90.0	0

tune. Compared with TRUMP, LBMP converges much faster and the range of parameters with good convergence is also broader. Moreover, the convergence of LBMP is theoretically guaranteed.

5.2 Link Utilization and Aggregate Throughput

We again set $w_l = w$ for all links. We first solve (2) by calling function "fmincon" in MATLAB to calculate the link utilization and aggregate throughput for the optimal rates allocation. The results are shown in Table 1.

To show the difference between LBMP and TRUMP in adjusting the link load, problems (3) and (5) (with $\mu = 0.0001$) are solved, respectively, by calling function "fmincon" in MATLAB for a series of ws in the interval $[0.001, 1]$. The results are summarized in Table 2. We plot the network throughput versus w in Figs. 4a, 4b, and 4c, and the link utilization versus w in Figs. 4d, 4e, and 4f. We further plot the aggregate throughput versus the link utilization in Fig. 5.

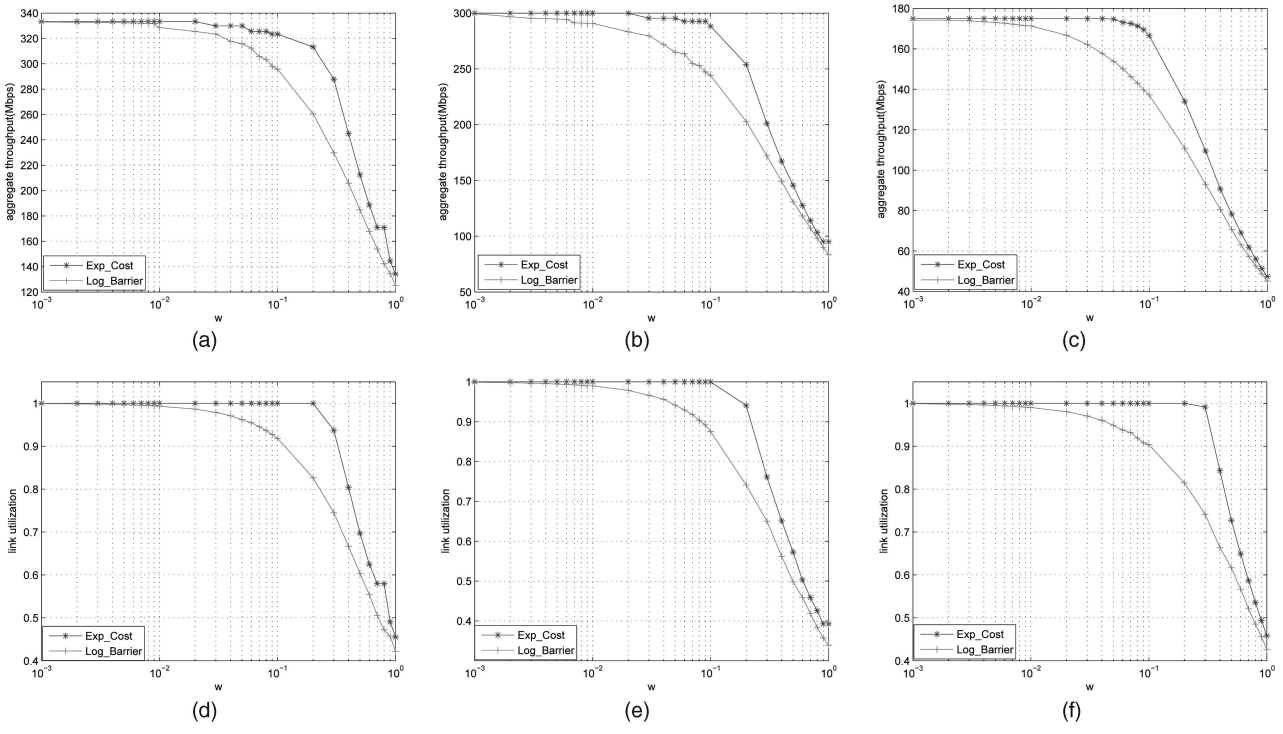

Fig. 4. Aggregate throughput and link utilization versus parameter w. (a) Access-core topology. (b) Abilene topology. (c) Cernet topology. (d) Access-core topology. (e) Abilene topology. (f) Cernet topology.

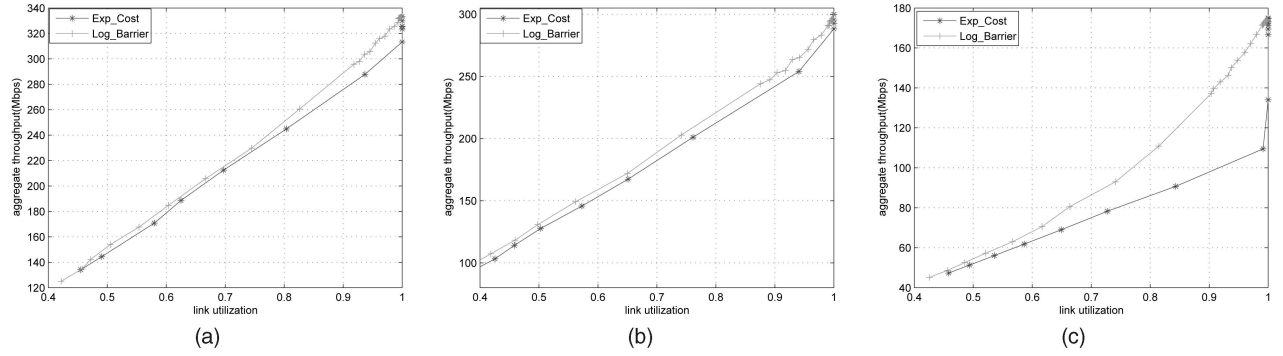

Fig. 5. Aggregate throughput versus link utilization. (a) Access-core topology. (b) Abilene topology. (c) Cernet topology.

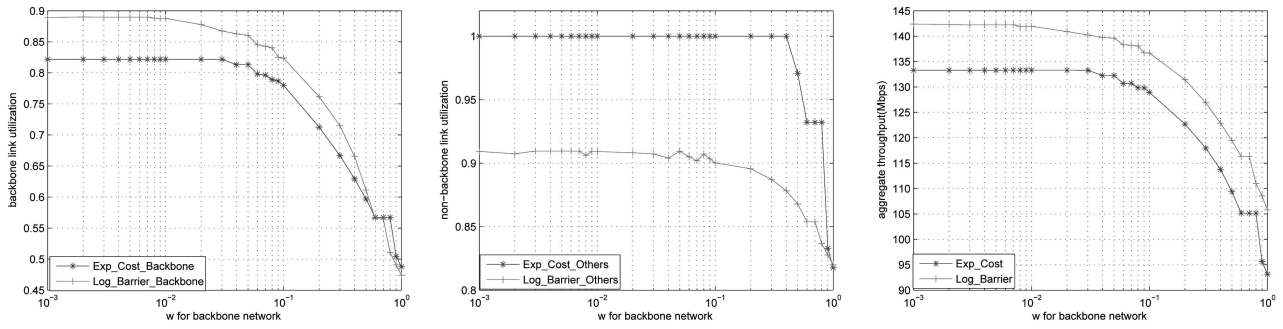

Fig. 6. Plots of LBMP's ability in adjusting load for the backbone links in Cernet topology. (a) Backbone link utilization. (b) Nonbackbone link utilization. (c) Aggregate throughput.

The results in Table 2 indicate that TRUMP reaches full link utilization before achieving the optimal throughput, which is not desirable because the delay can be excessive. LBMP, on the other hand, still maintains reasonable link utilization when achieving the optimal throughput. It is noticed that the w for LBMP in the table is different for different network topologies. Here, we give some suggestions about how to set the w. Recall that Theorem 2 offers the relation between the control parameters and the congestion measures; in particular, $p_l = \frac{w_l}{c_l - y_l}$, where y_l is the total load on link l. Here, the control parameter w_l in LBMP can be interpreted as the constant factor by which the average queuing delay of an M/M/1 queue is increased. Hence, if ignoring the queuing delay, we can set $w_l = 0$ for all l so as to achieve the optimal throughput. If queuing delay can't be ignored, a proper control parameter w_l should be set according to the degree that we concern the queuing delay.

From Fig. 4, we see that link utilization and aggregate throughput achieved by LBMP are quite close to those of TRUMP for $w = 1$. LBMP, however, offers smoother and gentler transitions, indicating that when w approaches zero, the aggregate throughput approaches the maximum aggregate throughput and the link utilization approaches one.

For TRUMP, the link utilization versus w curve has two obvious segments, indicating that as w approaches to zero, the aggregate throughput approaches the maximum aggregate throughput gradually, and the link utilization is always one when w is less than a certain value. As an example, for Access-core (Figs. 4a and 4d), when w is less than 0.3, the links achieve 100 percent utilization and the aggregate throughput is 300 Mbps. For Abilene (Figs. 4b and 4e), when w is less than 0.2, the links achieve 100 percent utilization and the aggregate throughput is 270 Mbps. And for Cernet in Figs. 4c and 4f, when w is less than 0.3, the links achieve 100 percent utilization and the aggregate throughput is 115 Mbps.

From Fig. 5, we see that, for all the three topologies, LBMP generally performs better than TRUMP. For Cernet, as shown in Fig. 5c, for link utilization of 90 percent, the achieved aggregate throughput is 40 Mbps higher than that achieved by TRUMP. As the link utilization increases, the difference between them becomes more significant, indicating that LBMP distributes the network traffic in a more reasonable way.

From Figs. 4 and 5, we can see that LBMP controls the link utilization under 90 percent when achieving 90 percent of the optimal throughput. For TRUMP, however, to reach 90 percent of the optimal throughput, the link utilization is close to 100 percent. The network operator should configure w according to the network status. Unfortunately, the appropriate w is in a very narrow range, indicating that we should pay more attention to the utility than to the congestion cost. For example, $w = 0.01 \sim 0.1$ means that we should configure the ratio between the entire network utility and the congestion cost from 100:1 to 10:1.

5.3 Load Control of Critical Links

Finally, we investigate the effectiveness of LBMP in controlling the load of critical links. In particular, we focus on the Cernet topology whose backbones (critical links) are clearly known. We adjust w for the critical links. For other links, we set $w = 0.25$ in (3) and $w = 0.1, \mu = 0.0001$ in (5). Fig. 6 shows the link utilization versus w for backbone, the link utilization versus w for nonbackbone, and the aggregate throughput versus w, respectively.

From Fig. 6a, we see that as the tunable parameter w of backbone decreases from 1 to 0.01, the backbone link utilization from the TRUMP increases from about 47 percent to 82 percent, and the backbone link utilization from LBMP increases from about 47 percent to 88 percent. From Fig. 6b, we see that as w of backbone decreases from 1 to 0.01, the nonbackbone link utilization from the TRUMP increases from about 82 percent to 100 percent, and the backbone link utilization from LBMP increases from about 82 percent to 91 percent. From Fig. 6c, we see that as the tunable parameter w of backbone decreases from 1 to 0.01, the aggregate throughput from TRUMP increases from about 100 to 133 Mbps, and the aggregate throughput from LBMP increases from about 100 to 142 Mbps. The results clearly demonstrate that the logarithm-barrier-based LBMP is more flexible in adjusting the load of critical links.

6 Packet-Level Simulation Results

We have also conducted a series of *NS*-2 simulations for the LBMP protocol. In this section, we present representative results based on the Access-core topology, the Abilene backbone topology, and the Cernet topology. The settings of source-destination pairs and flows are the same as that in MATLAB experiments. Unless explicitly specified, the following default configurations are used in our simulation:

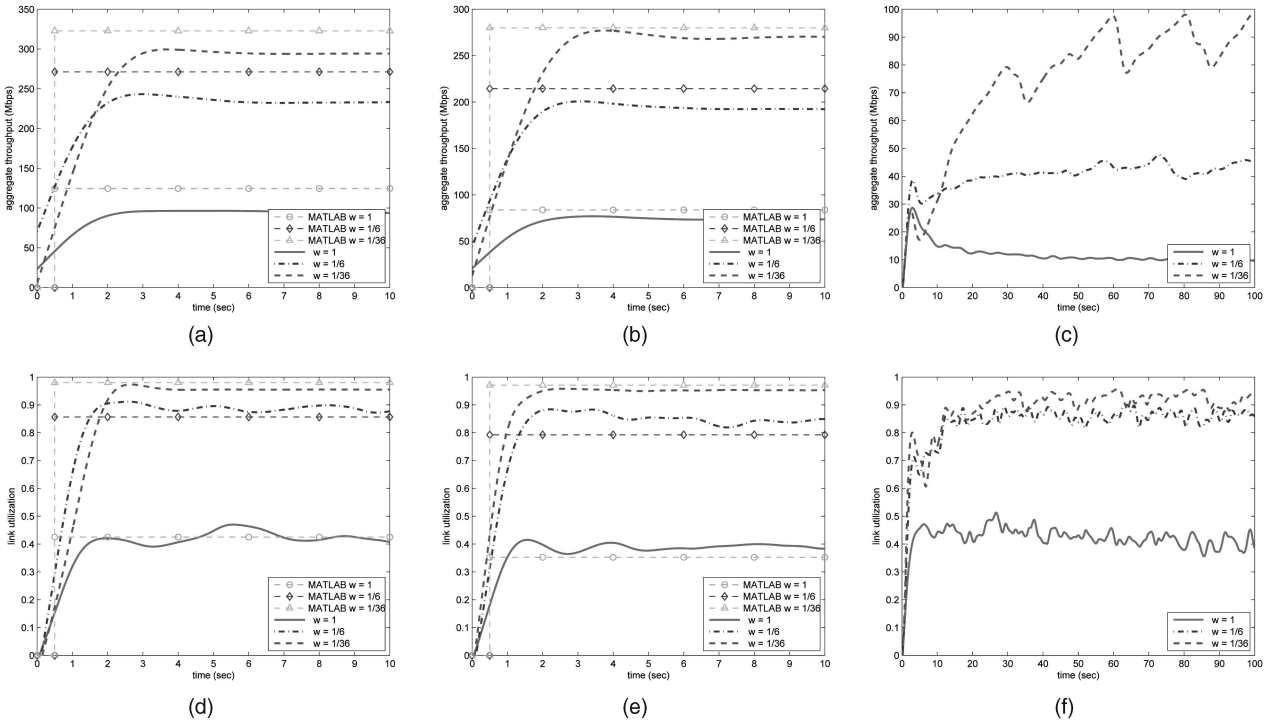

Fig. 7. Network throughput and link utilization with different w in NS2 simulations. (a) Access-core topology. (b) Abilene topology. (c) Cernet topology. (d) Access-core topology. (e) Abilene topology. (f) Cernet topology.

the link price update starts at time 0, and the link prices are updated every 20 ms, and the link flow rate update starts at time 0.5 s.

For the Access-core and Abilene topologies, all the links have a capacity of 100 Mbps and delay of 2 ms. For the Cernet topology, as shown in Fig. 2, the link capacities have been proportionally reduced by 100 times from their real values, to 100Mbps for backbone links and 25 Mbps for the rest.

To further understand the evolution and dynamics of LBMP, for Access-core and Abilene, we set $w = 1$ during the first 20 seconds, and $w = 1/36$ in the rest duration. Figs. 8a and 8b and Figs. 8d and 8e show the evolution of throughput and link utilization in the period 19.5-21 s, where dotted lines indicate the MATLAB simulation results. We can see that LBMP can quickly regulate the throughput and link utilization. Once the network becomes stable, the actual aggregate throughput and link utilization the network achieved are well consistent with the ideal numerical results.

6.1 Aggregate Throughput and Link Utilization

We first examine the aggregate throughput and link utilization of LBMP. For the three topologies, we plot the aggregate throughput and link utilization versus time with $w = 1$, $1/6$, and $1/36$ in Fig. 7. We can see that, for Access-core and Abilene, the simulation results are consistent with that of MATLAB under low delays. For the Cernet topology, given the relatively high delays, the flow rate oscillates for a while, and then becomes stable. In addition, the smaller the parameter w is, the higher the aggregate throughput and the link utilization are.

For the more realistic Cernet topology, we set the LBMP control parameters as follows: $w = 1/16$ for backbone links and $w = 1/16$ for other links from time 0.5 to 100 s, $w = 1$ for backbone links and $w = 1/16$ for other links from 100 to 200 s, and $w = 1/16$ for backbone links and $w = 1/16$ for other links from time 200 to 300 s. The simulation results are shown in Figs. 8c and 8f, where the solid line denotes the results of the whole network and the dashed line denotes the results of the backbone links. We see that the network can be easily tuned under LBMP through adjusting weight w for backbone, so as to regulate the throughput for all links. Specifically, when w of backbone links change, the link utilization of the entire network changes a little, but the backbone link utilization is reduced drastically, which largely relieves the potential congestions at these important links.

6.2 Selecting the Multiple Paths

There are many paths available between each source-destination pair. In this section, we study how many paths we need to provide LBMP for good performance, and how to select such paths. Given the representativeness of the Cernet topology, we will focus on it and set $w = 1/6$ in LBMP for all links from now on. In this simulation, three minimum-hop paths are chosen as possible paths for each source-destination pair. The shortest path (in terms of hop count) is referred to as the first path, the second shortest as the second path, and the third shortest as the third path.

First, we study how the number of paths available to each connection affects the performance and the network. We vary the number of paths available to each source-destination pair from 1 to 3. In Figs. 9a and 9d, we use 1, 2, 3-path to represent the routing strategies using the first path only, the first and second paths, and all the three paths, respectively. It is obvious that, when the number of paths increases, the aggregate throughput becomes higher and the link utilization becomes little higher. In particular, there is

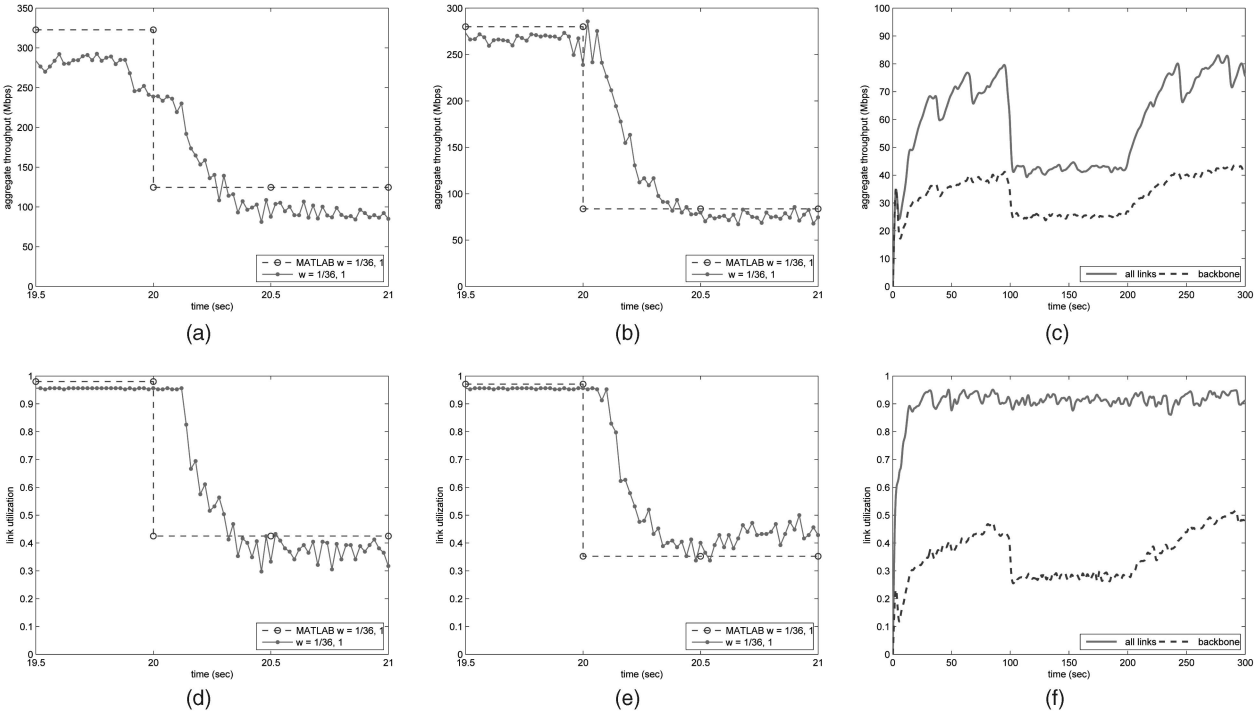

Fig. 8. Aggregate throughput and link utilization versus time with different w in NS2 simulations. (a) Access-core topology. (b) Abilene topology. (c) Cernet topology. (d) Access-core topology. (e) Abilene topology. (f) Cernet topology.

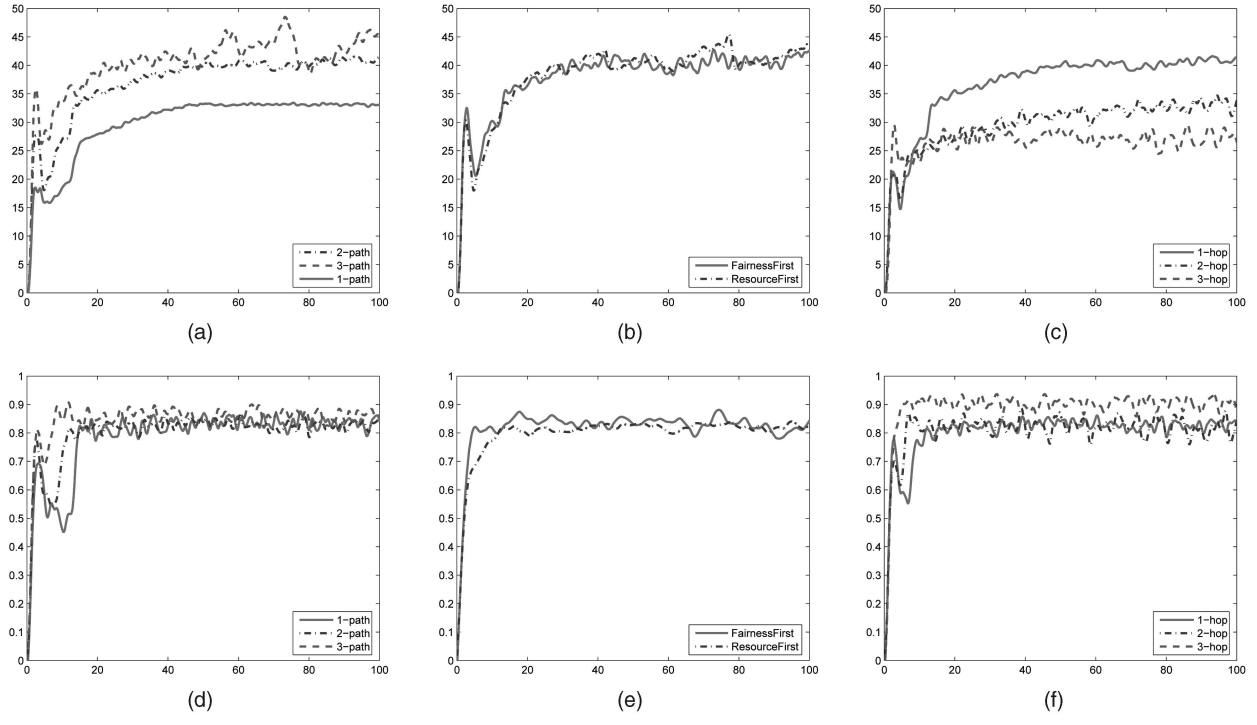

Fig. 9. Aggregate throughput and link utilization for Cernet topology versus time with different multipath. (a) Different number of paths. (b) Two paths or three paths. (c) Two paths with different hop count. (d) Different number of paths. (e) Two paths or three paths. (f) Two paths with different hop count.

significant improvement from using 1 path (the first path) to using 2 paths (the first and the second path). However, the improvement from 2 to 3 (the first, the second, and the third path) is marginal.

Second, we consider what will happen if some of connections use two paths and others use three paths? The six connections are divided into two groups and each group has three connections. Hop count of the two minimum-hop paths in the first group is greater than that in the second group. In Figs. 9b and 9e, the FairnessFirst denotes the results of each connection in the first group using the three minimum-hop paths, and the others use the

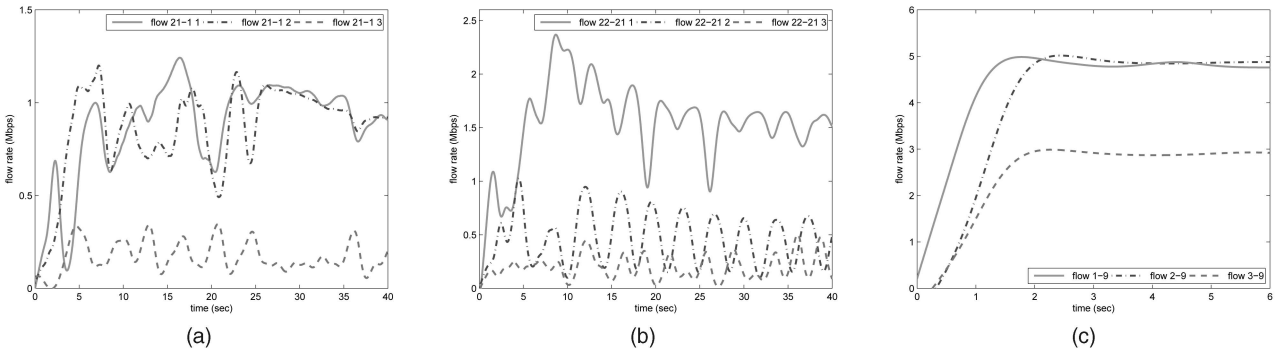

Fig. 10. Fairness of bandwidth sharing. (a) Rates of flows 1, 2, and 3 of source-destination pair 21-1 in Cernet. (b) Rates of flows 1, 2, and 3 of source-destination pair 22-21 in Cernet. (c) Rates of flows 1, 2, and 3 for the topology used in [10, Fig. 15a].

two minimum-hop paths. The ResourceFirst denotes the results of each connection in the first group using the two minimum-hop paths, and others using the three minimum-hop paths. The results show that the number of paths available to each connection has no observable impact on the performance.

Finally, we consider how the hop count of the paths affects the performance of the network. In Figs. 9c and 9f, we vary the hop count of the two paths available to each connection, where 1-hop denotes the results of each connection using the first and the second path; 2-hop denotes the results of each connection using the second and the third path; and 3-hop denotes the results of each connection using the second and the third path. It is obvious that, when the hop count decreases, the aggregate throughput becomes higher and the link utilization becomes lower.

Thus far, we find that selecting two (or three) shortest-hop paths per connection is sufficient for LBMP. This result is same as the one obtained by He. et al for TRUMP [11].

6.3 Fairness of Bandwidth Sharing

Finally, we examine the fairness of bandwidth sharing in LBMP. Consider the source-destination pair 21-1. The parameter pairs (number of hop, RTT) of its flows are configured in turn as (5, 134), (5, 134), and (7, 94). Corresponding simulation result is shown in Fig. 10a. From this figure, we see that the rates of the flows with the same number of hops and RTT are almost identical and decrease against the number of hops. We then consider the source-destination pair 22-21. The parameter pairs (number of hop, RTT) of its flows are configured in turn as (3, 65), (5, 135), and (7, 185), and the corresponding simulation result is shown in Fig. 10b. We can see that the flow with the least number of hops and RTT has the maximum flow rate, and the rest two have very low rates.

To further understand the fairness of LBMP, we examine it on the topology used by the original TRUMP evaluation [10, Fig. 15a], where the link (the bottleneck link) bandwidth between node 7 and node 8 is 100Mbps and others are 200Mbps. All the flows have a shared destination (node 9), and the sources are nodes 1, 2, and 3. The parameter pairs (number of hop, RTT) of the three flows are configured in turn as (3,30), (3,100), and (6,100). Fig. 10c plots the results for LBMP in this topology. We observe that flows 1 and 2, which have different RTT but identical number of hops on their paths, share bandwidth fairly. On the other hand, flow 3 with twice as many hops as flow 2, receives roughly half the bandwidth of flow 2. This indicates that flows with more hops occupy more network resources, and accordingly are allocated with less bandwidth. Such results are consistent with that obtained through TRUMP, and hence, the LBMP protocol ensures fairness as TRUMP does.

7 Conclusion and Future Work

This paper proposed LBMP, a Logarithm-Barrier-based Multipath Protocol for Internet traffic management. LBMP jointly considers user utility and routing/congestion control. It translates the multipath utility maximization problem into a sequence of unconstrained optimization problems, with infinite logarithm barriers being deployed at the constraint bounds. We demonstrated that setting up barriers is much simpler than choosing traditional cost functions and, more importantly, it makes optimal solution achievable. We further demonstrated a practical distributed implementation of LBMP.

We evaluated the performance of LBMP through both numerical analysis and packet-level simulations. The results showed that LBMP achieves high throughput and fast convergence over diverse representative network topologies. Such performance is comparable to TRUMP and is often better; yet its optimality and convergence are theoretically guaranteed, and it is flexible in differentiating link weights.

There are many possible venues to enhance LBMP. We are particularly interested in a version of LBMP using *window-based* flow control to pace the transmission of packets [15]. In our ongoing work, we are also examining the the local stability of LBMP and designing dynamic adjustment algorithms for the control parameter $\mathbf{w}$.

Acknowledgments

The authors are grateful to the anonymous referees for their comments on this paper. This work has been supported in part by the NSFC Project(60970104), the 973 Project of China (2009CB320501), the 863 Project of China (2008AA01A326), and the Program for New Century Excellent Talents in University. J. Liu's work is supported by a Canada NSERC Discovery Grant and an MITACS NCE Project Grant.

REFERENCES

[1] S. Athuraliya and S.H. Low, "Optimization Flow Control with Newton-Like Algorithm," *J. Telecomm. Systems,* vol. 15, pp. 345-358, 2000.

[2] D.P. Bertsekas, *Nonlinear Programming,* second ed. Athena Scientific, 1999.

[3] M. Chiang, "Balancing Transport and Physical Layer in Wireless Multihop Networks: Jointly Optimal Congestion Control and Power Control," *IEEE J. Selected Areas in Comm.,* vol. 23, no. 1, pp. 104-116, Jan. 2005.

[4] M. Chiang, S.H. Low, A.R. Calderbank, and J.C. Doyle, "Layering as Optimization Decomposition: A Mathematical Theory of Network Architectures," *Proc. IEEE,* vol. 95, no. 1, pp. 255-312, Jan. 2007.

[5] A.L.H. Chow, L. Golubchik, J.C.S. Lui, and W.-J. Lee, "Multipath Streaming: Optimization of Load Distribution," *Performance Evaluation,* vol. 62, no. 4, pp. 417-438, 2005.

[6] B. Fortz and M. Thorup, "Optimizing OSPF Weights in a Changing World," *IEEE J. Selected Areas in Comm.,* vol. 20, no. 4, pp. 756-767, May 2002.

[7] H. Han, S. Shakkottai, C.V. Hollot, R. Srikant, and D. Towsley, "Multi-Path TCP: A Joint Congestion Control and Routing Scheme to Exploit Path Diversity on the Internet," *IEEE/ACM Trans. Networking,* vol. 14, no. 6, pp. 1260-1271, Dec. 2006.

[8] J. He, M. Bresler, M. Chiang, and J. Reford, "Towards Robust Multi-Layer Traffic Engineering Optimization of Congestion Control and Routing," *IEEE J. Selected Areas in Comm.,* vol. 25, no. 5, pp. 868-880, June 2007.

[9] J. He, M. Suchara, and M. Chiang, "Rethinking Internet Traffic Management: From Multiple Decompositions to a Practical Protocol," *Proc. ACM Int'l Conf. Emerging Networking Experiments and Technologies (CoNEXT '07),* 2007.

[10] J. He, M. Suchara, M. Bresler, J. Rexford, and M. Chiang, "From Multiple Decompositions to TRUMP: Traffic Management Using Multipath Protocol," to be published in *Elsevier Computer Networks.*

[11] J. He and J. Rexford, "Towards Internet-Wide Multipath Routing," *IEEE Network Magazine,* vol. 22, no. 2, pp. 16-21, Mar. 2008.

[12] F. Kelly, A. Maulloo, and D. Tan, "Rate Control for Communication Networks: Shadow Prices, Proportional Fairness and Stability," *J. Operational Research Soc.,* vol. 49, no. 3, pp. 237-252, 1998.

[13] F. Kelly, *The Mathematics of Traffic in Network.* Princeton Univ. Press, 2005.

[14] F. Kelly and T. Voice, "Stability of End-to-End Algorithms for Joint Routing and Rate Control," *ACM SIGCOMM Computer Comm. Rev.,* vol. 35, no. 2, pp. 5-12, 2005.

[15] S. Low, L. Peterson, and L. Wang, "Understanding Vegas: A Duality Model," *J. ACM,* vol. 49, no. 2, pp. 207-235, 2002.

[16] S. Low, "A Duality Model of TCP and Queue Management Algorithms," *IEEE/ACM Trans. Networking,* vol. 11, no. 4, pp. 525-536, Aug. 2003.

[17] S.C.M. Lee, J. Jiang, D.-M. Chiu, and J.C.S. Lui, "Interaction of ISPs: Distributed Resource Allocation and Revenue Maximization," *IEEE Trans. Parallel and Distributed Systems,* vol. 19, no. 2, pp. 204-218, Feb. 2008.

[18] X. Lin and N. Shroff, "Utility Maximization for Communication Networks with Multipath Routing," *IEEE Trans. Automatic Control,* vol. 51, no. 5, pp. 766-781, May 2006.

[19] J. Mo and J. Walrand, "Fair End-to-End Window-Based Congestion Control," *IEEE/ACM Trans. Networking,* vol. 8, no. 5, pp. 556-567, Oct. 2000.

[20] R. Srikant, *The Mathematics of Internet Congestion Control.* Birkhauser, 2004.

[21] D. Palomar and M. Chiang, "Alternative Decompositions for Distributed Maximization of Network Utility: Framework and Applications," *IEEE Trans. Automatic Control,* vol. 52, no. 12, pp. 2254-2269, Dec. 2007.

[22] T. Voice, "Stability of Multipath Dual Congestion Control Algorithm," *IEEE/ACM Trans. Networking,* vol. 15, no. 6, pp. 1231-1239, Dec. 2007.

[23] J. Wang, L. Li, S. Low, and J. Doyle, "Can TCP and Shortest-Path Routing Maximize Utility," *Proc. IEEE INFOCOM,* pp. 2049-2056, Apr. 2003.

[24] J. Wang, D. Wei, and S. Low, "Cross-Layer Optimization in TCP/IP Networks," *IEEE/ACM Trans. Networking,* vol. 13, no. 3, pp. 568-582, June 2005.

[25] D. Wei, C. Jin, S. Low, and S. Hegde, "FAST TCP: Motivation, Architecture, Algorithms, Performance," *IEEE/ACM Trans. Networking,* vol. 14, no. 6, pp. 1246-1259, Dec. 2006.

[26] Abilene Backbone. http://www.internet2.edu/network/, 2010.

[27] China Education and Research Network. http://www.edu.cn/english_1369/index.shtml, 2010.

Ke Xu (M'02-SM'09) received the BS, MS, and PhD degrees in computer science from Tsinghua University, China, in 1996, 1998, and 2001, respectively. Currently he is a full professor in the Department of Computer Science of Tsinghua University. His research interests include next generation Internet, traffic management, switch and router architecture, and P2P and overlay network. He is a senior member of the IEEE and a member of the ACM.

Hongying Liu received the PhD degree in mathematics from Xidian University, Xi'an, China, in 2000. She has been an associate professor of mathematics of systems at the Beijing University of Aeronautics and Astronautics (BUAA), since 2003. Her research interests include optimization, game theory, and applied probability, focusing on applications in networks and statistical signal processing.

Jiangchuan Liu (S'01-M'03-SM'08) received the BEng degree (cum laude) from Tsinghua University, Beijing, China, in 1999, and the PhD degree from The Hong Kong University of Science and Technology, in 2003, both in computer science. He was a recipient of Microsoft Research Fellowship (2000), a recipient of Hong Kong Young Scientist Award (2003), and a co-inventor of one European patent and two US patents. He coauthored the Best Student Paper of the IWQoS'08 and the Best Paper (2009) of the IEEE Multimedia Communications Technical Committee (MMTC). He is currently an associate professor in the School of Computing Science, Simon Fraser University, British Columbia, Canada, and was an assistant professor in the Department of Computer Science and Engineering at The Chinese University of Hong Kong from 2003 to 2004. His research interests include multimedia systems and networks, wireless ad hoc and sensor networks, and peer-to-peer and overlay networks. He is an associate editor of the *IEEE Transactions on Multimedia*, and an editor of the *IEEE Communications Surveys and Tutorials*. He is a senior member of the IEEE and a member of the Sigma Xi.

Jixiu Zhang received the BEng degree from Tsinghua University, Beijing, China, in 2004. He is currently a master student at the School of Software and Microelectronics, Peking University. His research interests include traffic control and network security.