

# Joint Traffic Blocking and Routing under Network Failures and Maintenances

Chao Liang

ECE Dept.

Polytechnic University

Brooklyn, NY, 11201

Email: cliang@photon.poly.edu

Zihui Ge

AT&T Labs - Research

180 Park Avenue, POB 971,

Florham Park, NJ 07932

Email: gezihui@research.att.com

Yong Liu

ECE Dept.

Polytechnic University

Brooklyn, NY, 11201

Email: yongliu@poly.edu

**August 1, 2006**

## Abstract

Under device failures and maintenance activities, network resources reduce and congestion may arise inside networks. As a result, users experience degraded performance on packet delays and losses. Traditional approaches focused on rerouting traffic to alleviate network congestion and improve users' performance. However, due to the network capacity reduction, traffic rerouting alone cannot always satisfy users' performance requirements. In this paper, we study a dual approach that combines traffic blocking (rate-limiting) at the edge of a network and traffic rerouting inside the network. While ingress traffic blocking reduces the utility of network users, overall network performance can be significantly improved due to alleviated congestion and shorter routing paths. Therefore, it is important to design ingress traffic blocking and routing jointly to achieve a good balance between the two factors. Working towards this goal, we formulate a joint ingress blocking and routing optimization problem. We develop mechanisms to introduce blocking and routing differentiations among users with different service priorities and with different level of impact to network congestions. Using network topologies and traffic demands collected from a Tier-1 ISP network, we evaluated our approach. Our result shows that by blocking only a small fraction of traffic, one can greatly reduce network congestion under severe failures and maintenance activities. Our solution efficiently identifies the optimal blocking and routing differentiation among heterogeneous users and achieves much better performance in comparison with proportional traffic blocking. The proposed algorithms can be easily adopted by network service providers in their traffic engineering practices.

## I. INTRODUCTION

In a large scale system such as an ISP or an enterprise network, it is unrealistic to assume or to ensure the well-functioning of all components. Unexpected failures, such as a cut on a fiber or a conduit, a breakdown of network device (electronic or optical amplifier, line card, port, router, etc.) may occur at any time and can take a few hours up to several days to repair. In addition, important maintenance activities, such as a router IOS upgrade, can also take network resources out of service in the meantime. Therefore, it is important to design the network topology and select the routing strategy in a way such that the network can be robust to sporadic component failures, even though it means added cost and reduced efficiency in many cases. On the other hand, it is equally important to have the optimal adaptation strategy so that network resources can be reallocated to best accommodate traffic demand under failure conditions. This is done by adjusting the routing of the traffic in the presence of failures.

There are two classes of intra-domain routing protocols that are commonly used in the Internet. The first class is link-state-routing, represented by Open Shortest Path First (OSPF [1]) and Intermediate System-to-Intermediate System (IS-IS [2]) routing. In OSPF and IS-IS, each link is associated with a positive weight and traffic is routed along the shortest paths defined by the link weights. In cases of ties where several outgoing links are on the shortest paths to the destination, the flow is split evenly among them. In the second class, represented by Multi-protocol Label Switching (MPLS [3]), routing can be more flexibly defined. In MPLS, routing path and splitting fraction can be arbitrarily chosen based on the source and destination of the traffic flows. In both cases, routing configurations (link weights, splitting fractions) are determined in tune with the "normal" traffic and network conditions. Under failures or maintenance activities, network bandwidth resource may be greatly reduced. Even though routing protocols can converge to new routes that bypass failures, the routes are not tailored for the surviving bandwidth resource, causing

congestion inside the network. As a result, network users would experience performance degradation on packet delays and losses.

To address this problem, network operators have developed proactive and reactive countermeasures. For planned maintenance activities, a set of routing configurations can be computed based on the expected surviving bandwidth and demand matrix. These new configurations are then deployed in network before the planned intrusions take place. For hardware failures, this procedure is triggered by network alarms and operation tickets in a reactive manner, where the up-to-date network topology and capacity information is used in deriving new routing configurations that alleviate network congestion.

Adapting to new routing paths under network capacity losses allows full utilization of the available bandwidth. However, in some cases, surviving network capacity can no longer accommodate the traffic demand due to serious failures that take down a significant portion of the network bandwidth. As congestion is unavoidable in such cases, significant packet delays and massive packet losses on congested links are expected. An alternative to this unavoidable traffic loss at the internal network links is to drop the traffic at the edge of the network. This is achieved by configuring rate limiting ACLs (access control list) at the periphery of the network. In this way, traffic that would route through the congested links and hence are doomed lost can be blocked as early as possible. On the other hand, even if in some cases a set of the routes can be found to accommodate the overall traffic demand, the quality of the resultant routing paths (e.g., path delay) may deteriorate. For many delay sensitive traffic (e.g., voice of IP, gaming), such degradation may be intolerable. Enabling ingress blocking for some low priority traffic, although reducing network utility due to the blocked traffic, can lead to improved delay performance of the feasible routing paths, therefore improves the overall quality of the network service.

In this paper, we study the problem of identifying the optimal congestion mitigation strategy under critical network conditions by utilizing both ingress traffic blocking and traffic routing. Our goal is to minimize the impact of network capacity reduction due to failures and maintenances on the overall network service quality. In particular, we formulate a joint ingress blocking and routing optimization problem that identifies the best balance between the network performance experienced by the admitted traffic and the utility loss due to the blocked (rate-limited) demand. To the best of our knowledge, this is the first work that jointly consider both rate control for network demand pairs and adaptive route optimization under network failures and maintenances. Our formulation includes both blocking and routing differentiation among user demands with different service priorities. We propose solution techniques for both the source-destination (MPLS) and the link-state (OSPF/IS-IS) intra-domain routing protocols. By using realistic network and traffic demand data collected from a Tier-1 ISP network, we evaluate our approach and study the tradeoff between traffic blocking and the path delay quality. We find that our proposed solution can effectively identify the desired tradeoff. Compared with the state-of-the-art approach where only traffic rerouting is considered under severe network capacity loss, our solution can significantly reduce (by 60% in our experiment) the average path delay of the surviving network by blocking a small amount (4%) of low priority traffic. This is encouraging since it suggests that by considering this new dimension of control (ingress blocking), the overall network performance under failures and maintenances can be greatly enhanced.

The remainder of the paper is organized as follows. In Section II, we review related work. We formulate the joint ingress blocking and routing problem in Section III. The methodology of solving the joint optimization problem is described in Section IV. We present in Section V experimental results to demonstrate the effectiveness of the proposed approach. We conclude the paper and list directions for future work in Section VI.

## II. RELATED WORK

Most intra-domain routing in the Internet is carried out by dynamic routing protocols, such as OSPF [1], IS-IS [2], and MPLS [3]. Ideally, an optimal routing should simultaneously provide good QoS perceived by end users (e.g., small delay, low loss, and high throughput) and good network-wide performance (e.g., balanced load, robustness against network failures, and resilience to malicious attacks). Even though the routing optimization problem has been well-studied in literature [4], [5], it is still very difficult to operate a large network in its “optimal” state. When the network topology is large, it is challenging to calculate and implement optimal routes, especially in a distributed fashion. Approximation algorithms have been proposed to find routes that achieve close to optimal performances [6], [7]. Furthermore, it has been shown that Internet traffic presents high variability in multiple time-scales [8], [9]. Dynamic update of optimal routes in response to traffic changes is prohibitively expensive [10]. Meanwhile, timely

estimation of traffic demand matrix is a challenging task [11], [12], [13], where the derived traffic matrix usually has limited accuracy. To address these issues, extensive research has been made on identifying routes that are robust to traffic variations [10], [14], [15] and traffic demand estimation errors [16], [17].

Random router and link failures are encountered on a daily basis in large IP networks [18]. Network operators also take down routers and links for routine network maintenances. The state-of-the-art solution for network under failures and maintenances is to reroute traffic, where network routes are re-configured locally or globally to route the original traffic demand in the reduced network topology. Backup paths can be pre-configured for fast routing restoration [19], [20], [21], [22]. Research has also been conducted to make routing configurations, such as link weights, robust to network resource reduction [23], [24] so that routing algorithms can quickly converge to new routes under failures and maintenances. However, due to the reduction in network resources, users experience degraded performance after routing reconfigurations. Large scale failures can cause the network incapable of supporting the supplied traffic demand, leading to long network delays and high packet losses. To avoid excessive network congestion, our solution blocks traffic demand at the network ingress points, so as to maintain good performance for admitted users. To the best of our knowledge, this is the first work on combining traffic blocking and rerouting for critical network conditions.

Ingress traffic blocking is essentially a rate control problem for network demand pairs. Classic network rate control studies the optimal rate allocation for users with *fixed* routes. A utility-price based framework was proposed to study the fairness and stability of congestion control schemes [25], [26], [27], [28]. Solutions to the distributed optimization problem with different utility functions lead to different levels of fairness among users, such as *max-min fairness* and *proportional fairness*. Recently, the interaction between TCP congestion control and dynamic IP routing has been studied as a joint utility maximization problem [29], [30]. Joint rate control, routing and resource allocation algorithms have been proposed to achieve the optimal resource utilization in wireless and sensors networks [31], [32], [33], [34], [35]. We study the joint ingress blocking and routing problem in the context of network failures and maintenances. The goal is to maintain good network-wide delay performance at the price of minimal network traffic loss. We also consider the blocking and routing differentiation among users with different service priorities.

### III. FORMULATION: JOINT BLOCKING AND ROUTING

In this section, we formulate the joint optimization problem with both ingress blocking and routing. We start with a baseline model designed for MPLS type of flow routing. We then augment the baseline model with additional constraints and variations of its objective function to introduce ingress blocking and routing differentiation among users with different service priorities. The baseline model is also simplified to utilize source-destination based routing. At the end of the section, we show how to use the baseline model in link-state shortest routing.

#### A. MPLS Baseline Model

Consider a network  $G_0 = (V_0, E_0)$  shared by a set of users  $D$ . Let  $I_i$  and  $E_i$  denote the ingress and egress points for user  $i \in D$  respectively. The expected traffic volume of user  $i$  is  $d_i^0$ . Under failures or maintenances, the network topology is reduced to  $G = (V, E)$ , with surviving capacity  $c_j$  for each link  $j \in E$ . To maintain a good network-wide delay performance, ingress routers conduct admission control to block excessive traffic. Ingress router  $I_i$  admits  $d_i \leq d_i^0$  amount of traffic from user  $i$ . We quantify users' satisfaction using utility functions of their admitted traffic rates. Let  $U_i(d_i)$  denote user  $i$ 's utility at rate  $d_i$ . We assume  $U_i(\cdot)$  is an increasing and concave function. It follows that user  $i$ 's utility loss due to traffic blocking is  $U_i(d_i^0) - U_i(d_i)$ . Let  $\mathcal{B} \triangleq \{d_i, i \in D\}$  be the vector of admitted traffic for all users. The aggregate network utility loss is

$$\mathcal{J}_U(\mathcal{B}) = \sum_{i \in D} (U_i(d_i^0) - U_i(d_i)). \quad (1)$$

Since  $U_i(d_i)$  is increasing and concave in  $d_i$ ,  $\mathcal{J}_U(\mathcal{B})$  is decreasing and convex in  $\mathcal{B}$ .

Admitted traffic will be routed to minimize network-wide delay. We start with a baseline model for MPLS type of flow routing. Link weights based shortest path routing, such as OSPF/IS-IS, will be discussed later. For a flow routing, we denote by  $\tau_i^l$  the traffic rate of user  $i$  allocated to link  $l$ ,  $l \in E$ . A legitimate set of routes for user  $i$

$\{\tau_i^l, l \in E\}$  should satisfy *flow conservation* on all nodes in the network:

$$\sum_{l \in \mathcal{O}(v)} \tau_i^l - \sum_{l \in \mathcal{P}(v)} \tau_i^l = \begin{cases} d_i & \text{if } v = I_i \\ -d_i & \text{if } v = E_i \\ 0 & \text{otherwise} \end{cases}, \quad (2)$$

where  $\mathcal{O}(v)$  is the set of outbound links and  $\mathcal{P}(v)$  is the set of inbound links at node  $v$ .

Given all users' routes  $\mathcal{R} \triangleq \{\tau_i^l, i \in D, l \in E\}$ , the total traffic rate on link  $l$  is  $f_l = \sum_{i \in D} \tau_i^l$ . Similar to [5], [4], we approximate the congestion delay on link  $l$  using  $M/M/1$  formula  $\frac{1}{c_l - f_l}$ . If the propagation delay on link  $l$  is  $p_l$ , the average end-to-end delay for user  $i$  is

$$\phi_i(\mathcal{B}, \mathcal{R}) = \frac{1}{d_i} \sum_{l \in E} \tau_i^l \left( \frac{1}{c_l - f_l} + p_l \right). \quad (3)$$

The network-wide aggregate delay is thus the weighted sum (by traffic volume) of all users, as shown below.

$$\mathcal{J}_D(\mathcal{B}, \mathcal{R}) = \sum_{i \in D} d_i \phi_i = \sum_{l \in E} \left( \frac{f_l}{c_l - f_l} + f_l p_l \right) \quad (4)$$

Obviously  $\mathcal{J}_D(\mathcal{B}, \mathcal{R})$  increases with the admitted traffic  $\mathcal{B}$  given any fixed routes  $\mathcal{R}$ .  $\mathcal{J}_D(\mathcal{B}, \mathcal{R})$  is also a convex function of  $\mathcal{B}$  and  $\mathcal{R}$ .

Our objective is to find a joint solution of blocking  $\mathcal{B}$  and routing  $\mathcal{R}$  that achieves a good balance between the network delay  $\mathcal{J}_D$  and utility loss  $\mathcal{J}_U$ . Working towards this, we formulate a weighted sum optimization problem:

$$\min_{\{\mathcal{B}, \mathcal{R}\}} (1 - \alpha) \mathcal{J}_D(\mathcal{B}, \mathcal{R}) + \alpha \mathcal{J}_U(\mathcal{B}, \mathcal{R}), \quad (5)$$

subject to (2) and

$$d_i \leq d_i^0, \forall i \in D, \quad (6)$$

$$f_l \leq c_l, \forall l \in E, \quad (7)$$

$$\tau_i^l \geq 0, \forall i \in D, \forall l \in E, \quad (8)$$

$$d_i \geq 0, \forall i \in D, \quad (9)$$

where the weight  $0 < \alpha < 1$  controls the balance between the network delay performance and utility loss in the objective function. Constraints (6) state that admitted traffic of a user cannot exceed offer demand. Constraints (7) state that the aggregate traffic over a link cannot exceed the surviving capacity of the link. From (1) and (4), the objective function is convex in  $\mathcal{B}$  and  $\mathcal{R}$ . The constraints defined in (2) and (6)~(9) are linear. Overall, system (1)~(9) defines a convex optimization problem.

For a given  $\alpha$ , by solving this convex optimization problem, we can obtain the optimal joint solution of blocking  $\mathcal{B}^*(\alpha)$  and routing  $\mathcal{R}^*(\alpha)$ . The network delay is  $\mathcal{J}_D^*(\alpha)$  and the utility loss is  $\mathcal{J}_U^*(\alpha)$ . We can tune  $\alpha$  to achieve the desired balance between the network delay and the utility loss. We will demonstrate this through experiments in Section V.

### B. Differentiation between Users

A simple blocking strategy is to block all users proportionally, i.e.,  $\mathcal{B}(\eta) \triangleq \{d_i = \eta d_i^0, 0 < \eta < 1, i \in D\}$ . Let  $\mathcal{R}^*(\eta)$  be the minimal delay routing under proportional blocking  $\mathcal{B}(\eta)$ :

$$\mathcal{R}^*(\eta) \triangleq \underset{\{\mathcal{R}\}}{\text{argmin}} \mathcal{J}_D(\mathcal{B}(\eta), \mathcal{R}).$$

Under  $\mathcal{B}(\eta)$  and  $\mathcal{R}^*(\eta)$ , the network-wide delay is  $\mathcal{J}_D^*(\eta)$  and the aggregate user utility loss is  $\mathcal{J}_U^*(\eta)$ . Since  $\mathcal{J}_D^*(\eta)$  increases with  $\eta$  and  $\mathcal{J}_U^*(\eta)$  decreases with  $\eta$ , one approach to achieving balance between  $\mathcal{J}_D^*(\eta)$  and  $\mathcal{J}_U^*(\eta)$  is to try to satisfy the network delay performance requirement by blocking as little traffic as possible. In this case, the joint blocking and routing problem can be formulated as:

$$\max_{0 < \eta \leq 1} \eta, \quad \text{subject to } \mathcal{J}_D^*(\eta) \leq B_D,$$

where  $B_D$  is the upper bound requirement on the network-wide delay. In proportional blocking, the optimal blocking ratio  $\eta^*$  is independent of the utility functions of individual users and can be identified efficiently through binary search.

In real network environments, network users are heterogeneous. Network service providers give them different service priorities according to the service level agreement (SLA). Users with higher priorities should be blocked less and assigned to routes with smaller delays. In addition, users with the same priority have different ingress and egress points. They contribute differently to network congestion even if they have the same traffic rates. Consequently they should be blocked (i.e., rate-limited) and routed differently. Therefore, user differentiation in blocking and routing is necessary to achieve the optimal network-wide performance.

**Blocking Differentiation.** To introduce blocking differentiation between users, we introduce different lower bounds for the admitted traffic rates of users with different priorities. In other words, we modify the constraint in (6) to  $d_i^m \leq d_i \leq d_i^0$  for user  $i \in D$ . In the extreme case, if we set  $d_i^m = d_i^0$ , no traffic of user  $i$  can be blocked. The variable  $d_i$  will be replaced by the constant  $d_i^0$  in (5). We can also introduce blocking differentiation in our framework by manipulating user utility functions. Assign a “higher-value” utility function to a user will make him less likely to be blocked at his ingress point.

**Routing Differentiation.** To implement routing differentiation, we allow different users between the same pair of ingress-egress points to take different routes. In addition, we use a *weighted* network delay performance:

$$\mathcal{J}_D \triangleq \sum_{i \in D} w_i d_i \phi_i = \sum_{l \in E} \left( \frac{z_l}{c_l - f_l} + z_l p_l \right),$$

where  $w_i$  is the weight for user  $i$ , and  $z_l = \sum_{i \in D} w_i \tau_i^l$ . A user with a higher weight is likely to be assigned to a shorter route. The optimization problem changes to

$$\min_{\{d_i, \tau_i^l\}} (1 - \alpha) \sum_{l \in E} \left( \frac{z_l}{c_l - f_l} + z_l p_l \right) + \alpha \sum_{i \in D} (U_i(d_i^0) - U_i(d_i)). \quad (10)$$

The linear fraction in the first term makes it no longer convex. To obtain an approximation solution, one may still approximate the first term by a combination of linear functions of  $(z_l, f_l)$  and the second term by a set of linear functions of  $d_i$ . Then (10) can be solved approximately as a linear programming problem.

Even without delay performance weights, it is possible that the joint optimization returns multiple paths for a group of users sharing the same ingress-egress pair. Then routing differentiation within this group can still be achieved by assigning users with higher priorities to shorter paths. We will elaborate on this in the following section.

### C. Simplification: Source-Destination Routing

In an MPLS routing formulation, every user  $i$  has a set of routing variable  $\{\tau_i^l, l \in E\}$ . Thus, different user demands with the same ingress-egress points can have different routing paths depending on their SLA. However, in a source-destination based routing, all users sharing the same ingress-egress point pair are enforced to use the same set of routes. Ideally, one would like to adopt source-destination based routing in the optimization problem (5), since this largely reduces the number of routing variables. The concern is whether we sacrifice the quality of the routing solution by doing so.

In fact, as we proved in Appendix A, if there is no differentiation in users’ delay performance, the optimal source-destination routing can achieve the same performance as the optimal MPLS routing. For source-destination routing, we can adopt routing variables  $\{x_{(s,d)}^l, (s,d) \in V \times V, l \in E\}$ , where  $x_{(s,d)}^l$  denotes the fraction of traffic from ingress node  $s$  to egress node  $d$  routed to link  $l$ . Then flow routing variables  $\{\tau_i^l, i \in D, l \in E\}$  in (5) can be replaced by  $\tau_i^l = d_i x_{(I_i, E_i)}^l$ . However, optimizing variables  $d_i$  and  $x$  appear in product forms and makes it difficult to solve. We can put all users sharing the same ingress-egress point pair  $(s, d)$  into one group  $D(s, d) = \{i \in D | I_i = s \text{ and } E_i = d\}$ . For each group  $D(s, d)$ , we define a group utility function as the solution of the following optimization problem

$$U_{(s,d)}(z) \triangleq \max_{\{d_i, i \in D(s,d)\}} \sum_{i \in D(s,d)} U_i(d_i), \quad (11)$$

subject to

$$\sum_{i \in D(s,d)} d_i \leq z, \quad (12)$$

$$d_i^m \leq d_i \leq d_i^0. \quad (13)$$

$U_{(s,d)}(z)$  represents the maximal aggregate utility if the total traffic rate allocated to user group  $D(s,d)$  is  $z$ . It can be shown  $U_{(s,d)}(z)$  is an increasing and concave function of  $z \in \left[ \sum_{i \in D(s,d)} d_i^m, \sum_{i \in D(s,d)} d_i^0 \right]$ , if  $U_i(d_i)$  is concave in  $d_i \in [d_i^m, d_i^0]$ ,  $\forall i \in D(s,d)$  (See proof in Appendix B). We then treat all users in  $D(s,d)$  as one ‘‘super’’ user with a variable rate  $z \in \left[ \sum_{i \in D(s,d)} d_i^m, \sum_{i \in D(s,d)} d_i^0 \right]$  and utility function  $U_{(s,d)}(z)$ . We can first solve the blocking and routing problem as in (5) to obtain the optimal group rate  $z_{(s,d)}^*$  and routes  $\{x_{(s,d)}^l\}$  for users in that group. The admitted traffic rates for all users in group  $D(s,d)$  are:

$$\{d_i^*, i \in D(s,d)\} = \underset{\sum_{i \in D(s,d)} d_i = z_{(s,d)}^*}{\operatorname{argmax}} U_i(d_i)$$

and the flow routing variables for user  $i$  can be calculated as:

$$\tau_i^l = x_{(s,d)}^l d_i^*.$$

If there are multiple paths between an ingress-egress pair, one can assign the shorter ones to users with higher priorities to achieve routing differentiation. Table I shows the algorithm that constructs a set of routes for a given ingress-egress pair from the routing variables in the increasing order of the path delay. The set of links carrying non-zero traffic of the specified user group  $G$  can be determined from the optimization results ( $\tau^l$ ). In each iteration, the shortest path is found from this link set. The traffic rate using this path is set as the minimum of all routing variables on the path. We then update the routing variables on the path by deducting this traffic rate and remove the links whose associated routing variable becomes zero from the candidate link set. This repeats for the next shortest path until the link set becomes empty. Finally, we assign paths to users in the reverse order of their priority (shorter path for higher priority).

TABLE I  
MULTI-PATH ALLOCATION ALGORITHM

<b>Definition:</b>	
$G$	specified user group
$\vec{E}$	set of links carrying traffic of the user group $G$
$\phi_l$	link delay on link $l$ , $l \in \vec{E}$
$\tau^l$	group traffic rate on link $l$ , $l \in \vec{E}$
<b>Routine:</b>	
while $\vec{E} \neq \{\emptyset\}$	
$M =$ subset of links on OSPF path based on delay $\{\phi_l, l \in \vec{E}\}$	
$\tau_{min} = \min \{\tau^l, l \in M\}$	
Output the traffic rate and path( $\tau_{min}, M$ )	
for each $l \in M$	
$\tau^l = \tau^l - \tau_{min}$	
if $\tau^l$ equal to 0	
Update $\vec{E} = \vec{E} - l$	

#### D. Link State Routing

Link weight based shortest path routing algorithms, such as OSPF [1]/IS-IS [2], are widely used in network traffic engineering practice. It has been shown in [6] that link weight optimization is an NP-hard problem. However, heuristic algorithms can often efficiently find link weights that achieve performance that is close to the optimal of its counterpart (i.e., MPLS routing). Based on this observation, we develop an approximation algorithm for the joint blocking and routing problem under OSPF/IS-IS.

While the coupling between traffic blocking and routing largely increases the complexity of the heuristic search, we can decouple this complexity by adopting our MPLS baseline model to find the optimal blocking configuration

first. More specifically, let  $\mathcal{B}^*(\alpha)$  and  $\mathcal{R}^*(\alpha)$  be the optimal MPLS solution for some  $\alpha$  that achieves the desired level of balance between the network delay  $\mathcal{J}_D^*(\alpha)$  and utility loss  $\mathcal{J}_U^*(\alpha)$ . A heuristic algorithm, such as that in [6], is consequently applied to search for the best link weights to route the “supplied” traffic demand  $\mathcal{B}^*(\alpha) = \{d_i^*, i \in D\}$  (i.e., the admitted traffic demand calculated by the MPLS model). Network delay under the routing returned by the heuristic algorithm might be higher than that of the optimal MPLS routing  $\mathcal{J}_U^*(\alpha)$ . This can be addressed by introducing a delay margin in the MPLS baseline model to compensate for the performance gap between the optimal MPLS routing and the heuristic link-state routing.

#### IV. SOLUTION METHODOLOGY

To solve the joint blocking and routing problem formulated in the previous section, we first need to choose appropriate user utility functions. We assume the number of end-to-end connections in a demand pair is proportional to its expected traffic volume  $d_i^0$ . Under this assumption, we choose utility functions as

$$U_i(d_i) \triangleq M d_i^0 \left( 1 - \exp\left(-k \frac{d_i}{d_i^0}\right) \right), \quad (14)$$

with  $M$  and  $k$  are constants. One interpretation is that the utility of a demand pair is the summation of the utilities of all connections in that demand pair, and the utility of a connection is an increasing and concave function of its traffic admission ratio. Consequently, larger demand pairs will have higher weights in the utility optimization.

In order to transform the convex optimization problem into a linear programming problem, we use piece-wise linear functions to approximate link congestion delays and user utility functions.

- *Congestion Delay*: The total congestion delay on link  $l$  is  $\theta_l = \frac{f_l}{c_l - f_l}$ , where  $c_l$  is its link capacity and  $f_l$  is its total traffic load. This link delay function is a convex and increasing function of link load. Similar to [6], we approximate it by a collection of piece-wise linear functions:

$$\theta_l = \max_{1 \leq i \leq 6} \left( k_i \frac{f_l}{c_l} + b_i \right),$$

where  $k_i = 2^{4i-2}$  and  $b_i = -2^{4i-2} + 5 * 2^{2i-2} - 1$ ,  $i \in \{1, \dots, 6\}$ .

- *Utility Loss*: For the utility function defined in (14), we choose  $M = \frac{25}{\sum_{i \in D} d_i^0}$  and  $k = 5$ . For traffic demand  $i$ , the utility loss function

$$L_i(d_i) = U_i(d_i^0) - U_i(d_i) = M d_i^0 \left( \exp\left(-k \frac{d_i}{d_i^0}\right) - \exp(-k) \right)$$

is a convex and decreasing of the traffic admission ratio. Similar to the congestion delay approximation, we approximate the utility loss function  $L_i(d_i)$  by a set of linear functions:

$$L_i(d_i) = M d_i^0 \max_{1 \leq m \leq 6} \left( k_m \frac{d(i)}{d^0(i)} + b_m \right),$$

where parameters  $\{(k_m, b_m), m \in \{1, \dots, 6\}\}$  determine the turning points of the piece-wise linear approximation. Figure 1 shows the approximation curve for the normalized utility loss as a function of the traffic blocking ratio.

With the above linear approximations, the joint blocking and routing optimization can be solved efficiently as a Linear Programming problem by the AMPL/CPLEX [36] toolkit.

#### V. EXPERIMENTS

##### A. Evaluation Network Settings

The network traffic data in our experiments are collected from AT&T’s North American commercial backbone network, which consists of tens of Point of Presence(PoPs), hundreds of routers, thousands of links. The PoP-level network topology on June 23, 2006 is used in our evaluation. The router-level topology is first obtained using the method [37]. We then reduce the router-level topology to a PoP-level topology by congregating the router-level links between the same pair of PoPs into a single PoP-level link. The capacity of a PoP-level link is set as the sum of the capacities of all underlying router-level links. The traffic matrixes are estimated using the *tomo-gravity* method [13], which can provide accurate estimates, especially for large traffic matrix elements. The final PoP-level

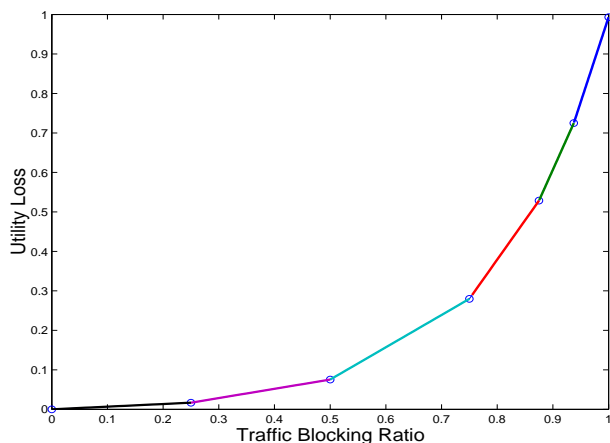


Fig. 1. Utility Loss Piece-wise Linear Approximation

traffic matrix contains around 400 Origin-Destination flows at rates ranging from tens of Mbps to tens of Gbps. We emulate network failures or maintenance activities by reducing link capacities either randomly or according to the shared risk groups [38] presented in the lower layer topology graph. In the next subsection, we present our result based on one synthetic failure setting where we identify the node with the highest link degree and cut the capacities of all its adjacent links by half. The result shown is representative for other severe network failure scenarios.

## B. Numerical Results

1) *Tradeoff between Traffic Blocking and Network Delay*: The parameter  $\alpha$  in the optimization objective function controls the balance between the network delay and utility loss from blocking. We vary the tradeoff parameter  $\alpha$  from 0.01 to 0.99 and solve the joint optimization problem numerically. We plot in Figure 2 the average traffic

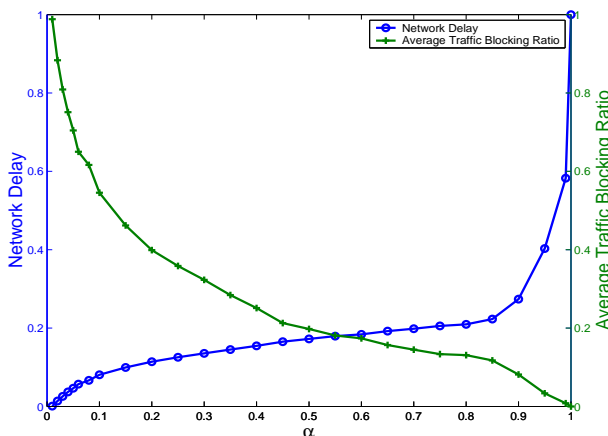


Fig. 2. Average Traffic Blocking Ratios and Network Delays at Different Tradeoff Parameters

blocking ratios and network delays as a function of  $\alpha$ , where the delay cost has been normalized by the network delay without ingress blocking. Recall that in equation (5), when  $\alpha$  is small, the user utility loss has a small weight in the overall objective function. As a result, a large portion of traffic has to be blocked in order to quickly bring down the network delay. As  $\alpha$  increases, the user utility loss takes a larger weight and less traffic is blocked. Consequently, the network delay increases. When  $\alpha$  approaches 1, the utility loss dominates the network delay in the joint optimization, the traffic blocking ratio quickly drops close to 0. Meanwhile, network delay shoots up dramatically. This corresponds to the state-of-the-art approach where only traffic rerouting is used in responses to network failures.

To illustrate the tradeoff between network delay and traffic blocking, we plot in Figure 3 network delay as a function of traffic blocking ratio. The horizontal dashed line represents the network delay before the failure. Figure 3 shows that the network delay drops quickly with a little bit of traffic blocking. With a blocking ratio of 4%,



the network delay can be reduced to the level before the failure. By blocking around 12% traffic, the network delay can be reduced about 5 times. This demonstrates that ingress traffic blocking is very effective in alleviating network congestion under failures and maintenances. One can significantly improve network delay performance by sacrificing a little on user utility loss. Furthermore, the small percentage of blocked traffic is from low priority users. The delay decreasing trend slows down when the blocking ratio gets larger.

By examining the tradeoff curve, a network operator can choose an appropriate  $\alpha$  parameter to achieve a desired level of balance between the network delay and traffic blocking. The solution of the corresponding joint optimization will tell them how much traffic to block for each demand pair and how to route admitted traffic in the networks.

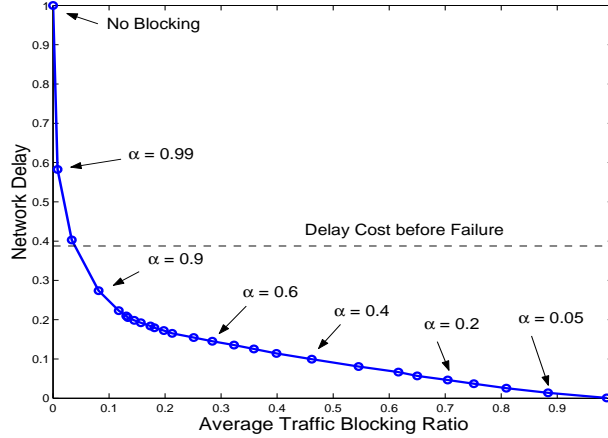


Fig. 3. Network Delay vs. Average Traffic Blocking Ratio

2) *Blocking Differentiation between Demand Pairs*: The objective of joint blocking and routing is to reduce the network delay with minimal traffic blocking. Depending on their traffic rate and ingress-egress points, different demand pairs contribute differently to network congestion. In the solution of the joint blocking and routing optimization, the demand pairs are blocked differently. The general trend is that demand pairs with larger end-to-end delays are likely to be blocked more than those with smaller delays. Figure 4 demonstrates the correlation between traffic blocking ratios and average delays of demand pairs at four different  $\alpha$  values. For each  $\alpha$ , after solving the joint optimization problem, we index all demand pairs in the increasing order of their end-to-end delays. We then plot their traffic blocking ratios. In these figures, traffic blocking ratios only take several discrete values. This is because we use a piece-wise linear function of blocking ratio to approximate the network utility loss. The solution of the resulted linear programming problem corresponds to the extreme points in the blocking and routing design space. From those plots, we observe a *positive* correlation between traffic blocking ratios and end-to-end delays of demand pairs. Demand pairs with larger end-to-end delays are likely to be blocked more than demand pairs with smaller delays. In Figure 4(a), the demand with the largest end-to-end delay is blocked more than 90%, the demand with the smallest delay is only blocked around 25%. With the increase of  $\alpha$ , the average traffic blocking ratio decreases. Figure 4 also shows that the maximal traffic blocking ratio decreases as  $\alpha$  increases.

These results suggest a rule for some heuristic algorithms to select demand pairs for blocking *without* explicitly solving the joint optimization problem: *blocking the long delay demand pairs first*. To identify those “long” demand pairs, one can calculate the optimal routing without any blocking to get the optimal routes and queuing delays on all the links. Based on the algorithm described in Table I, one can identify all paths of a demand pair and use the end-to-end delay on its longest path as its likelihood of being blocked. Figure 5 illustrates the correlation between the above defined longest path delays and the likelihoods of being blocked of demand pairs in our experiments. We first sort all demand pairs in the increasing order of their longest path delays. We solve the joint blocking and routing optimization at different  $\alpha$  values. For each  $\alpha$ , we calculate the average traffic blocking ratio  $y(\alpha)$ . If a demand pair  $x$  is blocked, we plot one dot at coordinates  $(x, y(\alpha))$  in Figure 5. The figure verifies that the optimal blocking strategy prefers to block demand pairs with larger delays first. However, the gaps between points on horizontal lines, especially for small average blocking ratios, also indicate that the optimal blocking strategy cannot be replaced by the simple heuristic algorithm that blocks demand in the decreasing order of longest path

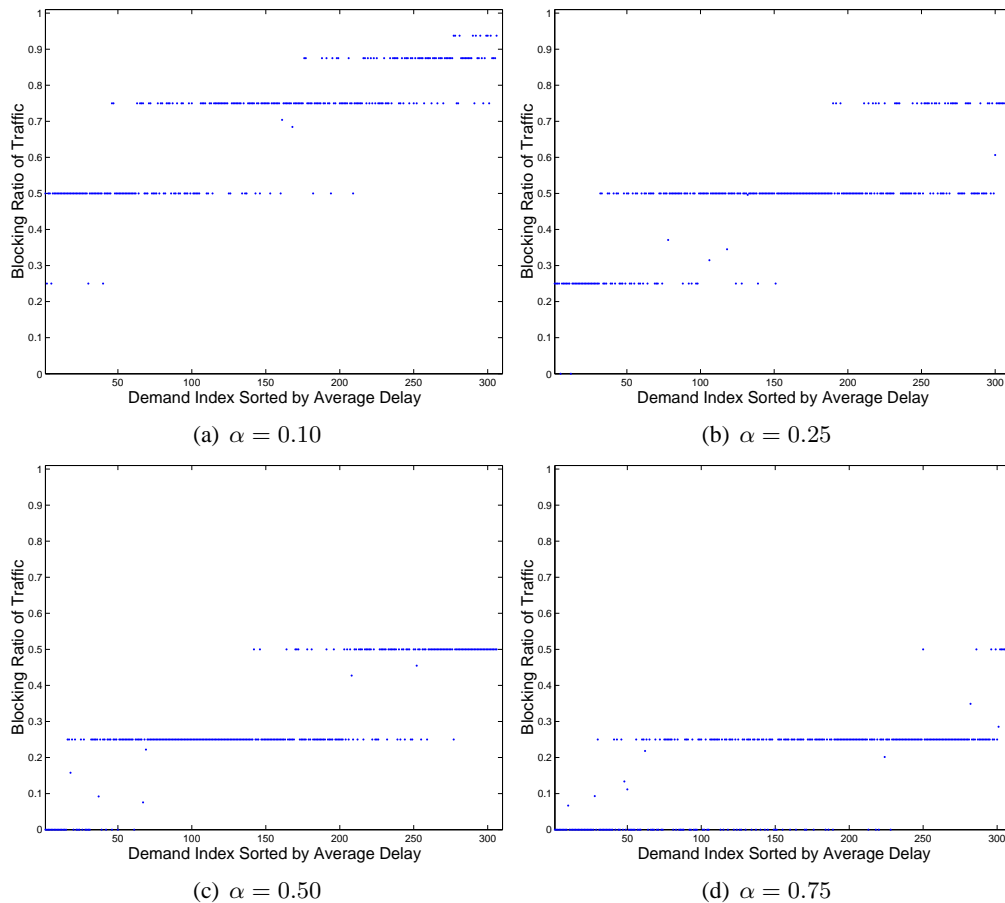


Fig. 4. Traffic Blocking Ratio v.s. Average End-to-End Delay

delays.

3) *Comparison with Proportional Blocking*: The simplest traffic blocking strategy is to block all demand pairs proportionally. To demonstrate the effectiveness of our joint blocking and routing method, we compare our results with the simple proportional blocking. After proportionally cutting traffic of all demands, we calculate the optimal routing for the blocked traffic matrix to get the network delay cost under proportional blocking. Figure 6 compares the network delay cost of the two methods with the same average traffic blocking ratio. We vary the traffic blocking ratio from 0 to around 0.55. From this figure, we can see our optimization method outperforms the simple proportional blocking. With the same average traffic blocking ratio, joint blocking and routing achieves lower network delay than proportional blocking. The difference is quite large when the blocking ratio is small. The gap closes down when more and more traffic get blocked. Figure 7 compares the Cumulative Distribution Function (CDF) of the delays of all demand pairs in three cases: no blocking, proportional blocking at ratio 4% and optimal blocking at average ratio 4%. From the curve of no blocking we observe that under network failure, the performance of part of the network degrades severely. There is a large delay variance. While the proportional blocking decreases the average delay, we can see some demand pairs still have large delays. It indicates that the network congestion can not be alleviated effectively by proportional blocking. The joint optimization method not only decreases the average delay of demand pairs, but also effectively reduces the delay variance. This verifies that our joint blocking and routing method can maintain a good network-wide delay performance with minimal ingress traffic blocking.

## VI. CONCLUSION AND FUTURE WORK

In this paper, we study the joint ingress blocking and routing problem under network failures and maintenances. We have shown that ingress traffic blocking is very effective in alleviating network congestion caused by failures and maintenances. The proposed joint blocking and routing approach can efficiently find the optimal differentiation

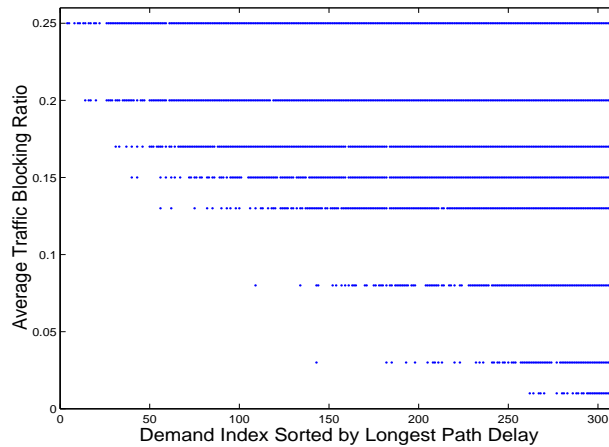


Fig. 5. Evolution of Demand Pair Blocking

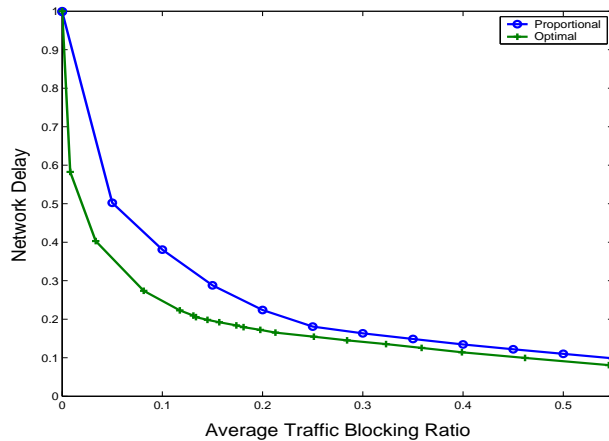


Fig. 6. Comparison with Proportional Blocking

among users with different service priorities and ingress-egress combinations. Our approach can be used to achieve a desired level of balance between network delay and user utility. Our study opens up a new design space for network traffic engineering.

Future research can be pursued in the following directions.

- *Link-State Routing.* Our evaluation in this paper is based on MPLS routing. We have shown in Section III-D that MPLS baseline model can be used to solve the blocking problem for a link-state routing. We will evaluate the performance of our proposed approach using popular link-state routing protocols, such as OSPF and IS-IS.
- *Router-Level Solution.* Limited by the speed of the linear programming solver, we were only able to obtain the optimization solution for the PoP-level topology. How to map a PoP-level solution to a router-level configuration deserves further study. Another direction to obtain a router-level solution is to reduce the size of the router-level joint optimization problem by sacrificing the optimality of the solution. We will study the tradeoffs in developing such simplifications.
- *Weighted Delay Performance.* We formulated in Section III-B the joint optimization problem when users have different delay performance weights. It is difficult to solve exactly the resultant fractional programming problem. We will approximate it by a linear programming problem to obtain approximation solution.
- *Optimization for Packet Losses.* In this paper, we focused on the network delay performance. Another important metric under network congestion is link packet loss probabilities. We plan to modify the current framework to account for packet losses. The decision to be made by a network provider is to either drop packets reactively on routers inside his network or drop packets proactively on ingress routers. One challenging issue is that packet losses will make traffic flows *non-conservative* inside the network.

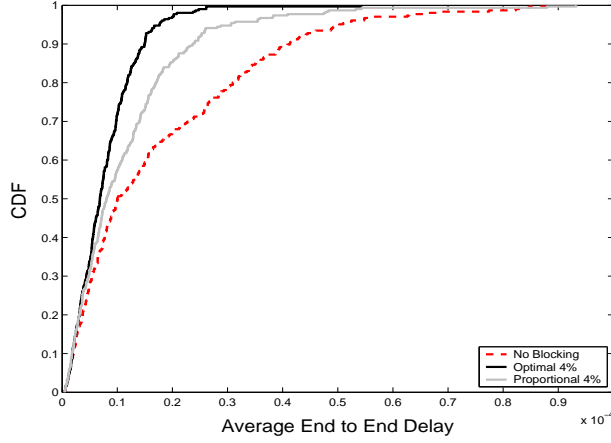


Fig. 7. CDF of Demand Average End to End Delay

#### APPENDIX A

**Claim:** If there is no delay differentiation among users, joint blocking and source-destination based routing has the same optimum as the baseline model in (5).

*Proof:* Assume  $\mathcal{B}^* = \{d_i^*, i \in D\}$  and  $\mathcal{R}^* = \{\tau_i^l, i \in D, l \in E\}$  is the optimal solution of (5). The traffic rate on link  $l$  is  $f_l^*$ . For any ingress-egress point pair  $(s, d) \in (V \times V)$ , let  $D(s, d) = \{i \in D | I(i) = s \text{ and } E(i) = d\}$  be the set of users with ingress point  $s$  and egress point  $d$ . Calculate the aggregate traffic rate of users in group  $D(s, d)$  on link  $l$ :

$$y_{(s,d)}^l = \sum_{i \in D(s,d)} \tau_i^l.$$

Since  $\{\tau_i^l, l \in E\}$  satisfy flow conservation (2) and implement demand  $d_i^*$ . It is easy to verify that  $\{y_{(s,d)}^l, l \in E\}$  satisfy flow conservation and implement the aggregate demand between  $s$  and  $d$ ,  $d_{(s,d)}^* \triangleq \sum_{i \in D(s,d)} d_i^*$ ,

$$\sum_{l \in \mathcal{O}(v)} y_{(s,d)}^l - \sum_{l \in \mathcal{P}(v)} y_{(s,d)}^l = \begin{cases} d_{(s,d)}^* & \text{if } v = s \\ -d_{(s,d)}^* & \text{if } v = d \\ 0 & \text{otherwise} \end{cases}. \quad (15)$$

Therefore  $\mathcal{Y} = \{y_{(s,d)}^l, (s, d) \in (V \times V), l \in E\}$  is a legitimate source-destination routing for the admitted demand  $\mathcal{B}^* = \{d_i^*, i \in D\}$ . The traffic rate on any link  $l$  under  $\mathcal{Y}$  equals  $f_l^*$ . Therefore the aggregate network delay under  $\mathcal{Y}$  equals that of  $\mathcal{R}^*$ . The joint blocking and routing solution  $\mathcal{B}^*$  and  $\mathcal{Y}$  achieve the optimum of the baseline MPLS model in (5). Since any source-destination based routing  $\mathcal{Y}$  can be represented by a flow routing  $\mathcal{R}$ , the optimal joint blocking and source-destination routing will have exactly the same performance as the optimum of (5). ■

#### APPENDIX B

**Claim:** The group utility function defined in equation (11) is an increasing and concave function of the aggregate group traffic rate  $z$ .

*Proof:* Let  $\{d_i^{(j)}, i \in D(s, d)\}$  denote the optimal rate allocation among users in group  $D(s, d)$  if the aggregate group rate is  $z_j$ :

$$\{d_i^{(j)}, i \in D(s, d)\} \triangleq \underset{\sum_{i \in D(s,d)} d_i = z_j}{\operatorname{argmax}} \sum_{i \in D(s,d)} U_i(d_i), \quad (16)$$

Then

$$U_{(s,d)}(z_j) = \sum_{i \in D(s,d)} U_i(d_i^{(j)}).$$

If  $\{d_i^{(1)}, i \in D(s, d)\}$  is the rate allocation for  $z_1, \forall z_2 > z_1$ , construct another rate allocation as

$$d_i^+ = d_i^{(1)} + (z_2 - z_1) \frac{d_i^0 - d_i^{(1)}}{\sum_{j \in D(s, d)} (d_j^0 - d_j^{(1)})}.$$

Obviously,  $\{d_i^+, i \in D(s, d)\} > \{d_i^{(1)}, i \in D(s, d)\}$  (element-wise vector comparison). Since  $U_i(\cdot)$  is increasing function, we have  $\sum_{i \in D(s, d)} U_i(d_i^+) > U_{(s, d)}(z_1)$ . Now that  $\sum_{i \in D(s, d)} d_i^{(+)} = z_2$ , through the definition of  $U_{(s, d)}(\cdot)$ ,

$$U_{(s, d)}(z_2) \geq \sum_{i \in D(s, d)} U_i(d_i^+) > U_{(s, d)}(z_1).$$

In other words, the group utility function increases with the aggregate group rate.

Given the optimal group rate allocations  $\{d_i^{(1)}\}$  and  $\{d_i^{(2)}\}$  for aggregate group rate  $z_1$  and  $z_2$ , construct a rate allocation  $\{d_i^{(\alpha)}\}$  for the aggregate group rate  $\alpha z_1 + (1 - \alpha)z_2$  as follows:

$$d_i^{(\alpha)} = \alpha d_i^{(1)} + (1 - \alpha) d_i^{(2)}.$$

Since  $U_i(\cdot)$  is concave function,

$$\begin{aligned} \sum_{i \in D(s, d)} U_i(d_i^{(\alpha)}) &\geq \sum_{i \in D(s, d)} (\alpha U_i(d_i^{(1)}) + (1 - \alpha) U_i(d_i^{(2)})) \\ &= \alpha U_{(s, d)}(z_1) + (1 - \alpha) U_{(s, d)}(z_2). \end{aligned}$$

Consequently,

$$\begin{aligned} U_{(s, d)}(\alpha z_1 + (1 - \alpha)z_2) &\geq \sum_{i \in D(s, d)} U_i(d_i^{(\alpha)}) \\ &\geq \alpha U_{(s, d)}(z_1) + (1 - \alpha) U_{(s, d)}(z_2). \end{aligned}$$

Therefore, the group utility function  $U_{(s, d)}(z)$  is an increasing and concave function of the aggregate group rate  $z$ . ■

## REFERENCES

- [1] J. Moy/IETF, "OSPF Version 2(RFC 2328)," 1998.
- [2] R. Callon/IETF, "Use of OSI IS-IS for routing in TCP/IP and dual environments(RFC 1195)," 1990.
- [3] D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell, and J. McManus/IETF, "Requirements for Traffic Engineering Over MPLS (RFC 2328)," 1999.
- [4] D. G. Cantor and M. Gerla, "Optimal Routing in a Packet-Switched Computer Network," *IEEE Transactions on Computer*, vol. C-23, no. 10, pp. 1062–1069, 1974.
- [5] R. G. Gallager, "A Minimum Delay Routing Algorithm Using Distributed Computation," in *IEEE Transaction on Communications*, january 1977, pp. 73–85.
- [6] B. Fortz and M. Thorup, "Internet Traffic Engineering by Optimizing OSPF Weights," in *Proceedings of IEEE INFOCOM*, 2000, pp. 519–528.
- [7] T. Ye and S. Kalyanaraman, "A recursive random search algorithm for large-scale network parameter configuration," in *Proceedings of ACM SIGMETRICS*, 2003.
- [8] W. Leland, M. Taqqu, W. Willinger, and D. V. Wilson, "On the self-similar nature of Ethernet traffic," *IEEE/ACM Transactions on Networking*, vol. 2, no. 1, February 1994.
- [9] V. Paxson and S. Floyd, "Wide Area traffic: The failure of poisson modelling," in *Proceedings of ACM SIGCOMM*, 1994.
- [10] B. Fortz and M. Thorup, "Optimizing OSPF/IS-IS Weights in a Changing World," in *IEEE Journal on Selected Areas in Communications*, 2002, pp. 756–767.
- [11] A. Medina, N. Taft, K. Salamatian, S. Bhattacharyya, and C. Diot, "Traffic matrix estimation: Existing techniques and new directions," in *Proceedings of ACM SIGCOMM*, 2002.
- [12] M. Roughan, A. Greenberg, C. Kalmanek, M. Rumsewicz, J. Yates, and Y. Zhang, "Experience in measuring backbone traffic variability: Models, metrics, measurements and meaning," in *Proceedings of Internet Measurement Workshop*, 2002.
- [13] Y. Zhang, M. Roughan, N. Duffield, and A. Greenberg, "Fast Accurate computation of large-scale IP traffic matrices from link loads," in *Proceedings of ACM SIGMETRICS*, 2003.
- [14] C. Zhang, Z. Ge, J. Kurose, Y. Liu, and D. Towsley, "Optimal Routing with Multiple Traffic Matrices: Tradeoff between Average Case and Worst Case Performance," in *Proceedings of IEEE International Conference on Network Protocols (ICNP)*, 2005.
- [15] C. Zhang, Y. Liu, W. Gong, J. Kurose, R. Moll, and D. Towsley, "On Routing Optimization with Multiple Traffic Matrices," in *Proceedings of IEEE INFOCOM*, 2005.

- [16] D. Applegate and E. Cohen, "Making Intra-Domain Routing Robust to Changing and Uncertain Traffic Demands: Understanding Fundamental Tradeoffs," in *Proceedings of ACM SIGCOMM*, 2003.
- [17] M. Roughan, M. Thorup, and Y. Zhang, "Traffic Engineering with Estimated Traffic Matrices," in *Proceedings of Internet Measurement Conference*, 2005.
- [18] G. Iannaccone, C.-N. Chuah, R. Mortier, S. Bhattacharyya, and C. Diot, "Analysis of link failures in an IP backbone," in *Proceedings of Internet Measurement Workshop*, Nov. 2002.
- [19] K. Murakami and H. S. Kim, "Optimal capacity and flow assignment for self-healing ATM networks based on line and end-to-end restoration," *IEEE/ACM Transactions on Networking*, vol. 6, no. 2, pp. 207–221, 1998.
- [20] J. Anderson, B. T. Doshi, S. Dravida, and P. Harshavardhana, "Fast Restoration of ATM Networks," *IEEE Journal on Selected Areas in Communications*, vol. 12, no. 1, pp. 128–138, 1994.
- [21] A. Bremner-Barr, Y. Afek, H. Kaplan, E. Cohen, and M. Merritt, "Restoration by Path Concatenation: Fast Recovery of MPLS Paths," *Distributed Computing*, vol. 15, no. 4, 2002.
- [22] D. Applegate, L. Breslau, and E. Cohen, "Coping with network failures: routing strategies for optimal demand oblivious restoration," in *Proceedings of ACM SIGMETRICS*, 2004, pp. 270–281.
- [23] A. Nucci, B. Schroeder, S. Bhattacharyya, N. Taft, and C. Diot, "IGP Link Weight Assignment for Transient Link Failures," in *IEEE ITC18*, 2003. [Online]. Available: [citeseer.nj.nec.com/572602.html](http://citeseer.nj.nec.com/572602.html)
- [24] A. Sridharan and R. Guerin, "Making OSPF/IS-IS Routing Robust to Link Failures," University of Pennsylvania, Tech. Rep., July 2004.
- [25] F. Kelly, A. Maulloo, and D. Tan, "Rate control in communication networks: shadow prices proportional fairness and stability," *Journal of the Operational Research Society*, 1998.
- [26] S. Kunniyur and R. Srikant, "End-to-end congestion control schemes: Utility functions, random losses and ECN marks," in *Proceedings of IEEE INFOCOM*, 2000, pp. 1323–1332.
- [27] S. H. Low, "A duality model of TCP and queue management algorithms," *IEEE/ACM Transactions on Networking*, vol. 11, no. 4, August 2003.
- [28] M. Vojnović, J. Boudec, and C. Boutremans, "Global fairness of Additive Increase and Multiplicative Decrease with heterogeneous roundtrip times," in *Proceedings of IEEE INFOCOM*, 2000.
- [29] J. He, M. Chiang, and J. Rexford, "TCP/IP interaction based on congestion prices: Stability and optimality," in *Proceedings of ICC*, June 2006.
- [30] J. Wang, L. Li, S. H. Low, and J. C. Doyle, "Cross-layer optimization in TCP/IP networks," *IEEE/ACM Transactions on Networking*, vol. 13, no. 3, pp. 582–595, 2005.
- [31] X. Lin and N. B. Shroff, "Joint Rate Control and Scheduling in Multihop Wireless Networks," in *43rd IEEE Conference on Decision and Control*, 2004.
- [32] L. Xiao, M. Johansson, and S. Boyd, "Simultaneous routing and resource allocation via dual decomposition," *IEEE Transactions on Communications*, vol. 52, no. 7, pp. 1136–1144, 2004.
- [33] X. Lin and N. B. Shroff, "The Impact of Imperfect Scheduling on Cross-Layer Rate Control in Multihop Wireless Networks," in *Proceedings of IEEE INFOCOM*, 2005.
- [34] L. Chen, S. H. Low, M. Chiang, and J. C. Doyle, "Optimal Cross-layer Congestion Control, Routing and Scheduling Design in Ad Hoc Wireless Networks," in *Proceedings of IEEE INFOCOM*, 2006.
- [35] C. Zhang, J. Kurose, Y. Liu, D. Towsley, and M. Zink, "A Distributed Algorithm for Joint Sensing and Routing in Wireless Networks with Non-Steerable Directional Antennas," in *Proceedings of IEEE International Conference on Network Protocols (ICNP)*, 2006.
- [36] AMPL/CPLEX, "<http://www.ilog.com>."
- [37] A. Feldmann, A. Greenberg, C. Lund, N. Reingold, J. Rexford, and F. True, "Deriving Traffic Demands for Operational IP Networks: Methodology and Experience," *IEEE/ACM Transactions on Networking*, vol. 9, no. 3, pp. 265–279, 2001.
- [38] J. Yates, P. Sebos, A. Greenberg, and G. Hjalmtysson, "Auto-discovery of shared risk link groups," in *Proceeding of IEEE Conference on Optical Fiber Communication*, 2001.