

Provided for non-commercial research and education use.  
Not for reproduction, distribution or commercial use.



This article was published in an Elsevier journal. The attached copy is furnished to the author for non-commercial research and education use, including for instruction at the author's institution, sharing with colleagues and providing to institution administration.

Other uses, including reproduction and distribution, or selling or licensing copies, or posting to personal, institutional or third party websites are prohibited.

In most cases authors are permitted to post their version of the article (e.g. in Word or Tex form) to their personal website or institutional repository. Authors requiring further information regarding Elsevier's archiving and manuscript policies are encouraged to visit:

<http://www.elsevier.com/copyright>



# Multihoming route control among a group of multihomed stub networks

Yong Liu, A.L. Narasimha Reddy \*

*Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX 77843, USA*

Available online 20 December 2006

---

## Abstract

In recent years, commercial “multihoming route control” devices are used by multihomed stub networks to optimize the routing of their Internet traffic. Previous studies have shown that “multihoming route control” can improve the Internet communication performance of the multihomed stub networks significantly. In this work, we study “multihoming route control” among a group of multihomed stub networks that may belong to an organization and exchange large amount of data regularly. We show that greedy “multihoming routing control” schemes could lead to oscillations. We propose a user-optimal routing based “multihoming routing control” scheme that improves routing performance without causing oscillations. The proposed scheme is simpler to implement than optimal routing based approaches. We show through simulations that the proposed scheme achieves performance close to optimal routing and works well in various network conditions.

© 2006 Elsevier B.V. All rights reserved.

*Keywords:* Multihoming route control; User-optimal routing; Selfish routing; Optimal routing

---

## 1. Introduction

Multihoming [1] has been traditionally used by stub networks for improving service availability. In recent years, Multihoming Route Control (MRC) technology [2] has been employed by multihomed stub networks to improve their Internet access performance. MRC devices choose the best ISP for Internet traffic of stub networks according to measured qualities of alternate paths via different ISPs. Measurement based analysis of the benefit of multihoming [3] shows that MRC may improve Internet access performance significantly for both enterprises and large data centers.

Multihoming route control is usually done in a distributed manner: each stub network adaptively changes the ISP of its traffic to a destination network according to its own view of the quality of alternate paths via different

ISPs. When (1) the traffic controlled by multiple MRC devices shares bottleneck links and (2) the controlled traffic accounts for a significant fraction of the total load on the bottleneck links, the MRC done by different stub networks may interact with each other. In such situations, greedy MRC approaches may cause oscillations, as we will illustrate in Section 5.1.

In this paper, we study MRC among a group of multihomed stub networks, for example, the networks of branches of an enterprise that are multihomed and exchange considerable traffic regularly among themselves. Since access links of stub networks are more likely to be the bottlenecks along end to end Internet paths, MRC among a group of multihomed stub networks is more likely to cause oscillations. A global optimal routing based coordination method can be employed to avoid such possible oscillations [4]. In this work, we propose a distributed “user-optimal routing” [5] based MRC scheme to solve the above problem. The basic idea is to use multiple paths provided by multihoming simultaneously and move traffic gradually during changes in network environment. Specifically, our

---

\* Corresponding author. Tel.: +1 9798457598.

*E-mail addresses:* [yongliu@ece.tamu.edu](mailto:yongliu@ece.tamu.edu) (Y. Liu), [reddy@ece.tamu.edu](mailto:reddy@ece.tamu.edu) (A.L. Narasimha Reddy).

scheme calculates “user-optimal routing” using the gradient projection method that is originally used in solving optimal routing problems. User-optimal routing is simpler to implement. Moreover, as we will show in Section 5, it can achieve similar performance as global optimal routing for this problem.

While our approach is mainly designed for MRC among a group of multihomed stub networks, it can also be applied to Internet traffic to a number of “top” Internet destination networks that account for a large portion of the Internet traffic of the stub networks. In this study, we assume there are a few top Internet destinations for each stub network and we use the same MRC algorithm for these Internet destinations. We call the traffic among such a group of stub networks as *inhouse* traffic and call the traffic between a stub network and networks not in the group as *Internet* traffic.

The rest of this paper is organized as follows: Related work is discussed in Section 2. In Section 3, we give the network model of the MRC problem. In Section 4, we introduce the user-optimal routing based MRC scheme. In Section 5, we first give an example to illustrate possible oscillations caused by greedy MRC, then we compare the performance of our scheme with the optimal solution and show the dynamic characterization of our scheme using simulations. Future work are discussed in Section 6. Conclusions are drawn in Section 7.

## 2. Related work

The benefits of MRC are studied using both Internet measurements [3] and emulation [6]. Tao et al. [7] have measured quality of alternate paths among between three campus networks. They show packet losses are bursty and no path is consistently better than others. Goldenberg et al. [8] have studied the optimization of cost and performance for multihoming. All the above work does not consider interaction between MRC of different stub networks.

The performance of user-optimal routing, also called selfish routing, has been studied analytically [9] and using simulations [10]. Qiu et al. [10] show selfish routing achieves performance close to optimal routing for intra-domain routing. Similarly, our work shows selfish routing based MRC achieves performance close to optimal routing based MRC.

Centralized and distributed gradient projection algorithms for optimal routing have been proposed in [11–13]. Measurement based multi-path optimal routing has been proposed in [14,15] where the cost information is propagated using an overlay network. Our MRC approach uses measurement based gradient projection algorithm to calculate the user-optimal solution. It is similar to [14,15] but does not require overlay infrastructure to exchange information, thus is simpler to implement.

In summary, our work leverages previous work on optimal routing and user-optimal routing and targets a new problem on multihoming route control.

## 3. Network model

There are two types of multihoming: NAT (Network Address Translation) [16] based and BGP [17] based. NAT based multihoming is usually used by small to middle size stub networks because it does not require the stub network to have an independent IP block and maintain a BGP router. BGP based multihoming is usually used by a large stub network that has independent IP address block(s) and maintains a BGP router. Accordingly, multihoming route control devices can be classified into NAT based and BGP based categories, see [18] for a survey.

In this paper, we study MRC for large stub networks that employ BGP based multihoming. We also assume the stub network advertises its IP address block(s) to all its ISPs<sup>1</sup>. In this case, the stub network can send outgoing traffic via either of its ISPs, but it cannot control which ISP the ingress traffic comes from. This is not a problem for MRC among a group of multihomed stub networks since all traffic is controlled by the MRC devices of the originating stub networks. For networks where MRC of ingress traffic is desirable, NAT based MRC should be considered which is beyond this work. The task of MRC for BGP based multihoming is to map egress traffic onto available paths provided through BGP based multihoming.

Existing MRC schemes work as follows [2]: the MRC device use some means to direct the BGP router(s) of local network to select a particular ISP for egress traffic to an IP address prefix. Because BGP uses single route for an address prefix at any time, MRC is also restricted to use a single route. It is essential for ISP networks to use single path for routing of inter-domain traffic in order to make BGP scalable. However, for stub networks, it is not a problem to use multiple paths for egress traffic. Therefore, in this work, we assume stub networks can use multiple paths for egress traffic. To deploy our approach, multiple routes need to be assigned for interested destination networks and the forwarding engine uses our algorithm to decide the fraction of traffic to send on each route. Hashing based traffic splitting methods, like [19], or reordering robust TCP, such as [20,21], can be used to avoid the packet reordering caused by multi-path routing.

We assume that the traffic controlled by the stub networks accounts for only a small part of the total traffic on any link of the backbone networks which is usually true. Under this assumption, the MRC of traffic of the stub networks would not affect the quality (or level of load) of backbone links and we abstract a network path between the ISP edge routers of two stub networks as a directed “virtual” link with a given quality that may change over-time. In our simulations, we also abstract paths from ISP edge routers of a stub network to an Internet destination

<sup>1</sup> While it is possible to control the incoming traffic direction through selective advertisement of addresses to different ISPs, such a control is only possible over longer timescales and may leave the stub network vulnerable to network failures.

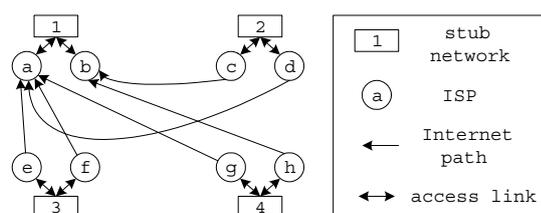


Fig. 1. Topology consisting of multihomed stub networks, the edge routers of their ISPs and the paths among them.

as “virtual” links, thus ignore the access link bottleneck of remote Internet destination networks. The above abstraction of “virtual” links is only for evaluation simulations. Our algorithm works without such requirement. Since we are studying MRC among a group of stub networks, we also abstract each stub network as a node. Therefore, a network we are studying consists of: (1) nodes representing stub networks in the group; (2) nodes representing ISP edge routers of the stub networks; (3) nodes representing a number of top Internet destinations of the stub networks; (4) links representing access links of the stub networks; (5) virtual directed links between ISP edge routers of different stub networks; (6) virtual directed links between ISP edge routers of the stub networks and the Internet destinations.

Depending on whether the group of networks multihome to the same set of ISPs, there are two types of topologies for traffic among the stub networks: (1) Symmetric topology: When all the nodes multihome to the same set of ISPs, the path from a stub network  $i$  via ISP  $k$  to any other stub network, say  $j$ , normally reach  $j$  via ISP  $k$ . Thus, the alternate paths between these two stub networks are “parallel”, they merge only inside the stub networks we considered. (2) Asymmetric topology: When nodes multihome to different set of ISPs, the alternate paths from a stub network to another stub network are not necessarily “parallel”. Two paths to a stub network may merge in one ISP of the stub network or in other AS between the two stub networks. This is decided by the BGP relationship of ASes between the two stub networks.

Fig. 1 shows the partial topology of a “ $4 \times 2$ ” network (“ $A \times B$ ” means the topology has  $A$  stub networks and each stub network has  $B$  ISPs). The ISPs of different stub networks are different. For clarity, we only draw paths from stub networks 2, 3, and 4 to stub network 1. Other paths among the stub networks and paths to and from Internet destinations are ignored.

#### 4. User-optimal routing based MRC among a group of multihomed stub networks

“User-optimal routing” [5] is optimal from the point of view of each user. It is also called “selfish routing”. Like previous work on “user-optimal routing” [10], we assume traffic consists of a lot of “infinitesimal flows” and each user controls such an infinitesimal flow. At equilibrium of user-optimal routing, each flow is routed along a path with

minimum end to end delay. Thus no user can reduce the delay of its traffic by changing the routing of its own traffic unilaterally. Previous work [10] showed that “selfish routing” can achieve similar performance as “optimal routing” [11] for intra-domain routing. As we will show in Section 5, for MRC, the performance of user-optimal routing based approach is also close to optimal routing based approach. One of the advantages of user-optimal routing is that it is distributed in nature and easier to implement.

In this section, we first introduce the concept of optimal routing formulation of our MRC problem. Then, we give the user-optimal routing formulation of our MRC problem. In the end, we give our algorithm for the MRC problem. We give the optimal routing formulation here because it is the base of our main algorithm and we will compare the performance of our algorithm to the optimal solution.

#### 4.1. Optimal routing

##### 4.1.1. Formulation

The problem we studied can be formulated as a form of optimal routing [11] problem. In a general optimal routing problem, routing traffic on a link incurs some cost that is a function of the total load on the link and the optimal solution maps all traffic on all “physically possible” paths such that the overall cost is minimized. In our problem, only a limited number of paths are given and we need only to map traffic to these paths.

The cost function used in optimal routing is usually a continuous non-decreasing convex function. A common cost function used in earlier optimal routing work is the delay on the link weighted by the traffic volume on the link. The objective is to find a routing solution that minimizes the average delays experienced by all traffic. Using this cost function, other performance metrics, packet loss rate and congestion are partly considered because these metrics are correlated with queuing delay of a link. In practice, the loss can also be considered by replacing the delay in our formulation with a virtual delay function. In this work, we use half of the expected TCP hand shake time [22] as the virtual one way delay function, i.e.  $\text{delay} + T_s \frac{\text{loss\_rate}}{1-2\text{loss\_rate}}$  ( $\text{loss\_rate} < 0.5$ ), where  $T_s$  is the TCP SYN timeout, initially three seconds [23]. Here, we assume the reverse path has same delay and loss rate as the forward path. TCP handshake round trip time is used in comparing qualities of alternate paths by previous studies on multihoming, e.g. [3]. When loss rates are small, e.g. less than 1%, the sum virtual delays of links along a path is roughly same as the virtual delay of the path.

Before giving the optimal routing formulation of our problem, we define following symbols.

- $N$ : set of nodes representing stub networks;
- $M_i$ : set of nodes representing Internet destinations of  $i \in N$ ;
- $K_i$ : set of nodes representing ISP edge routers of  $i \in N$ ;

- $P_{ij}$ : set of virtual directed links representing valid paths between ISP edge routers of  $i$  and ISP edge routers of  $j$ , where  $i, j \in N$ ,  $i \neq j$ , or paths from ISP edge routers of  $i$  to Internet destination  $j$ , where  $i \in N$ ,  $j \in M_i$ ;
- $(i, j)$ : link from  $i$  to  $j$ , where  $i \in N$ ,  $j \in K_i$  or  $i \in K_j$ ,  $j \in N$ , or virtual link from  $i$  to  $j$ , where  $i \in K_m$ ,  $j \in M_n$ ,  $n \in N$  or  $i \in K_m$ ,  $j \in K_n$ ,  $(i, j) \in P_{mn}$ ,  $m, n \in N$ ,  $m \neq n$ ;
- $d_{ij}(x)$ : virtual delay function of directed link  $(i, j)$ , where  $x$  is the load on link  $(i, j)$ ,  $i \in N$ ,  $j \in K_i$  or  $i \in K_j$ ,  $j \in N$ ;
- $d_{ijw}$ : virtual delay of virtual link  $w$ , where  $w \in P_{ij}$ ,  $i, j \in N$ ,  $i \neq j$ , or  $i \in N$ ,  $j \in M_i$ ;
- $r_{ij}$ : traffic demand from  $i$  to  $j$ , where  $i, j \in N$ ,  $i \neq j$  or  $i \in N$ ,  $j \in M_i$ ;
- $x_{ijw}$ : fraction of  $r_{ij}$  routed along path  $w \in P_{ij}$ , where  $i, j \in N$ ,  $i \neq j$  or  $i \in N$ ,  $j \in M_i$ ;
- $u_{ij}$ : load of ingress Internet traffic on link  $(i, j)$ ,  $i \in K_j$ ,  $j \in N$ ;
- $S_{ik}$ : set of paths (virtual links), egress traffic on which passes link  $(i, k)$ , where  $i \in N$ ,  $k \in K_i$ , i.e.  $\{(m, n) | m = k, n \in K_j, j \in N, j \neq i, (m, n) \in P_{ij}\} \cup \{(m, n) | m = k, n \in M_i\}$ ;
- $S_{ki}$ : set of paths (virtual links), ingress traffic on which passes link  $(k, i)$ , where  $i \in N$ ,  $k \in K_i$ , i.e.  $\{(m, n) | m \in K_j, n = k, j \in N, j \neq i, (m, n) \in P_{ij}\}$ .

The optimal routing formulation of our problem is as follows.

Minimize:

$$C(x) = \sum_{i \in N, j \in (N \setminus i) \cup M_i, w \in P_{ij}} x_{ijw} r_{ij} d_{ijw} + \sum_{i \in N, k \in K_i} t_{ki} d_{ki}(t_{ki}) + \sum_{i \in N, k \in K_i} t_{ik} d_{ik}(t_{ik}) \quad (1)$$

Subject to:

$$x_{ijw} \geq 0, \quad (i \in N, j \in (N \setminus i) \cup M_i, w \in P_{ij}) \quad (2)$$

$$\sum_{w \in P_{ij}} x_{ijw} = 1, \quad (i \in N, j \in (N \setminus i) \cup M_i) \quad (3)$$

$$t_{ik} = \sum_{w \in S_{ik}} x_{ijw} r_{ij}, \quad (i \in N, k \in K_i) \quad (4)$$

$$t_{ki} = \sum_{w \in S_{ki}} x_{ijw} r_{ij} + u_{ki}, \quad (i \in N, k \in K_i) \quad (5)$$

where (1) is the objective function, i.e. virtual delays of Internet paths (virtual links) experienced by traffic controlled by MRC devices plus virtual delays on access links weighted by traffic volumes; (2) is non-negativity of routing vectors; (3) is to ensure that all traffic are routed; (4) and (5) are traffic volume on access links.

A measurement based multipath optimal routing algorithm [14] can be used to route traffic in response to network changes. However, the algorithm needs a overlay network infrastructure to measure the first derivatives of the cost functions of all links in the network. Thus it needs to measure the first derivatives using traffic disturbance method. In our approach no such information exchange is required.

#### 4.1.2. Characterization of optimal routing

We give the characterization of optimal routing here, because it is useful to design an algorithm for user-optimal routing as we will see in Section 4.2.2. According to [11], the characterization of the optimal solution  $x^*$  is:

$$x_{ijw}^* > 0 \Rightarrow \frac{\partial C(x^*)}{\partial x_{ijw'}} \geq \frac{\partial C(x^*)}{\partial x_{ijw}}, \quad (i \in N, j \in (N \setminus i) \cup M_i; w, w' \in P_{ij}) \quad (6)$$

In other words, at the optimal point, for any source–destination pair, shifting a small amount of traffic from one path to an alternate path that is not used by the source–destination pair will increase the total cost, and shifting a small amount of traffic to an alternate path that is used by the source–destination pair would not change the total cost. In summary, at the optimal point, changing routing solution  $x$  would not lower the total cost.

### 4.2. User-optimal routing

#### 4.2.1. Characterization of user-optimal routing

The characterization of user-optimal routing is that user-optimal routing allocation is positive only on paths with minimum end to end delay [24]. In our MRC problem, it is as follows:

$$x_{ijw}^* > 0 \Rightarrow \begin{cases} d_{ijw'} + d_{ik'}(t_{ik'}) + d_{l'j}(t_{l'j}) \geq d_{ijw} + d_{ik}(t_{ik}) + d_{lj}(t_{lj}), \\ (i \in N, j \in (N \setminus i), w, w' \in P_{ij}, w = (k, l), w' = (k', l')), \\ x_{ijw}^* > 0 \Rightarrow d_{ijw'} + d_{ik'}(t_{ik'}) \geq d_{ijw} + d_{ik}(t_{ik}), \\ (i \in N, j \in M_i, w, w' \in P_{ij}, w = (k, j), w' = (k', j)) \end{cases} \quad (7)$$

When the delay function of a link is continuous and nondecreasing, there exists a unique equilibrium solution [24].

#### 4.2.2. Formulation of user-optimal routing

Because the similarity of the characterization between network-optimal routing and user-optimal routing, user-optimal routing can be solved using algorithms for network-optimal routing with a specific cost function [25]. Specifically, we need to define the cost function of a link,  $l$ , as

$$D_l(x) = \int_0^x d_l(t) dt \quad (8)$$

Therefore, the user-optimal routing problem can be formulated as the following optimal routing problem and can be solved using the algorithms for optimal routing. Minimize:

$$D(x) = \sum_{i \in N, j \in (N \setminus i) \cup M_i, w \in P_{ij}} x_{ijw} r_{ij} d_{ijw} + \sum_{i \in N, k \in K_i} D_{ki}(t_{ki}) + \sum_{i \in N, k \in K_i} D_{ik}(t_{ik}) \quad (9)$$

subject to: (2)–(5)

The solution for this optimal routing problem has the characterization given by (6). Thus

$$x_{ijw}^* > 0 \Rightarrow \frac{\partial D(x^*)}{\partial x_{ijw'}} \geq \frac{\partial D(x^*)}{\partial x_{ijw}} \quad (10)$$

$$\Leftrightarrow \begin{cases} d_{ijw'} + d_{ik'}(t_{ik'}) + d_{l'j}(t_{l'j}) \geq d_{ijw} + d_{ik}(t_{ik}) + d_{lj}(t_{lj}), \\ (i \in N, j \in (N \setminus i), w, w' \in P_{ij}, w = (k, l), w' = (k', l')); \\ d_{ijw'} + d_{ik'}(t_{ik'}) \geq d_{ijw} + d_{ik}(t_{ik}), \\ (i \in N, j \in M_i, w, w' \in P_{ij}, w = (k, j), w' = (k', j)) \end{cases} \quad (11)$$

That is equivalent to (7). I.e., the solution has the same characterization as the user-optimal routing solution. And because of the user-optimal solution is unique, as shown in Section 4.2.1, we actually get the user-optimal solution.

### 4.3. Distributed asynchronous algorithm

An important class of algorithms for solving optimal routing problems are the gradient projection methods [11]. They are also suitable for distributed implementation. In this work, we implement a distributed asynchronous gradient projection algorithm [11] to solve the user-optimal routing problem formulated in Section 4.2.2. Minor modifications are made to improve the convergence speed: normalization of delay difference and randomized waiting time. This algorithm is more practical than other projection methods because it requires simpler calculations at each iteration. Because important information needs to be exchanged during optimal routing calculation, the sum of first derivatives, are end to end delays of alternate paths that can be measured directly, the distributed algorithm can be implemented more easily than normal optimal routing algorithm. The algorithm for each node is shown as Algorithm 1.

**Algorithm 1** (User-optimal routing based MRC (for traffic from  $i$  to  $j$ , where  $i \in N$ ;  $j \in N$ ,  $j \neq i$  or  $j \in M_i$ )).

```

loop
  measure all one-way end to end virtual delays,
   $l_{ijw}; w \in P_{ij}$ 
  find the minimum virtual delay to  $j$ ,  $l_{ij\bar{w}}$ 
  for all  $w \in P_{ij}, w \neq \bar{w}$  do
     $x_{ijw}^{t+1} = \max\{0, x_{ijw}^t - \alpha^t(l_{ijw} - l_{ij\bar{w}})/l_{ijw}\}$ 
  end for
   $x_{ij\bar{w}}^{t+1} = 1 - \sum_{w \in P_{ij}, w \neq \bar{w}} x_{ijw}^t$ 
  wait for random time  $T \in [0.5T_0, 1.5T_0]$ ;
end loop

```

Our algorithm works as follows:

- (1) Each node measures the end to end delays of alternate paths from it to a remote node every  $t$  seconds, where  $T$  is uniformly distributed in  $[0.5T_0, 1.5T_0]$  to avoid update synchronization. Each measurement consists of a number of samples to filter noise. Because our algorithm is based on the difference of delays of alternate paths, it does not require clock synchronization of different nodes.

- (2) After the node gets measured end to end delays, it updates the routing vector  $x$  for this destination as shown in Algorithm 1. This is according to the gradient projection algorithm [11].  $\alpha^t$  is the step size, for distributed implementation it is usually a constant.

In this algorithm, we assume that the traffic demand and path quality does not change too rapidly compared to its convergence speed. This is true when the network traffic consists of a lot of small flows. The algorithm converges to user-optimal routing given  $\alpha$  is small enough [12].

## 5. Evaluation

In this section, we first show that greedy MRC may cause oscillations using simulations. Then we evaluate our user-optimal routing based MRC scheme. The evaluation consists of two parts: (1) Since our scheme is based on user-optimal routing, it is important to ensure it will not cause network wide performance degradation. We perform a number of simulations to compare the performance of user-optimal routing based MRC with the optimal solution. (2) We study the dynamic behavior of our algorithm in various dynamic network environments.

We wrote a flow level simulator for the simulations. The delay and loss rate on each access link are calculated according to a queuing model. To get the delay and loss rate for different link utilizations, we use piecewise linear approximation models built from samples of ns-2 [26] simulations. The two models are Poisson queuing model, M/M/1, and a Pareto ( $\beta = 1.5$ ) queuing model, P/M/1. The parameters of the ns-2 simulation are as follows: The average packet length is 558 bytes (calculated from a backbone trace); The link capacity is 100 Mbps; The buffer size of each link is equal to the product of 250 ms and link speed. The resulted virtual delay function of a link is a continuous, non-decreasing, convex function.

We generate traffic matrices using a simplified version of the Gravity model [27]. To generate inhouse traffic between nodes (stub networks), we assign two uniformly distributed random numbers to each node  $i$ ,  $O_i, D_i \in [0, 1]$ . Then the traffic demand from node  $i$  to node  $j$  is calculated as  $\alpha O_i D_j$ , where  $\alpha$  is a parameter,  $O_i$  and  $D_i$  model how active node  $i$  is as a sender and as a receiver. Similarly, to generate Internet traffic, we assign two uniformly distributed random numbers to each node  $i$ ,  $O'_i, D'_i \in [0, 1]$ . The egress and ingress Internet traffic of the node are  $\beta O'_i$  and  $\beta D'_i$ , where  $\beta$  is a parameter,  $O'_i$  and  $D'_i$  model how active node  $i$  is in sending and receiving Internet traffic. In our simulations, we choose the  $\alpha$  and  $\beta$  to make the expected volume of inhouse traffic as 50% of the total traffic. The egress Internet traffic is randomly distributed to five Internet destinations (each has a random weight uniformly distributed in  $[0, 1]$ ). The ingress Internet traffic is randomly distributed on all ingress access links (each has a random weight uniformly distributed in  $[0, 1]$ ).

We generate a network topology in two steps: (1) we map stub networks onto some of 17 major cities of the United States; (2) we generate the path delay (i.e. propagation delay) of alternate paths by multiplying the measured one way delay between the two cities on AT&T backbone [28] with a random number uniformly distributed in  $[1, 1.6]$ .

### 5.1. Possible oscillations of greedy MRC

In this section, we use simulations to show the possible oscillations of “greedy” MRC schemes for routing among a group of multihomed stub networks. Here, we scale the traffic matrix to make the maximum link utilization 95% assuming traffic are equally split among alternate paths.

The greedy MRC scheme we illustrated here works as follows: (1) the gateway of a stub network keeps measuring the one way delay and loss rate of alternate paths to other stub networks and a few top Internet destinations; (2) the gateway decides whether to change the routing path for traffic to a destination after a random interval that is uniformly distributed between 0.5 and 1.5 s; (3) the gateway changes the path for a destination when the “virtual delay” of currently used path is 20% larger than the minimum of alternate paths and is larger than 40 ms (we define “virtual delay” as a metric of overall path quality, see Section 4.1 for details).

We randomly assign a path for traffic to a destination at the beginning of simulation. The result for a “ $8 \times 3$ ” topology is shown in Fig. 2. We can see the average virtual delay of all the traffic in the network keeps fluctuating that reflects the oscillations of the “greedy” MRC scheme. Similar observations were made for other greedy approaches. These observations motivated our approach.

### 5.2. Performance compared to optimal routing

We calculate the global optimal routing (“gopt”), user-optimal routing (“uopt”) and static load-balancing (“elb”)

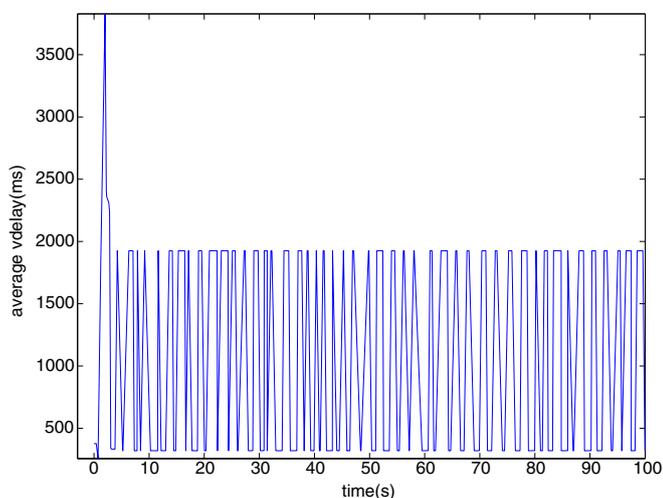


Fig. 2. Possible oscillations of greedy MRC.

MRC solutions for randomly generated topologies and traffic matrices. The “static load-balancing” here is to split traffic evenly among all alternate paths. The topologies are of size “ $4 \times 2$ ”, “ $4 \times 3$ ”, “ $8 \times 2$ ” and “ $8 \times 3$ ”. For each size, we generate one symmetric topology, six asymmetric topologies and five traffic matrices. We scale the traffic matrices to make the maximum link utilization 60%, 95%, 110% and 125% assuming basic load-balancing routing is used.

From the simulation results, we observe that the topology type (symmetric/asymmetric) (see Section 3), queuing model type (Pareto/Poisson) do not make significant difference to the simulation results. The explanations are as follows: (1) although packet arrival process under Pareto queuing model is more bursty than Poisson queuing model, the delay and loss rate still increase near and above 100% utilization. When link is over-utilized, the delay and loss rate for both type of packet arrival become very close. That is why we see the similar performance for both kind of queuing models in our simulations. (2) Because of the Internet traffic on links is not balanced, optimal routing and user-optimal routing have similar performance improvement for symmetric and asymmetric topologies. Results for different sizes of networks ( $4 \times 2$ ,  $4 \times 3$ ,  $8 \times 2$  and  $8 \times 3$ ) are also similar. Therefore, only the results for topologies of  $8 \times 2$ , asymmetric topologies and Pareto queuing model are presented here, as shown in Figs. 3–6.

For each simulation input (topology, queuing model, traffic matrix), we calculate the total routing cost, average delay, average loss rate and maximum link utilization for “gopt”, “uopt” and “elb”. The average, minimum and maximum values for simulations of each configuration (same topology type, same queuing model, same network size, traffic matrices that resulted same maximum link utilization for “elb”) are plotted in Figs. 3–6.

For the queuing models we studied, queuing delay is very small from utilization of 0 up to utilization of 90%. Therefore, from Fig. 4, we can see the average delays for all three routing approaches change very little when average utilization increases from 30.6% to 63.8%. However, because the difference of propagation delays of alternate paths, “uopt” and “gopt” can reduce the average delay by nearly 2 ms.

From Figs. 5 and 6, we can see the “uopt” and “gopt” based MRC reduces cost by reducing loss rate in the network. This is equivalent to reduce the maximum link utilizations in the network. Since for queuing models we are studying increasing link utilization up to 90% will not cause packet losses and obvious queuing delay increase, the “uopt” and “gopt” based MRC result in higher maximum link utilization (usually less than 100%) when the average utilization is low.

The conclusions we draw here are: (1) The performance of both approaches are very similar, which is consistent with previous work [10]. (2) Because the delay and loss rate increase significantly as the link utilization approaches

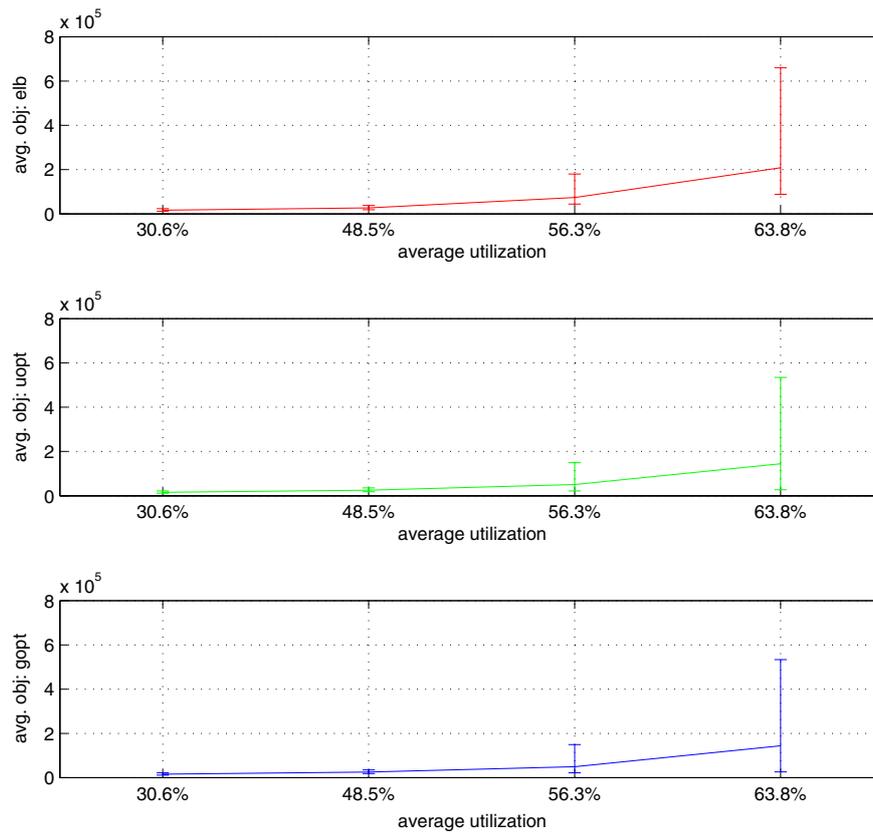


Fig. 3. Optimization objective of 30 simulations: average, minimum and maximum.

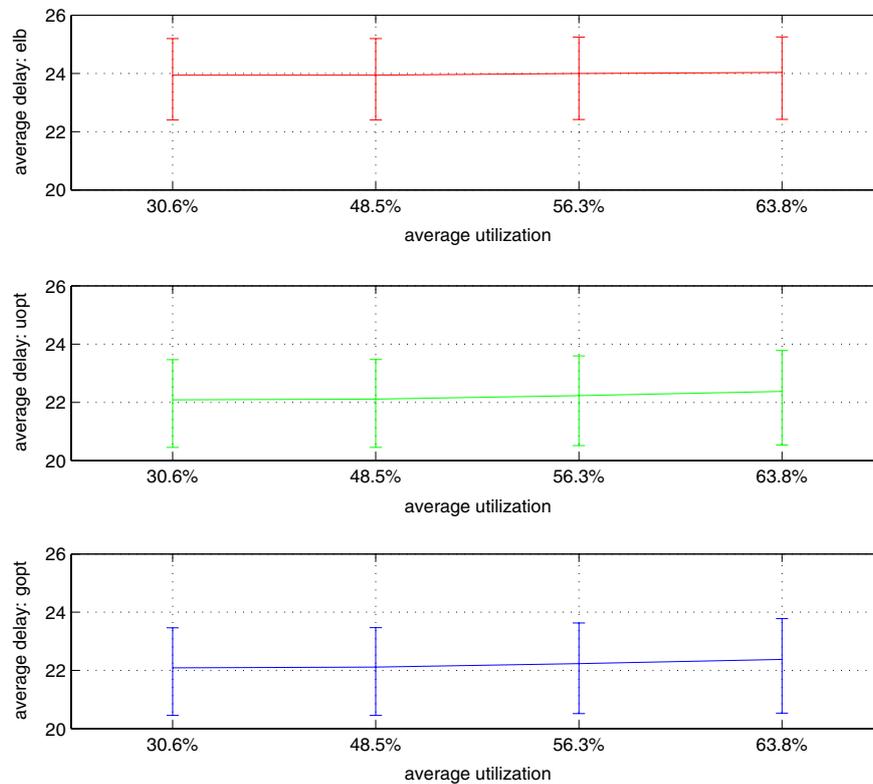


Fig. 4. Average delay of 30 simulations: average, minimum and maximum.

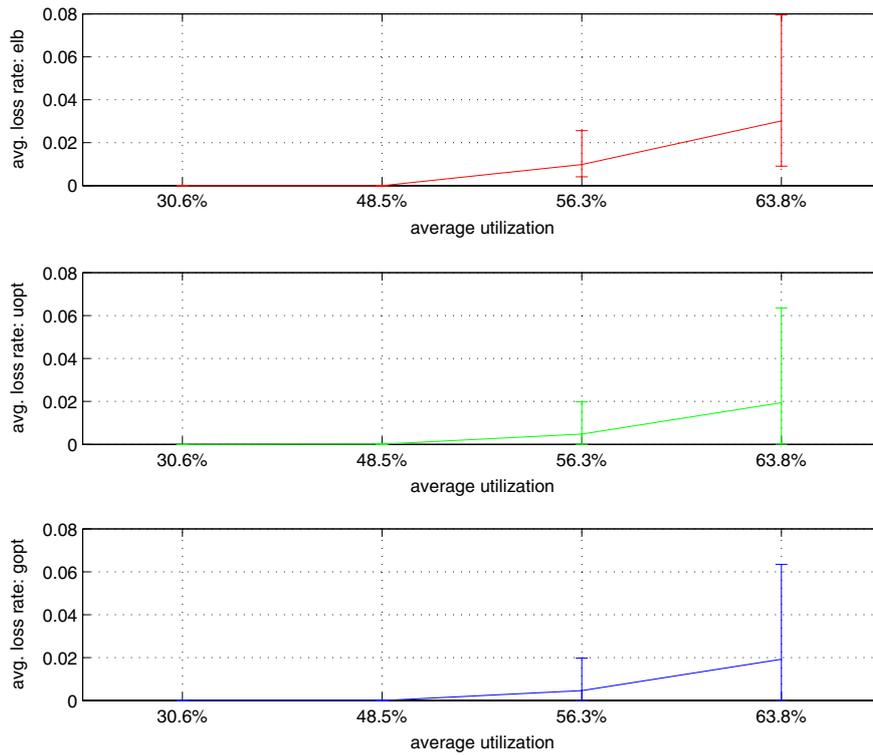


Fig. 5. Average loss rate of 30 simulations: average, minimum and maximum.

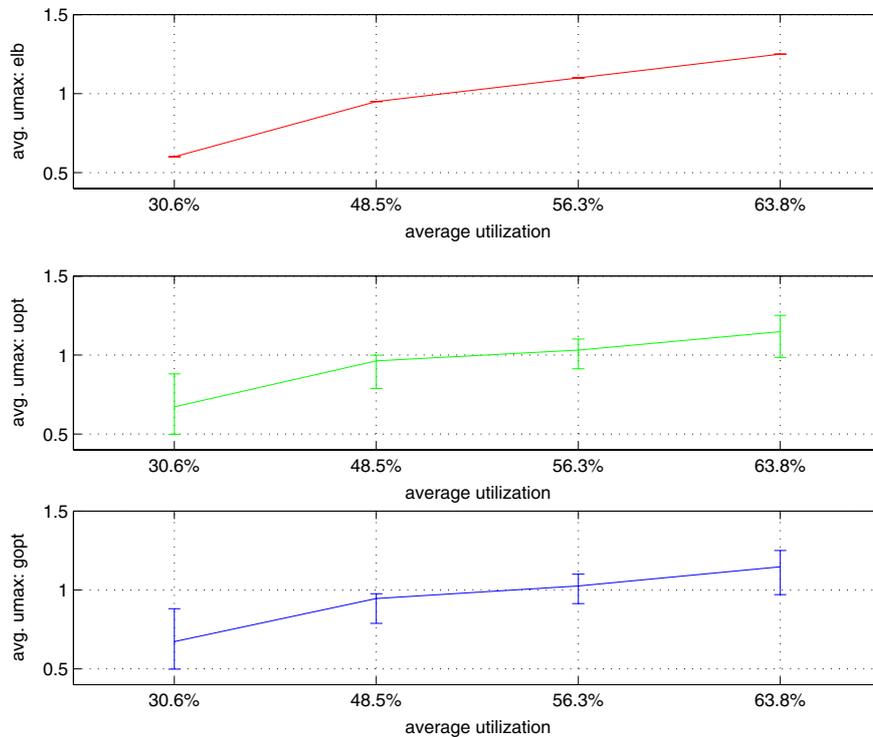


Fig. 6. Maximum link utilization of 30 simulations: average, minimum and maximum.

100% or exceeds 100%, the performance gains of user-optimal routing based MRC increase with link utilization. (3) User-optimal routing based MRC can achieve lower delays and lower loss rates at the same time.

### 5.3. Dynamic behavior

We study the convergence of our algorithm under following three types of scenarios: (1) we generate a random

$8 \times 3$  topology and a random traffic matrix, assign initial routing allocation according to the “static load balancing” algorithm and let the algorithm converge to user-optimal equilibrium; (2) at the equilibrium point, we select 30% of paths among the stub networks and increase the delay of them by 50 ms and let the algorithm converge; (3) at the equilibrium point, we select 30% of paths among the stub networks and mark the path as disconnected and let the algorithm converge. Here, we define convergence as

the state when the maximum difference of virtual delays of alternate paths used by traffic to one destination are not larger than 5 ms. To converge to a state where the maximum difference is not larger than 1 ms or less, it takes more time but gets similar overall performance. In the simulations, the traffic matrix is scaled such that the maximum utilization for the “elb” routing approach is 110%.

Results of two sets of simulations are shown in Fig. 7. The different sub-figures in each figure correspond to the

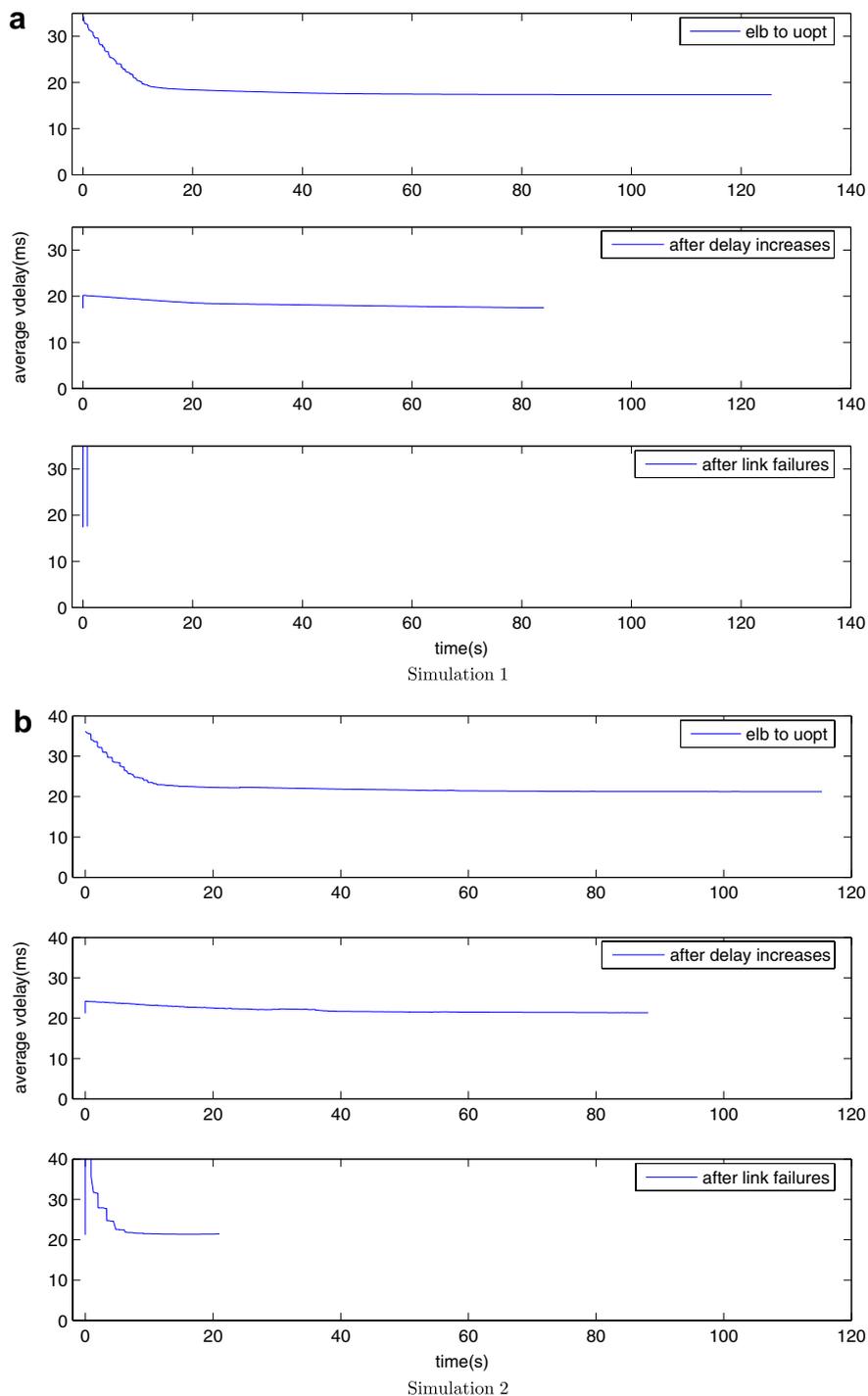


Fig. 7. Convergence of user-optimal routing based MRC in different scenarios (step size = 0.02).

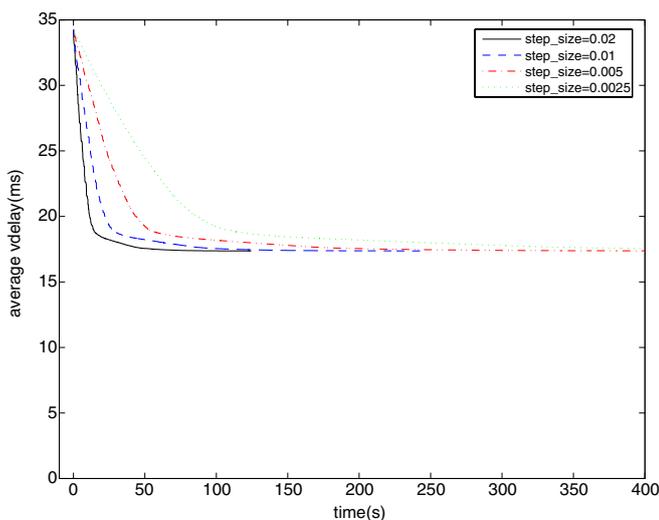


Fig. 8. Effect of different step sizes.

three scenarios (1), (2) and (3) explained above. We can see that the algorithm converges fast to a near-equilibrium point, in a few seconds. The convergence time for the “link failure” scenario is shorter than other scenarios because the algorithm responds to large virtual delay difference faster. The link failures cause traffic to be switched to other paths immediately and cause high virtual delay on some other paths.

We also study the affect of different step sizes on the convergence of the algorithm. The convergence from “static load balancing” to “user-optimal equilibrium” of one previous simulation (as shown in Fig. 7) is shown in Fig. 8. We can see that the algorithm converges quickly for several step sizes, converging faster with larger step sizes.

## 6. Future work

In this work, we assume the errors of measurement can be smoothed out. A stochastic approximation [29] analysis of the convergence of our algorithm when measurement results are corrupted by errors is desirable.

Although fixed step size is good for implementation of distributed MRC, properly decided adaptive step size may increase the convergence speed of the user-optimal routing based MRC.

## 7. Conclusions

In this paper, we have studied Multihoming Route Control (MRC) among a group of multihomed stub networks. We showed that simple greedy MRC approaches could cause oscillations and degrade the performance of the data communication among the group of stub networks. We proposed a user-optimal routing based distributed Multihoming Route Control scheme

that is simple to implement. We have shown through extensive simulations that the proposed MRC scheme improves performance in various network conditions without any oscillations. We have also shown that the user-optimal routing algorithm converges reasonably fast and achieves performance close to that of global optimal routing.

## References

- [1] P. Smith, BGP multihoming techniques, NANOG 23 (2001).
- [2] D. Passmore, Multihoming route optimizers. *Bussiness Communications Review* (2001).
- [3] A. Akella, B. Maggs, S. Seshan, A. Shaikh, R. Sitaraman, A measurement-based analysis of multihoming, in: *ACM SIGCOMM*, 2003, pp. 353–364.
- [4] Y. Liu, A.L.N. Reddy, Route optimization among a group of multihomed stub networks, in: *IEEE GLOBECOM*, 2005.
- [5] J. Wardrop, Some theoretical aspects of road traffic research, in: *The Institute of Civil Engineers*, vol. 1, 1952, pp. 325–378.
- [6] A. Akella, S. Seshan, A. Shaikh, Multihoming performance benefits: an experimental evaluation of practical enterprise strategies, in: *USENIX Annual Technical Conference, General Track*, 2004, pp. 113–126.
- [7] S. Tao, K. Xu, Y. Xu, T. Fei, L. Gao, R. Guérin, J.F. Kurose, D.F. Towsley, Z.-L. Zhang, Exploring the performance benefits of end-to-end path switching, in: *Proceedings of IEEE ICNP*, 2004, pp. 304–315.
- [8] D.K. Goldenberg, L. Qiu, H. Xie, Y.R. Yang, Y. Zhang, Optimizing cost and performance for multihoming, in: *ACM SIGCOMM*, 2004.
- [9] T. Roughgarden, É. Tardos, How bad is selfish routing? *J. ACM* 49 (2) (2002) 236–259.
- [10] L. Qiu, Y.R. Yang, Y. Zhang, S. Shenker, On selfish routing in Internet-like environments, in: *ACM SIGCOMM*, 2003, pp. 151–162.
- [11] D.P. Bertsekas, R. Gallager, *Data Networks*, 2nd ed., Prentice-Hall, Upper Saddle River, NJ, 1992.
- [12] J.N. Tsitsiklis, D.P. Bertsekas, Distributed asynchronous optimal routing in data networks, *Automatic Control, IEEE Transactions* 31 (4) (1986) 325–332.
- [13] W.K. Tsai, Convergence of gradient projection routing methods in an asynchronous stochastic quasi-static virtual circuit network, *Automatic Control, IEEE Transactions* 34 (1) (1989) 20–33.
- [14] T. Güven, C. Kommareddy, R. La, M. Shayman, S. Bhattacharjee, Measurement based optimal multi-path routing, in: *IEEE INFOCOM*, 2004, p. Mar.
- [15] A. Elwalid, C. Jin, S.H. Low, I. Widjaja, MATE: MPLS adaptive traffic engineering, in: *IEEE INFOCOM*, 2001, pp. 1300–1309.
- [16] K. Egevang, P. Francis, *The IP Network Address Translator (NAT)*, RFC 1631 (1994).
- [17] Y. Rekhter, T. Li, S. Hares, *A Border Gateway Protocol 4 (BGP-4)*, RFC 4271 (2006).
- [18] F. Guo, J. Chen, W. Li, T. Chiueh, Experiences in building a multihoming load balancing system, in: *IEEE INFOCOM*, 2004.
- [19] S. Sinha, S. Kandula, D. Katabi, Harnessing TCP’s Burstiness using flowlet switching, in: *3rd ACM SIGCOMM Workshop on Hot Topics in Networks*, 2004.
- [20] S. Bhandarkar, A.L.N. Reddy, TCP-DCR: Making TCP Robust to Non-congestion Events, in: *NETWORKING*, 2004, pp. 712–724.
- [21] M. Zhang, B. Karp, S. Floyd, L.L. Peterson, RR-TCP: A Reordering-Robust TCP with DSACK, in: *ICNP*, 2003, pp. 95–106.
- [22] N. Cardwell, S. Savage, T.E. Anderson, Modeling TCP latency, in: *IEEE INFOCOM*, 2000, pp. 1742–1751.
- [23] R. Braden, *Requirements for Internet hosts – communication layers*, RFC 1122 (1989).
- [24] M. Beckmann, C.B. McGuire, C.B. Winsten, *Studies in the Economics of Transportation*, Yale University Press, New Haven, CT, 1956.

- [25] M. Florian, D. Hearn, *Network Routing*, chapter 6. Network equilibrium models and algorithms. Elsevier Science, 1995.
- [26] Network Simulator - ns2, <http://www.isi.edu/nsnam/ns/>.
- [27] J. Kowalski, B. Warfield, Modeling traffic demand between nodes in a telecommunications network, in: ATNAC, 1995.
- [28] AT&T U.S. Network Latency, Feb. 2005, [http://ipnetwork.bgtmo.ip.att.net/pws/network\\_delay.html](http://ipnetwork.bgtmo.ip.att.net/pws/network_delay.html).
- [29] H.J. Kushner, D.S. Clark, *Stochastic Approximation Method for Constrained and Unconstrained Systems*, Springer-Verlag, New York, 1978.



**Yong Liu** received his B.S. degree and M.S. degree in Electrical Engineering from Peking University, China, in 1998 and 2001 respectively and his Ph.D. degree in Computer Engineering from Texas A&M University in 2006. He is now with EMC Corporation. His research interests are in network routing and operating systems.



**Narasimha Reddy** received a B.Tech. degree in Electronics and Electrical Communications Engineering from the Indian Institute of Technology, Kharagpur, India, in August 1985, and the M.S. and Ph.D degrees in Computer Engineering from the University of Illinois at Urbana-Champaign in May 1987 and August 1990 respectively.

Reddy is currently a Professor in the department of Electrical and Computer Engineering at Texas A&M University. Reddy's research interests are in Computer Networks, Multimedia Systems, Storage systems, and Computer Architecture. During 1990–1995, he was a Research Staff Member at IBM Almaden Research Center in San Jose, where he worked on projects related to disk arrays, multiprocessor communication, hierarchical storage systems and video servers.

Reddy holds five patents and was awarded a technical accomplishment award while at IBM. He has received an NSF Career Award in 1996. He was a faculty fellow of the College of Engineering at Texas A&M during 1999–2000. His honors include an outstanding professor award by the IEEE student branch at Texas A&M during 1997–1998, an outstanding faculty award by the department of Electrical and Computer Engineering during 2003–2004, a Distinguished Achievement award for teaching from the former students association of Texas A&M University and a citation “for one of the most influential papers from the 1st ACM Multimedia conference”. Reddy is a senior member of IEEE Computer Society and is a member of ACM.