A Local Optimization Algorithm for Logical Topology Design and Traffic Grooming in IP over WDM Networks

by Kwang-Il Lee, Mark Shayman

TR 2003-3

A Local Optimization Algorithm for Logical Topology Design and Traffic Grooming in IP over WDM Networks[#]

Kwang-Il Lee, Mark Shayman
Department of Electrical and Computer Engineering and Institute for Systems Research
University of Maryland, College Park, MD 20742
{kilee, shayman}@glue.umd.edu

Abstract

In this paper we investigate logical topology design algorithms using local optimization technique. Since the problem of the optimal logical topology design for all traffic demands is NP-complete, we design a logical topology by sequentially constructing the shortest path for one source-destination pair at a time. The path is a locally optimized path in the sense that there are no other paths with less hop count that may be constructed from existing links and newly created links. For this we define an Estimated Logical Hop Count (ELH), which is the shortest logical hop count for a given source and destination when it is applied. Also, we propose two heuristic logical topology design algorithms making use of ELH: ELH with Maximum Traffic Demands (MTD) and with Resource Efficiency Factor (REF). Finally, we evaluate the performance of the proposed algorithms by GLASS/SSF simulator. The simulation results show that ELH with REF outperforms other well-known algorithms in terms of the weighted hop count and network throughput.

[#] Research partially supported by the Laboratory for Telecommunications Sciences under contract MDA 90499C2521.

1. Introduction

Wavelength-division multiplexing (WDM) networks are considered as a promising technology for the next generation wide area networks because of their reconfigurability and plentiful bandwidth [3]. WDM networks set up lightpaths dynamically by reconfiguring the optical switches and can provide single hop communication channels between end nodes. This eliminates the electronic processing at intermediate nodes along the path and significantly reduces delay. However, it is generally impossible to provide single hop connectivity between each pair of end nodes due to limited number of router interfaces and other scalability issues. Consequently, it is necessary to have electronic switching over multiple lightpaths for traffic between some source and destination pairs [1,8].

Much research has been done since the early 90's on the logical topology design and traffic grooming problem. That research focused largely on the optimization of objective functions such as weighted hop distance [3,6,8] and maximum link utilization [7,8,10,11]. However, the problem of logical topology design and traffic grooming is known to be NP complete. So, many algorithms deal with direct (single-hop) connection setup between source and destination pairs using heuristic functions. And, traffic grooming for multi-hop traffic is typically left for routing policy at a higher layer such as IP or MPLS [6,8]. Even though there are some approaches to provide multi-hop connection by branch exchanges after logical topology design, branch exchanges are done with only some lightpaths, not all lightpaths [10,11]. The problem is that it is very difficult to get an optimal topology without consideration of traffic grooming because the volume of multi-hop traffic is often quite large and thus its performance significantly affects the overall performance of the network.

In this work, we investigate heuristic algorithms that integrate logical topology design and traffic grooming for multi-hop traffic. The general structure of these algorithms is as follows: The source-destination pair traffic demands are ordered according to some criteria and considered sequentially. When a demand is considered, the algorithm makes the choice that is locally optimal in the sense that the demand is placed on a path that has the minimum possible number of logical hops considering all topologies that refine the partial topology existing when that demand is considered. We use simulation to investigate the performance of two algorithms of this type that differ in the criteria they use for ordering the demands. We show that one of these new algorithms outperforms the well-known existing algorithms for logical topology design.

This paper is organized as follows. Section 2 reviews related work for logical topology design and traffic grooming. Section 3 gives notations and objective functions used in this paper. We describe the local optimization problem for logical topology design and traffic grooming, and propose two heuristic algorithms in Section 4. And, in Section 5, we analyze the performance of the algorithms using various metrics and compare the performance to that of other proposed schemes. Finally, we conclude the paper in Section 6.

2. Related Work

[3,6] propose several heuristic logical topology design algorithms for optical networks. The primary goal of these algorithms is to construct logical topologies in order to maximize the single hop traffic. After designing the logical topology, they map residual multi-hop traffic onto the logical topology. The fundamental distinction between [3] and [6] is in the initialization of the logical topology.

[1,4,8] propose lightpath setup algorithms that consider either physical or logical hop count in the topology design. [1] uses physical hop count value for the computation of link utilization factor. Based on the link utilization factor, the lightpaths are setup based on interface availability in source and destination. And, [4,8] tries to minimize delay by providing direct lightpaths for source and destination pairs that have longer logical hop count. Traffic demands weighted by the logical hop count (relative to the incomplete logical topology) are sorted in descending order and lightpaths are established in that order. However, these algorithms only consider the case when interfaces are available in both source and destination. Grooming of multi-hop traffic during logical topology design is not considered.

[8] proposes a lightpath deletion algorithm for logical topology design. The algorithm first builds a fully meshed logical topology and deletes the lightpaths with lowest link utilization until all constraints are satisfied. And, [10] constructs an initial logical topology and assigns flows onto the topology. After that, it re-configures some lightpaths by branch exchanges in order to maximize the objective functions. This algorithm takes an optimization strategy after the logical topology design, but it does not deal with the optimization of the initial logical topology.

3. Logical Topology and Traffic Grooming Problem

The objective of the problem is to determine the logical topology and path assignments so as to optimize the objective functions for given traffic demands. The general problem is stated in many papers [1,3,5,7,8]. So, we define some basic notations and objective functions used in this paper.

3.1 Assumptions

The logical topology design describes the lightpath setup problem with constraints in optical networks. In our work we are mainly focused on the problem with one constraint, the number of electronic interfaces (degree), and other constraints are not considered. So, we will assume that sufficient wavelengths and wavelength converters are available so that whenever router interfaces are available at the end nodes, a lightpath can be setup--i.e., the routing and wavelength assignment problem is always solvable. This assumption has been made elsewhere in the literature [3,6].

Traffic grooming deals with the issue of the traffic mapping onto the logical topology. So, it finds the optimal path(s) to reach the destination satisfying the constraints. This can be achieved in several ways. In this paper, we require that all the traffic for a given source-destination pair use the same path. Hence, multi-path issues are not considered.

3.2 Notation

In this paper, we will use the following notation.

$G^o=(V,E^o)$	Optical network (physical) topology consisting of a weighted unidirectional graph, where V represents the set of integrated router-OXC nodes, and E^o represents the set of optical (physical) links.
$G^l=(V,E^l)$	Logical (virtual) topology. This is the output of the logical topology design
T	The traffic matrix T given by an N N matrix, where $N=\|V_o\|$. Each entry t_{sd} of the matrix represents aggregated traffic demands from source s to destination d.
LP^z_{sd}	a logical path z connecting from node s to node d consisting of a set of optical light paths. This is the output of the traffic grooming. $=\{E^{l,k_1}_{sd_1}, E^{l,k_2}_{d_1d_2}, \ldots, E^{l,k_m}_{d_md}\}$
Delay(z)	delay for a logical path z. The delay includes propagation delay incurred in the optical network and electronic processing delay at each intermediate router.
BW(z)	the bandwidth used (load) for a logical path z.

3.3 Objective Function

The goal this paper is to minimize the weighted delay and maximize the network throughput as shown in Equation (1) and (2). Since delay is mainly due to the electronic processing at the intermediate nodes, the delay can be measured in terms of average weighted hop count by replacing delay with hop count. And, the network throughput can be measured by the total traffic amounts accommodated by the logical topology. This is computed by the summation of total bandwidth of each traffic groomed logical path.

$$\text{Minimize: } \frac{\sum_z BW(z) \times delay(z)}{\sum_z BW(z)} \quad \text{and} \qquad \text{-------(1)}$$

$$\text{Maximize: } \sum_z BW(z), \forall z \qquad \text{-------(2)}$$

4. Heuristic Algorithms with Local Optimization

In this section, we propose two heuristic logical topology and traffic grooming algorithms so as to optimize our objective functions. The basic idea of the algorithms is to set up multi-hop lightpaths by considering logical topology design and traffic grooming simultaneously using a local optimization approach. This is enabled by making use of optical and logical topology graphs respectively denoted as $G^o = (V,E^o)$ and $G^l=(V,E^l)$.

4.1 Local Optimization Problem

The local optimization problem can be stated as follows: Given a partial logical topology with a set of traffic demands assigned to paths in this topology, and a source-destination pair of nodes s,d with traffic demand t_{sd}, find the shortest logical path from s to d with available bandwidth at least t_{sd} using either existing (logical) links or a combination of existing links and new links to be created. Let us consider a simple example explained in Figure 1. In this example, we assume that the number of interfaces at each node is two and lightpaths are bi-directional. We also assume that each existing link has sufficient residual bandwidth to accommodate the traffic demands being considered.

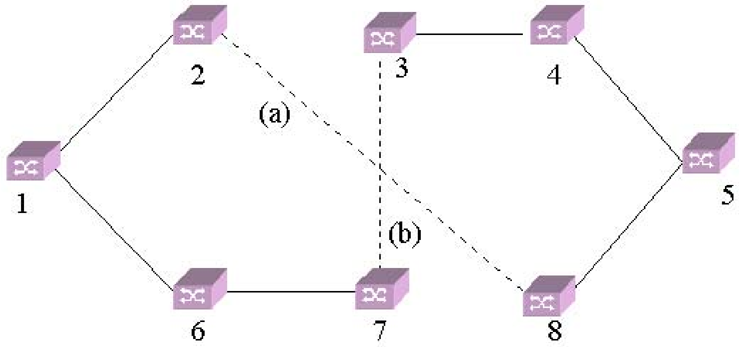

Figure 1. Example

In Figure 1, the dashed links represent potential links that are not setup yet. Now, we consider the situation for providing a path between node 2 and node 8 to accommodate a traffic demand t_{28}. This is very trivial. Since node 2 and node 8 each have an interface available, a direct logical link (a) can be setup. In this case, the hop count between node 2 and node 8 is one. Then, we consider the path provisioning between node 1 and node 5 to accommodate a traffic demand t_{15}. In this case, a direct link between the two nodes is not possible since neither node has a free interface. In the network, the only possible lightpath that can be setup is a (potential) link between nodes 3 and 7. If the logical path between node 1 and node 5 takes this logical link (b), then the hop count for the path between node 1 and node 5 is five. This is quite large. The alternative is to provide a logical path between node 1 and node 5 by traffic grooming. When we compute the shortest path for traffic grooming, we choose path (1-2-8-5) and the hop count is three. This is two hops less than the path obtained by adding a new lightpath. So, traffic grooming provides the better path between nodes 1 and 5.

If s has an available transmitter and d has an available receiver, then the shortest path is the one hop path obtained by creating a direct link from s to d (since we are assuming the optical network has resources to create such a lightpath). Otherwise, the shortest path will be a multi-hop path.

Since a lightpath can be setup only between nodes that have available interfaces the multi-hop lightpath setup problem can be defined as a node search problem as follows: We are given a partial (logical) topology with a set of traffic demands assigned to paths in this topology. We refer to the residual bandwidth of a link as the *available link bandwidth* (ALB). Let t be a given traffic demand. For each node x, let H(s,x,t) denote the minimum hop distance from s to x considering only links with ALB at least t. Given s and t, let f(s,t) denote the node x that minimizes H(s,x,t) *among those nodes that have available transmitters.* Similarly, let f^R(d,t) denote the node y that minimizes H(y,d,t) *among those nodes that have available receivers.* In case of a tie we choose a node with maximum ALB.

Given a traffic demand t_{sd} from s to d, let x = f(s,t_{sd}) and let y = f^R(d,t_{sd}). Assume that there is no existing direct link from x to y. We claim that the shortest path from s to d that has available bandwidth t_{sd} and includes at least one new link consists of the path from s to x with length H(s,x,t_{sd}), the newly created direct link from x to y, and the path from y to d with length H(y,d t_{sd}). We denote this path by P(s,x,y,d,t_{sd}). To prove this, first note that this path is at least as short as any other path with available bandwidth t_{sd} that contains exactly one new link. However, if there is a path that contains more than one new link, we can shorten that path by establishing a direct link between the node at the head end of the first new link and the node at the tail end of the last new link in the path. So no path with more than one new link can be optimal.

Suppose instead that in the partial topology there is already an existing link from x to y. In this case P(s,x,y,d,t_{sd}) is a path that uses only existing links. However, it still follows from the definitions of x and y that any path from s to d with bandwidth t_{sd} and containing at least one new link cannot be shorter than P(s,x,y,d,t_{sd}). (There may be a shorter path consisting only of existing links.)

It follows from the preceding arguments that the locally optimal path from s to d for the demand t_{sd} is either P(s,x,y,d,t_{sd}) or a path that uses only links that already exist in the partial topology--i.e., a traffic groomed path. Given the partial topology with already assigned traffic demands as described above, let LTD(s,d, t_{sd}) denote the length of the path P(s,x,y,d,t_{sd}) that requires addition of one link and let TG(s,d, t_{sd}) denote the length of the shortest "traffic groomed" path from s to d--i.e., the shortest existing path having available bandwidth of at least t_{sd}. We define the Estimated Logical Hop Count (ELH) as

$$\mathrm{ELH}(s,d,\ t_{sd}) = \mathrm{Min}[\mathrm{LTD}(s,d,\ t_{sd}),\ \mathrm{TG}(s,d,\ t_{sd})].$$

It is an estimate for the optimized hop count for the source-destination pair s,d with demand t_{sd}. From the preceding analysis we have the following result.

Theorem 1. Given a partial topology with traffic demands assigned to paths, ELH(s,d, t_{sd}) is equal to the number of hops in the locally optimal path from s to d with bandwidth t_{sd}. Any path that may be constructed from existing links and newly created links has at least length ELH(s,d, t_{sd}) .

4.2 Heuristic Algorithms

To obtain a heuristic algorithm, we couple local path optimization with a rule that specifies the order in which source-destination pairs should be considered. We will see that the ordering of source-destination pairs has a significant impact on the effectiveness of the local optimization. Here, we propose two heuristic algorithms that differ in the way they order the source-destination pairs.

1) Maximum Traffic Demands

A simple approach is to select at each step the source-destination pair with maximum traffic demand that has not yet been considered. In this approach, the traffic matrix is sorted in descending order and locally optimal paths are chosen sequentially. Whenever a path includes a link that does not already exist, that link is added to the partial logical topology.

[Algorithm 1] ELH with Maximum Traffic Demands

```
Step 1    Find s' and d' , $t_{s'd'} = \max[t_{sd}]$ for all s,d
Step 2    Compute a logical path $LP^{z}_{s'd'}$ for s'-d' pair
          containing  ELH(s',d', $t_{s'd'}$) hops
Step 3.1  $E^{l} = E^{l}\ \{E^{l}_{x'y'}\}$ if $E^{l}_{x'y'}\ LP^{z}_{s'd'}$, and $E^{l}_{x'y'}\ E^{l}$
Step 3.2  $T = T - \{t_{s'd'}\}$
Step 4    If T is empty, DONE
              Otherwise, go to Step 1
```

2) Maximum Resource Efficiency

When a traffic demand is assigned to a path, the efficiency with which it uses logical network resources depends on the number of (logical) hops; fewer hops means more efficiency. In this paper, we propose an algorithm that uses a resource efficiency factor. This value is computed by the division of traffic demand by the ELH. At each step of the algorithm, we select a source and destination pair with the maximum value and either groom the traffic demand or setup a lightpath.

[Algorithm 2] ELH with Resource Efficiency Factor

```
Step 1    Calculate $ELH(s,d, t_{sd})$ for all s,d
Step 2    Find s'and d' , $t_{s'd'}/ ELH(s',d', t_{s'd'}) = \max[t_{sd}/ ELH(s,d, t_{sd})]$
Step 3    Compute a logical path $LP^z_{s'd'}$ for s'-d' pair
              containing $ELH(s',d', t_{s'd'})$ hops
Step 4.1  $E^l = E^l \ \{E^l_{x'y'}\}$ if $E^l_{x'y'} \ LP^z_{s'd'}$, and $E^l_{x'y'} \ E^l$
Step 4.2  $T = T - \{t_{s'd'}\}$
Step 5    If T is empty, DONE
              Otherwise, go to Step 1
```

5. Performance Analysis

5.1 Simulation Environment

We analyze the proposed integrated logical topology design and traffic-grooming schemes through simulations using GLASS/SSF simulator[12,13]. We consider a 16-node NSFNet network topology as shown in Figure 2. We assume that all nodes have both OXC and router functionality. Also, each node has the capability to perform wavelength conversion so there is no wavelength continuity problem and each link has unlimited number of wavelengths. So, lightpaths can always be set up if the degree constraints are not violated. In our simulations, each node has five transmitters and receivers. The capacity of each wavelength is normalized to one bandwidth unit (BU) in our model.

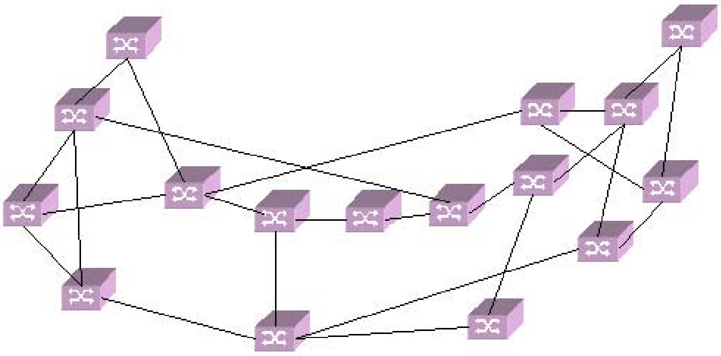

Figure 2. Network Topology

Each entry in the traffic matrix represents the aggregated traffic demand of a source-destination pair. It is generated independently using the uniform distribution between 0 and 0.5 BU. For the analysis, we used 15 different traffic matrices in our experiments. We compare our two algorithms with other logical topology design algorithms such as MRU[1], HLDA [3]and DLPA [8]. DLPA is a logical topology algorithm which deletes low utilized lightpaths from an initial fully-meshed topology[8].

5.2 Analysis

We measured the weighted hop distance value and network throughput as performance metrics as shown in Figures 3 and 4. In the figures, ELH-REF algorithm works better than any other algorithm as measured by either weighted hop distance or network throughput. ELH-REF reduces the weighted hop distance 8 to 19% and average 13%. Also, it increases the network throughput 9 to 16.7% and average 12% compared to other algorithms. This confirms that the resource efficiency factor in ELH-REF helps lightpaths be setup in order to maximize the network throughput as well as to provide shorter paths between nodes.

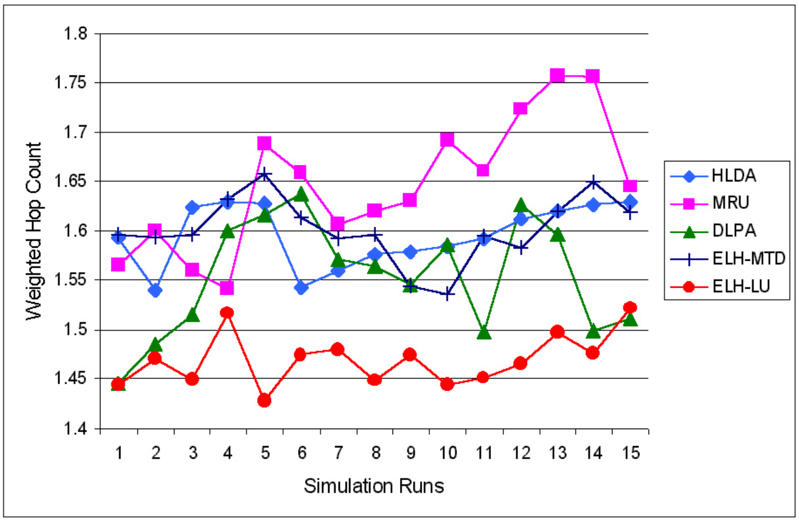

Figure 3. Weighted Hop Distance

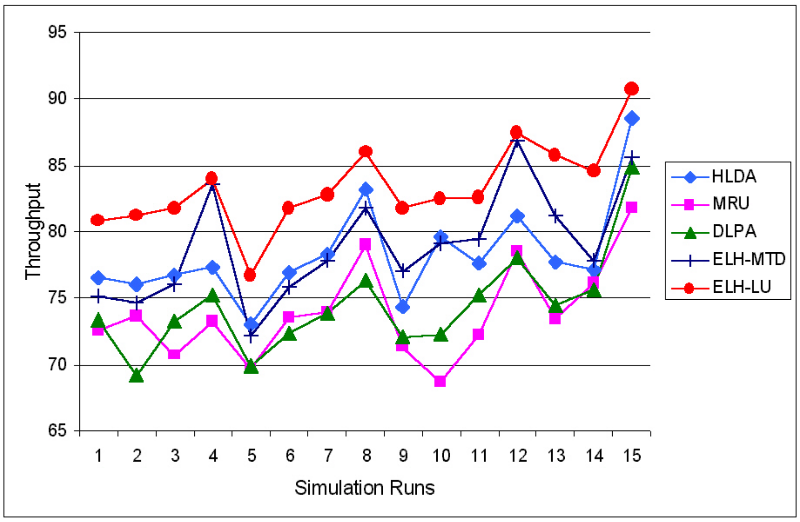

Figure 4. Network Throughput

ELH-MTD does not show as good results as ELH-REF. Our observation is that some lightpaths established for multi-hop traffic in ELH-MTD are underutilized and make some logical paths longer. So the performance of the algorithm depends on the utilization of the multi-hop lightpaths. This confirms that the optimization of the multi-hop traffic is critical for the performance. Also, HLDA that maximizes single hop traffic shows similar behavior as ELH-MTD.

In our experiments, we found that the MRU algorithm showed poorer performance. While ELH-REF divides traffic demands by logical hop distance, MRU divides traffic demands by *physical* (optical) hop distance. This makes MRU effective at optimizing the use of optical layer resources but not especially effective at optimizing the performance metrics we considered.

Lightpath deletion approach such as DLPA shows lower weighted hop distance and network throughput. DLPA deletes lower traffic demands one by one. During the deletion, lower traffic flows passing through deleted links are remapped into other links. Because the lower utilized links are deleted and remapped first, the higher traffic flows that are remapped later may be forced to take relatively longer paths or be blocked if enough network resources are not available.

6. Conclusion

In this paper, we describe the local optimization problem for logical topology design and traffic grooming. Because the consideration of all traffic demands in the logical topology design is NP-complete, we design the logical topology so as to provide an optimal path for one source and destination pair at a time. The optimal path is computed by considering logical topology and traffic grooming together. The length of the locally optimal path is called Estimated Logical Hop Count (ELH). And, we propose two heuristic algorithms using ELH: ELH-MTD and ELH-REF. We perform simulation analysis using GLASS/SSF simulator. By the simulations, we observed that ELH-REF shows better performance in terms of delay and network throughput than other known algorithms.

Reference

[1] K. Zhu, B. Mukherjee, “Traffic Grooming in an Optical WDM Mesh “Networks”, IEEE JSAC, pp. 122-133, Jan. 2002.

[2] M. Kodialam, T. V. Lakshman, “Integrated Dynamic IP and Wavelength Routing in IP over WDM Networks”, IEEE Infocom, pp. 358-366, 2001.

[3] R. Ramaswami, K. N. Sivarajan, “Design of Logical Topologies for Wavelength-Routed optical Networks”, IEEE JSAC, pp. 840-851, Jun. 1996.

[4] B. Mukherjee, Optical Communication Networks, New York, McGraw-Hill, 1997.

[5] D. Banerjee, B. Mukherjee, “Wavelength-routed Optical Networks: Linear Formulation, Resource Budgeting Tradeoffs and a Reconfiguration Study”, IEEE/ACM Trans. Networking, vol. 8, pp. 598-607, Oct. 2000.

[6] K. H. Liu, C. Liu, J. L. Pastor, A. Roy, J. Y. Wei, “Performance and Testbed Study of Topology Reconfiguration in IP over WDM”, IEEE Transactions on Communications, in press, 2002.

[7] R. Dutta, G. N. Rouskas, “A Survey of Virtual Topology Design Algorithms for Wavelength Routed optical Networks”, Optical Network Magazine, Jan. 2000.

[8] E. Leonardi, M. Mellia, M. A. Marsan, “Algorithms for the Logical Topology Design in WDM All-Optical Networks”, Optical Networks Magazie, pp. 35- 46. Jan. 2000.

[9] A. L. Chiu, E. H. Modiano, “Traffic Grooming Algorithms for Reducing Electronic Multiplexing Costs in WDM Ring Networks”, Journal of Lightwave Technology, vol. 18, no. 1, pp. 2-12, Jan. 2000.

[10] J-F, P. Labourdette, A.S. Acampora, “Logically Rearrangeable Lightwave Networks”, IEEE Transactions on Communications, pp. 1223-1230, Aug. 1991.

[11] A. S. Acampora, G. W. Hart, J.-F. P. Labourdette, **“**Branch-Exchange Sequences for Reconfiguration of Lightwave Networks”, IEEE Transactions on Communications, Vol. 42, No. 10, pp. 2822 –2832, Oct. 1994.

[12] GLASS simulator, http://dns.antd.nist.gov/glass/

[13] SSFnet, http://www.ssfnet.org