On the Performance of Shortest Path Routing Algorithms for Modeling and Simulation of Static Source Routed Networks – an Extension to the Dijkstra Algorithm

Nuno M. Garcia[1,2], Przemyslaw Lenkiewicz[2], Mário M. Freire[2], Paulo P. Monteiro[1,3]
[1] *Nokia Siemens Portugal S.A., Amadora, Portugal*
[2] *IT - Networks and Multimedia Group, Department of Computer Science, University of Beira Interior, Covilhã, Portugal*
[3] *Instituto de Telecomunicações, University of Aveiro, Aveiro, Portugal*
nuno.mgarcia@siemens.com, przemek.lenkiewicz@gmail.com, mario@di.ubi.pt, paulo.1.monteiro@nsn.com

Abstract

Shortest path routing algorithms, such as Dijkstra's algorithm present an overload problem when used to define routes for ring topologies in networks that implement source routing. This paper presents the effects of Dijkstra's shortest path routing in the simulation and modeling of static source routed networks, in particular we evaluated the effect of this routing scheme in the performance of Optical Burst Switched (OBS) networks. A new static shortest path algorithm is presented and its performance compared with the standard shortest path algorithm, using two new metrics. We propose the use of this routing algorithm in network simulators instead of standard Dijkstra, as it produces more symmetric and balanced routes over the network links, thus producing results that are closer to real networks which implement a more dynamic routing.

1. Introduction

Routing is defined as the task of moving data packets across a network from a source point to a destination point. This usually involves two distinct phases – the first is to find the optimal routing paths, taking into account a set of rules and constraints; and the second is the actual transport of the data packets through the network via the previously established path. The critical phase of routing is to select the optimal routing path, due to the complexity and large dimension of network topologies and to the very often large number of rules and constraints that have to be met. Source routed networks define its routing path tables at the ingress nodes, and for static source routed networks these paths remain unchanged unless reconfiguration is necessary by the addition or removal of nodes or links.

Routing Algorithms are responsible for the selection of the optimal path, and can be classified following a series of key features. When a network device needs to compute the next hop or the full path for a given data packet (or stream of packets, in the case of a circuit definition), some metrics are used to define the optimal property of the candidate paths. These metrics may include number of hops (or path length), the reliability, the available bandwidth, the load applied to the path, the delay the path will inflict to the packets, or even the monetary cost of using that particular path. If all constraints are considered equal to each of the possible hops in the network, only one constraint remains as a selection criterion – the length of the path (or number of hops, if all links are supposed of equal length). This length will depend of the other constraints. For instance, it will be proportional to the monetary cost of the path, inversely proportional to the path reliability, etc. The Dijkstra's algorithm [1] (DA) is the *de facto* standard for shortest path route selection, mostly due to its simplicity and efficiency. Popular network simulators as ns-2 [2], OWns [3], OBSim [4], OIRC OBS-ns Simulator [5] and others rely on the DA to compute the static paths that are used throughout the simulation process.

This paper describes the inadequacy of the use of Dijkstra's shortest path routing when modeling and simulating static source routed networks that include rings, such as e.g. Optical Burst Switched (OBS) networks. OBS networks were initially proposed by Qiao and Yoo around 1999 [6] and implement source routing (for more detail on OBS networks, please refer to [6]). Furthermore, from a simulation point of view, OBS routing is static, i.e., a path is defined for a set of

Second International Conference on Systems and Networks Communications (ICSNC 2007)
0-7695-2938-0/07 $25.00 © 2007 IEEE

ingress and egress points and kept throughout the simulation. When simulating this class of networks, DA tends to introduce unnecessary link overload, and thus induces into false conclusions. An extension to Dijkstra's is proposed and its performance assessed and compared with Dijkstra's.

The remainder of this paper is organized as follows: section two presents the shortest path algorithm and the problem raised when used on ring networks; section three presents the new algorithm; section four presents the comparative performance assessment for the two algorithms and section five concludes the paper.

2. Shortest path routing

Dijkstra's algorithm was designed to solve the single-source shortest path problem for a directed graph with non-negative weights. Real telecommunication networks fall in this class of graphs, although very often each link is bidirectional and, as so, it must be considered as a pair of opposite direction graph edges.

Open Shortest Path First (OSPF) [7] is a well known real-world implementation of DA used in network routing. In real networks, particularly in Ethernet networks, the Spanning-Tree Protocol (STP) [8] runs on the network before the OSPF. In a general way, a spanning tree of a graph is a sub-graph which is also a tree that contains all the nodes. In other words, in a network environment, where redundant links are common, the STP causes these links to appear closed for the operation of the network elements, as to eliminate the appearance of duplicate messages, such as e.g. neighbor discovery messages.

Rings are a particular interesting class of topologies in optical networks, because they allow an additional level of connectivity for each node (there are now two possible paths to the destination node instead of one), with the cost of a single additional link. Rings are common elements in existing or planned networks, such as the European Optical Network (EON) [9] or the NSFnet. Figure 1 and Figure 2 show a short version of EON (also termed COST 239) and the NSFnet network – in both figures, among others, several four node ring sub networks, can be detected, e.g. Amsterdam, Berlin, Prague, Luxembourg for EON and Pittsburgh, Princeton, Boston and Ithaca for NFSnet.

Due to the iterative nature of DA and also of other static routing algorithms, in networks that are not tree like, i.e. include links that create rings, the algorithm tends to place more paths over some links than over others, i.e., in the scenario of equal cost links, if a ring has a number even of nodes, then for any node there will always be two possible paths to reach the opposite node in the ring, and following the iterative nature of the algorithm, one of the links will be chosen more often than the others.

In real world networks, path and load balancing algorithms tend to utilize efficiently all the available connection links, e.g. using Equal Cost Multi Path (ECMP) algorithms [7], so this problem may not exist at all. But in simulators, very often DA is the only available routing tool and thus its output will be biased.

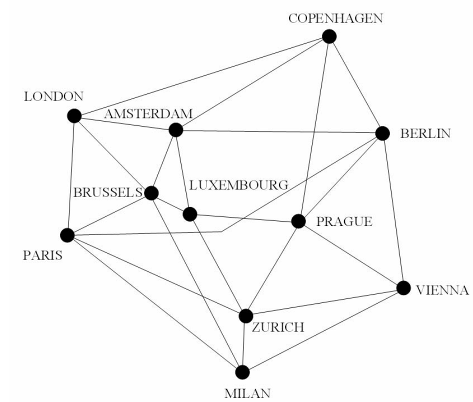

Figure 1 – The 11 node EON network topology.

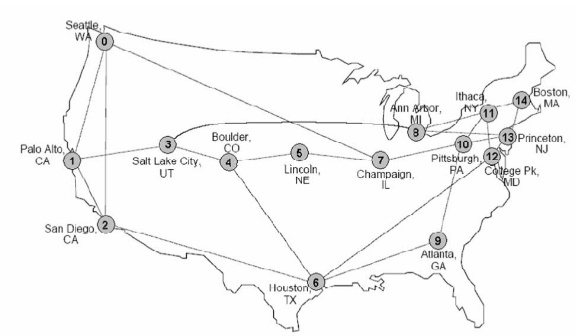

Figure 2 – The 14 node NSFnet network topology.

To better illustrate the nature of the problem, we use the term *path* to refer to a set of links over which packets travel between their ingress and egress point in the network, and *route* to the event of any packet using a link in a *path*. Within this context, a packet follows a *path* and a link has as many *routes* as many different paths use that link.

A bidirectional link l is defined as a pair (n_i,n_j) where n_i and n_j are two connected nodes. Note that $l=(n_i,n_j)=(n_j,n_i)$, although the number of routes that use it may be different according to each of its directions,

$$R(n_i,n_j)=\vec{R}_l;\ R(n_j,n_i)=\overleftarrow{R}_l. \qquad (1)$$

Furthermore, L is defined as the set of all links l_i and $\#L$ is its dimension.

If the network is fully connected, the number of routes that use any given link l in the network, is

$$R_l = \frac{\sum_{i \in L} R_i}{\# L} = 2 \qquad (2)$$

because of the bidirectional nature of the link. The asymmetry and heterogeneity in network routes result of the need to use some links more than others when a network is not fully connected. The creation of paths over a network topology, i.e. the assignment of these routes over network links performed by an iterative algorithm such as DA, will always prefer the first available link in the selection list, independently of concerns of overload and unbalance for that link.

Having this scenario in mind, we define two additional metrics to allow the assessment of the efficiency of a routing algorithm in terms of the homogeneous distribution of the number of routes and paths by the available links: balance and symmetry.

Let us define a balanced network as one that respects the following condition: for any link l, the sum of number of routes that use l is equal or very close to the number of routes created for any other link, i.e.

$$\vec{R}_l + \overleftarrow{R}_l \approx \frac{\sum_{i \in L} R_i}{\# L}, \forall l \in L \Leftrightarrow \sigma(R_i) < \varepsilon, i \in L \qquad (3)$$

where σ is the standard deviation for the values of R_i ($i \in L$), ε is a function of the number of routes, desirably small.

By a symmetric network, we understand that for any given link l, the number of routes that use the link in one direction is equal or close to the number of routes with the opposite direction, i.e.,

$$\vec{R}_l \approx \overleftarrow{R}_l, \forall l \in L. \qquad (4)$$

Cumulatively, a balanced symmetric network is

$$\vec{R}_l \approx \overleftarrow{R}_l \approx \frac{\sum_{i \in L} R_i}{2.\# L}, \forall l \in L. \qquad (5)$$

The smallest ring network that is not fully meshed is the 4 node ring network, as shown in Figure 3. The number of the routes that use the links on a four node ring network following two different implementations of a shortest path Dijkstra's like algorithm are depicted in Figure 3 A and B. Although we kept the order of the nodes (from 1 to 4, clockwise), this is not relevant for the results, as changing it would result in a permutation of the routes on the links.

As we can see, none of the network's graphs is symmetric. In Figure 3 A, each link receives 300% more load in one direction than in the other. This is also visible for Figure 3 B for links 1-2 and 3-4. Links 1-4 and 2-3 in Figure 3 B are balanced and symmetric.

We have found that this difference has higher impact in ring networks that have an even number of nodes, with the maximum asymmetry for the 4 node ring network. The reason for this fact is that in a network with an even number of nodes, there will be two equally expensive shortest paths between two diametrically opposed nodes, e.g., in the network depicted in Figure 3, between nodes *1* and *3* and nodes *2* and *4*. Likewise, if the network has more than four nodes, the routes that overload the links constitute a smaller share of the overall routes created and thus the asymmetry is higher in a 4 node ring network. If the number of nodes is odd and the links have equal costs, there will be no two equal cost possible paths between any two network points.

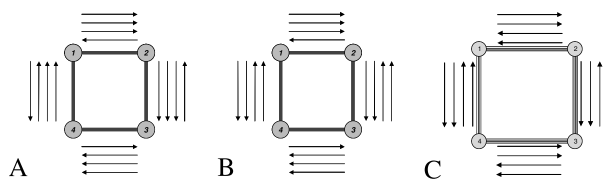

Figure 3 – Dijkstra's algorithm routes (version A and version B) in a four node ring network and routes for Extended Dijkstra (version C).

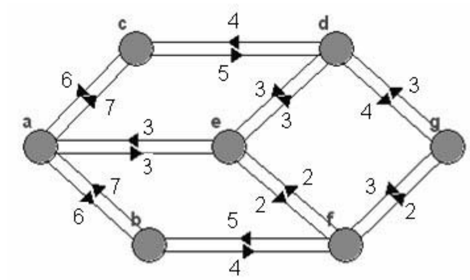

Figure 4 – Seven node network with nine bidirectional links showing routes per link.

Figure 4 shows a network with seven nodes and nine links. This network is actually composed of 3 four node rings sharing some of its links. We applied Dijkstra's algorithm to this topology and observed the number of routes that used each link. The effect of the iterative nature of DA is well observed on the routes created of the links shown in Figure 4, i.e., the links for the first nodes (a, b and c) bare almost three times more routes than the last nodes (e, f, g). Observing Figure 4 we conclude that the algorithm produces quasi-symmetric routes. Yet, we can see that link *a-c* is used in 7 paths, while link *f-g* is used in only 2 paths. If we set to simulate the performance of this network using DA, e.g. in the case of an OBS network, there will be a higher burst loss probability ratio for the nodes whose output links have received more routes. Similar conclusions may be drawn for any other simulations

Second International Conference on Systems and Networks Communications (ICSNC 2007)
0-7695-2938-0/07 $25.00 © 2007 IEEE

regarding static source routed packet or burst switched networks.

DA is defined as follows: a graph $G = (V, E)$ is defined, where V is a set of vertices and E is a set of edges. We denote S as the set of vertices whose shortest paths from the source have already been determined and $V\text{-}S$ are the remaining vertices. The other data structures needed are: d - an array of best estimates of shortest path to each vertex and p_i - an array of predecessors for each vertex.

The basic mode of operation is:

1. Initialize d and p_i,
2. Set S to empty,
3. While there are still vertices in $V\text{-}S$,
   i. Sort the vertices in $V\text{-}S$ according to the current best estimate of their distance from the source,
   ii. Add u, the closest vertex in $V\text{-}S$, to S,
   iii. Relax all the vertices still in $V\text{-}S$ connected to u.

The relaxation process (step 3.iii) updates the costs of all the vertices v, connected to a vertex u, if the best estimate of the shortest path to v by including (u,v) in the path to v can be improved. More complete descriptions of DA are extensively available in the Internet, e.g. in [10].

3. Extension to Dijkstra's Algorithm

Having in mind the limitations of the DA, we defined a new algorithm that aims to solve them. For that matter, we introduce the following concept: in a graph, each node is identified by a unique number, termed the node identifier, without any special order. We extended the initial DA as to detect possible equal cost routes, and use additional conditions based on the nodes identifiers to select on those routes. After the sorting phase (3.i), (u is already defined)

3.i.a. if there is another candidate u' to be the best vertex (that is, with equal cost), then

3.i.b. if the sum of the node identifiers related to u is equal to the sum of node identifiers related to u', then

3.i.b.1. choose u as the node that has the first lowest neighbor,

else

3.i.b.2. if the identifier for the source-node s of the path is lower than the identifier for the destination-node v of the path, then

3.i.b.2.1. choose u as the node which path has the highest sum of node identifiers,

else

3.i.b.2.2. choose u as the node which path has the lowest sum of node identifiers.

3.i.c. repeat 3.i.a. until there are no left candidates

As these additional conditions only run if there are two or more path candidates to shortest path, the Extended Dijkstra (ED) still provides strictly shortest paths. Appling the new algorithm to the test case of the four node ring network shown in Figure 3, we have new routes as shown in Figure 3 C, these being fully balanced and symmetric.

4. Performance Assessment

The performance assessment of the new algorithm was performed with the modified version of the simulator build in [4]. The OBS simulation parameters used are fully described in [4].

Two different metrics were obtained through simulation: number of routes created over each unidirectional link, and number of bursts dropped in the network. All the network simulation parameters were kept unaltered, except for the selected routing algorithm. Tested topologies were the four node ring, six node ring, eight node ring, two topologies with 7 and 9 nodes made of adjacent four node rings, and 11 nodes EON / COST 239 network. We present the OBS performance comparison figures for several topologies at different load scenarios.

Figure 5 shows the improvement obtained by the mere change on the routing algorithms. This figure shows three burst loss scenarios, around 50%, around 10% and around 1% for the following networks: 4 node ring, 6 node ring, 8 node ring and 9 nodes topology with 4 four node ring networks sharing links. In all the scenarios, the ED shows improvement in the burst loss ratio. As expected, more loaded networks show a smaller improvement – it is well known that overloaded OBS networks do not have room for additional improvement [6]. Also expectedly, more connected networks show higher improvement ratios than low connected networks. Value for burst loss ratio using ED when initial DA burst drop ratio was around 1% for the 9 nodes (8 ring plus central node) is not shown because it is very close to zero.

The burst loss ratios for the two routing algorithms for the COST 239 network topology with different loads are shown in Figure 6. Here it is visible that the change in the routing algorithm is responsible for an improvement of almost an order of magnitude when the network has 24 available data channels (traffic generation rate was kept constant to allow comparison). For 28 data channels, routing performed with the ED drops to zero.

To compare the balance of the networks for DA and for ED, following the definition in (3), the standard deviation of the values for the route matrices was measured for all the topologies. Figure 7 shows that for all tested topologies ED routing produces more balanced networks than DA routing.

The combined balance and symmetry of the networks for both routing algorithms, following the definition in (5) was also assessed. Figure 8 shows the differences between the two algorithms for the tested topologies. The two mean routes per link ($\vec{R}/\#Path$ and $\bar{R}/\#Path$, respectively) and the overall mean number of routes per link are shown. For regular topologies as four node and eight node rings, ED is clearly more symmetric than DA; on the 6 node ring both algorithms have similar performance regarding symmetry. On the other hand, for the 7 nodes and 9 nodes irregular topologies, ED is less symmetric than DA (in the case of COST 239, the difference is neglectible). The combination of these results with the results from Figure 5 and Figure 7, clearly show that the balance between all links is more important for the overall performance of the network than the link bidirectional symmetry.

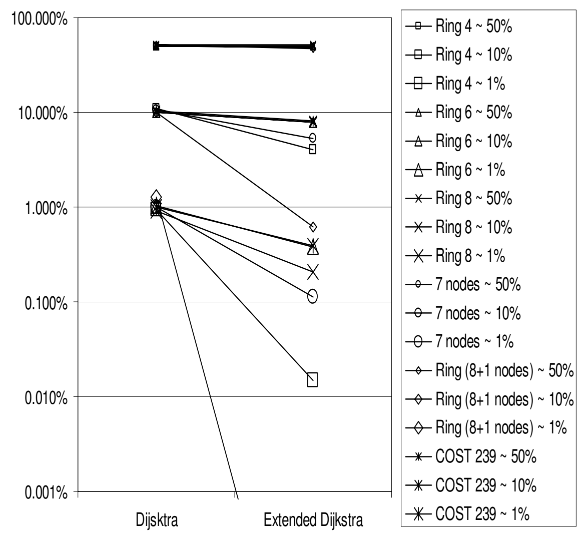

Figure 5 – Burst loss ratio vs. routing algorithm for three burst loss scenarios (around 50%, 10% and 1%) for four network ring and ring based topologies.

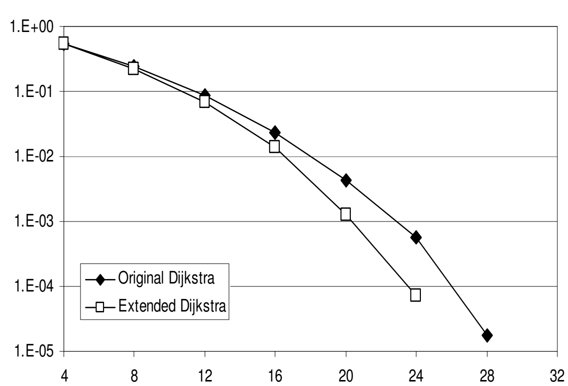

Figure 6 – OBS burst loss ratio for the COST 239 network vs. number of data channels, comparing DA and ED routing.

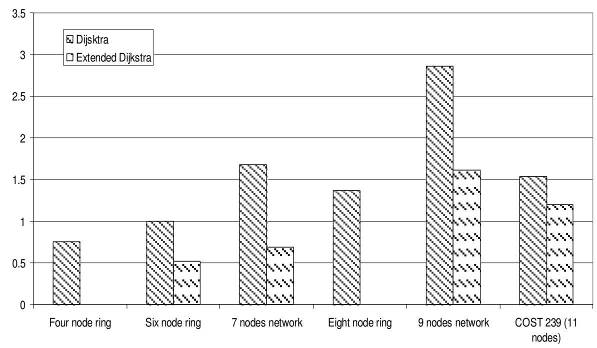

Figure 7 – Standard deviation for the number of routes for DA and ED routing, for several topologies.

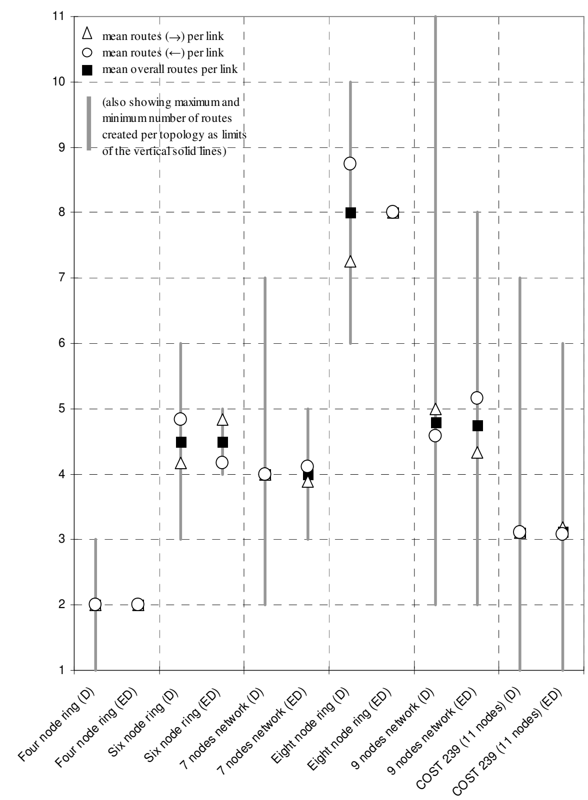

Figure 8 – Symmetry test for the DA (D) and ED routing, showing mean number of routes in each direction, and mean overall, maximum and minimum number of routes per link for several topologies.

5. Conclusions

The research motivation for this paper was the discovery of an apparent incongruence when counting the paths that were defined in a four node ring network following the *de facto* standard for network routing algorithm (Dijkstra's algorithm).

A new algorithm, termed "Extended Dijkstra", was devised and implemented in a simulator. Two new metrics – network balance and link symmetry – were

Second International Conference on Systems and Networks Communications (ICSNC 2007)
0-7695-2938-0/07 $25.00 © 2007 IEEE

also defined to measure the impact of the creation of routes on the links of the network.

As to test this assumption with a network switching paradigm, additional research was focused on the assessment of OBS networks using the two routing algorithms. It was proved that the Extended Dijkstra algorithm allowed better utilization of the network resources for all of the tested network topologies, without waiving Dijkstra's algorithm speed and simplicity.

Although routing in a real environment cannot be restricted to OSPF (an implementation of Dijkstra's algorithm), it was proved that there is still room to improve the initial Dijkstra algorithm, being the Extended Dijkstra our proposal. The Extended Dijkstra algorithm was tested through simulation, and its performance metrics, balance and symmetry where compared with the ones for the Dijkstra algorithm. Results show that for the overall network performance, balance in links is more relevant than symmetry in routes.

Furthermore, it was proved that several published results focusing on performance assessment of ring networks or networks than include rings or ones that rely on simulators that use plain Dijkstra algorithm routing, e.g. ns-2, OWns, OBSim or OIRC OBS-ns Simulator, could be over pessimistic because its static source routing shortest path only approach is not adequate when compared with still shortest path better traffic routing such as the one provided by Extended Dijkstra. In conclusion, the estimations made as to the nature and amount of the network resources necessary to achieve an acceptable level of performance on these networks, drawn from the results provided by these simulators, are over provisioned.

Shortest path routing is not the sole mechanism that defines routing in modern networks. Other approaches seek to optimize the set of paths used to route static traffic demands, not necessarily returning shortest paths routes. Nevertheless, the starting point in network routing is very often OSPF. The Extended Dijkstra algorithm presented here, still provides strictly the shortest paths, and thus may allow a faster convergence to routing table equilibrium in dynamically routed meshed networks.

Acknowledgements

Authors thank Jorge Castro from the Research Department at Nokia Siemens Portugal S.A. for very interesting discussions on this subject.

This work has been partially funded by Fundação para a Ciência e a Tecnologia, Portugal, through the grant contract SFRH/BDE/15527/2004 and through the CONDENSA Project contract POSC/EEA-CPS/60247/2004, and by Nokia Siemens Portugal S.A., Portugal.

Przemyslaw Lenkiewicz is currently with Microsoft Portugal and can be reached at v-prlenk@microsoft.com.

References

[1] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, pp. 269-271, 1959.

[2] Dawn Networking Research Labs, "The network simulator ns-2," in http://www.isi.edu/nsnam/ns/ 2002, accessed at January 10th, 2004.

[3] N. M. Bhide and K. M. Sivalingam, "Design of OWns: Optical Wavelength Division Multiplexing (WDM) Network Simulator," in *proceedings of First SPIE Optical Networking Workshop*, Dallas, TX, Jan 2000.

[4] J. J. P. C. Rodrigues, N. M. Garcia, M. M. Freire, and P. Lorenz, "Object-Oriented Modeling and Simulation of Optical Burst Switching Networks," in *proceedings of IEEE Global Telecommunications Conference (GLOBECOM '2004) - Conference Workshops, 10th IEEE Workshop on Computer-Aided Modeling Analysis and Design of Communication Links and Networks (CAMAD 2004)*, Dallas, Texas, USA, November 29th - December 3rd, 2004, pp. 288-292.

[5] R. Shenai, S. Gowda, and K. Sivalingam, "OIRC OBS-ns Simulator," in http://wine.icu.ac.kr/~obsns/index.php, 2005, accessed at 19 January 2006.

[6] C. Qiao and M. Yoo, "Optical burst switching (OBS) - A new paradigm for an optical Internet," *Journal of High Speed Networks*, vol. 8, issue 1, pp. 69-84, January 1999.

[7] J. Moy, "OSPF version 2," RFC 2328, Internet Engineering Task Force (IETF), 1998.

[8] IEEE Project 802 Working Group 802.1, "IEEE Spanning-Tree Protocol," 802.1D, IEEE, 1998.

[9] M. G. Tan and L. C. Sinclair, "Wavelgength Assignment Between the Central Nodes of the COST 239 European Optical Network," in *proceedings of The 11th UK Performance Engineering Workshop*, Liverpool, United Kingdom, pp. 235-247.

[10] R. Waldura, "Renaud Waldura's Dijkstra's Shortest Path Algorithm in Java," in http://renaud.waldura.com/doc/java/dijkstra/, 2005, accessed at 10-01-2006.