QoS Routing Algorithms under Inaccurate Routing Information for Bandwidth Constrained Applications[1]

Xavier Masip-Bruin, Sergi Sànchez-López, Josep Solé-Pareta, Jordi Domingo-Pascual
Departament d'Arquitectura de Computadors, Universitat Politècnica de Catalunya
C/Jordi Girona, 1-3 – 08034 Barcelona, Catalunya, Spain
{xmasip,sergio,pareta,jordid}@ac.upc.es

***Abstract-Multiprotocol Label Switching* and *Traffic Engineering* are proposed by the IETF to improve the network performance. Moreover, some QoS routing algorithms must be added in order to optimize the path selection process. However, in highly dynamic large networks, existing QoS routing algorithms suffer from a blocking probability, which in part is due to the existence of inaccuracy in the network state information used to select the path. This paper deals with the *BYPASS based routing* mechanism, which was introduced in a previous paper to overcome this routing inaccuracy effect in IP/MPLS scenarios. We suggest an enhancement of the BBR mechanism to optimize the bandwidth allocation by balancing the path length and the residual bandwidth, and its performance is evaluated by simulation.**

Keywords-QoS Routing, MPLS, routing inaccuracy

I. INTRODUCTION

In the present days, some modifications are being added to the initial best effort based Internet technology for allowing the network to support the current QoS traffic requirements. Best effort networks uses routing protocols such as OSPF or IS-IS, which do not take into account these QoS requirements in the path selection process.

New mechanisms for supporting QoS on the Internet, including *Traffic Engineering* (TE) [1] and *Multiprotocol Label Switching* (MPLS) [2], envisage both an optimal end-to-end path selection and a fast forwarding mechanism in the intermediate nodes. For such an environment, QoS routing protocols are being designed as improved versions of the above mentioned best effort routing protocols.

Some significant QoS Routing mechanisms are *Widest-Shortest Path* (WSP) [3], *Shortest-Widest Path* (SWP [4], *Minimum Interference Routing Algorithm* (MIRA) [5], *Profile-Based Routing* [6] and *Maximum Delay-Weighted capacity Routing Algorithm* (MDWCRA) [7]. Nevertheless, these algorithms suffer from a blocking probability, which in part is due to the existence of inaccuracy in the network state information used to select the path. QoS Routing mechanisms select the optimal route according to the routing information distributed throughout the network, contained in the link state databases. This is a critical point in the routing process that can easily lead to a wrong path (LSP in MPLS environments) selection when the routing information existing in the databases cannot perfectly represent a current picture of the network.

To keep the network state databases perfectly updated, an updating message should be triggered whenever setting up or releasing paths. In highly dynamic networks, this would require a large number of updating messages which implies a non desirable signaling overhead. To avoid this problem, usually three triggering policies are applied, namely *Threshold based policy*, *Equal class based policy* and *Exponential class based policy* [8]. The Threshold based policy consists of sending an updating message when the actual residual bandwidth differs (is lower or greater) from the last advertised residual bandwidth in a quantity defined by a threshold *tv*. The other two policies are based on a link bandwidth partitioning, in such a way that the total link capacity is divided into several classes. Being *Bw* (base class size) a fixed bandwidth value, the Equal class based policy establishes its classes according to (0,Bw), (Bw,2Bw), (2Bw,3Bw), etc, and the Exponential class based policy according to (0,Bw), (Bw,(f+1)Bw), ((f+1)Bw,(f^2+f+1)Bw), etc, where *f* is a constant value. Then, an updating message is triggered when the link capacity variation implies a change of class.

Although applying any triggering policy reduces the number of updating messages, i.e., the signaling overhead, it introduces an important problem known as routing inaccuracy problem, that is, the path selection is performed according to inaccurate network state information.

Another significant source of routing inaccuracy is due to the network state aggregation process typically implemented in hierarchical networks. A hierarchical network structure limits the number of links and nodes able to generate updating messages to make the network more scalable. This process inherently introduces a certain loss of information, since information about physical links are physical nodes (lower level) are not flooded to an upper hierarchical level.

[1] This work was partially funded by the MCyT (Spanish Ministry of Science and Technology) under contract FEDER-TIC2002-04531-C04-02 and the CIRIT (Catalan Research Council) 2001-SGR00226.

0-7803-7802-4/03/$17.00 © 2003 IEEE

In summary, if the information contained in the link state databases is not perfectly updated, the routing algorithm could select a path unable to support the path incoming request, since one or more links of that path could have less available resources than the required to set up that path. In this way, the incoming path request will be rejected in the setup process, which implies a call blocking increment. Several recent works exist in the literature addressing the routing inaccuracy effects on the path selection process [9-14].

R.Guerin and A.Orda [9] present different proposals for reducing the routing inaccuracy effects depending on the QoS constraint required by the incoming traffic. When bandwidth is required, authors suggest applying a modified version of the standard *Most Reliable Path* (MRP). When an end-to-end delay bound is required, the authors present two different approaches (rate-based and delay-based) and different solutions are generated for each model. The first approach has the advantage that the end-to-en delay bound depends on the available bandwidth on each link. The second approach has the disadvantage that tractable solutions can only be applied for relatively limited cases. Nevertheless, authors introduce a simplification based on splitting the end-to-end delay guarantee to a several minor problems that extends the cases where solutions can be applied.

In [10] D.Lorenz and A.Orda try to solve the problem of selecting an optimal path that guarantees a bounded delay by searching for the path most likely to satisfy this QoS requirement, namely the *MP Problem*. As in the solution presented in [9] for the delay-based approach, the complexity of this problem is reduced after splitting the end-to-end delay constraint in several local end-to-end delay constraints. How this partition is done is analyzed as the *Optimally partitioned MP* (OP-MP) Problem, and solutions based on using programming dynamics methods are presented.

Also searching for the most likely path, G.Apostopoulos et al. [11], propose a new routing mechanism named *Safety Based Routing* (SBR), to address the routing inaccuracy effects when computing explicit paths with bandwidth constraints. The SBR is based on computing the *safety* (S), a new link attribute incorporated to the path selection process, which represents the effects of the routing inaccuracy in the link state reliability. Two algorithms inferred from the SBR mechanism are proposed, the *shortest-safest path* and the *safest-shortest path*. The first selects the shortest path among the paths that minimize *S* and the second selects the path that minimizes *S* among the shortest paths. Obtained simulation results show a lower bandwidth blocking ratio when the *shortest-safest path* (SSP) is in use.

S.Chen and K.Nahrstedt [12] presents a mechanism, which is not based on computing the path able to support the traffic requirements with a larger probability but it is a new multipath distributed routing scheme, named *Ticket Based Probing* (TBP), to address the NP-complete delay-constraint least-cost routing problem. The TBP defines the imprecise state model by defining which information must be stored in every node, and then sends routing messages named *probes*, from the source node to the destination node to find a low cost path that fulfills the delay requirements. Results obtained by simulation show that the TBP achieves high success ratio and low-cost feasible path with minor overhead.

In [13] the *BYPASS Based Routing* (BBR) was suggested. The main difference between BBR and the rest of the above mentioned existing solutions is the routing behavior when, even applying any of these new algorithms a path that really cannot cope with the traffic bandwidth requirements is selected. This situation is differently managed in the routing mechanism proposed in this paper. In fact, unlike other algorithms that reject the incoming request, our solution tries to skip those links that impede the end-to-end path establishment by using different pre-computed paths. The BBR mechanism will be in more detail reviewed in Section II.

In this paper, we suggest an enhancement of the BBR mechanism to optimize the bandwidth allocation by balancing the path length and the residual bandwidth, and its performance is evaluated by simulation.

The remaining of this paper is organized as follows. Section II shortly describes the BBR mechanism. In Section III the enhancement of the BBR mechanism to optimize the bandwidth allocation is discussed, and in Section IV its performance is evaluated by simulation. Finally, Section V concludes the paper.

II. BBR REVIEW

Let G*(N,L,B)* describe a defined network, where *N* is the set of nodes, *L* the set of links and *B* the capacity bandwidth of the links. Suppose that a set of source-destination pairs *(s,d)* exists, named *P*, and that all the LSP incoming requests occur between elements of *P*. Let b_{req} be the bandwidth requested in an element $(s,d)\in P$. Let $G_r(N_r,L_r,B_r)$ represent the last advertised residual graph, where N_r, L_r and B_r are respectively the remaining nodes, links and residual bandwidths at the time of path setup. The main steps in the BBR performance are:

1) ***Obstruct-Sensitive Links***. A new parameter is introduced in the path selection process to represent the routing inaccuracy. This parameter is translated into a new class of link. In this way, an *Obstruct-Sensitive Link* (OSL) is a link that potentially is unable to support the traffic requirements in accordance with a certain link definition. Specifically, a link is OSL when b_{req} belongs to the updating range generated by the last advertised bandwidth value in this link in accordance with any triggering policy. Formally:

Rule1: Let L^{os} be the set of obstruct-sensitive links. Let $l^{os}\in L^{os}$ be a link of the residual graph G_r. A link l_i is defined as OSL depending on the triggering policy in use. So:

- *Threshold triggering policy*: Let b^i_r be the last advertised bandwidth for a link l_i. This link l_i is defined as OSL, that is l^{os}_i if

$$l_i=l^{os}_i \mid l^{os}_i \in L^{os} \Leftrightarrow b_{req} \in (b^i_r(1-tv), b^i_r(1+tv)]. \quad (1)$$

- *Exponential-class triggering policy*: Let B_{l_j} and B_{u_j} be the minimum and the maximum bandwidth values allocated to class *j*. A link l_i is defined as OSL, that is l^{os}_i if

$$l_i = l^{os}_i \mid l^{os}_i \in L^{os} \Leftrightarrow b_{req} \in (B_{l\_j}, B_{u\_j}]. \quad (2)$$

2) ***Working Path Selection***. Once the number of OSLs is known, three parameters can be used to select the optimal path, that is, the number of hops, the number of OSLs and the residual bandwidth. In [9] two different routing algorithms are inferred from the BBR mechanism.

- *Shortest-Obstruct-Sensitive Path (SOSP):* the shortest path among the paths with the minimum number of OSLs is selected
- *Obstruct-Sensitive-Shortest Path (OSSP):* the path with the minimum number of OSLs among the shortest paths is selected.

In both algorithms if more than one feasible path exists, one of them is randomly selected.

3) ***BYPASS Paths computation****:* Once the route is selected, the BBR mechanism computes an alternative path that bypasses each OSL. These new paths are named *bypass-paths* and are used when an OSL cannot cope with the traffic requirements. If more than one *bypass-path* exists, the route that minimizes the number of OSLs is chosen (other options, such as either to minimize the number of hops or to maximize the residual available bandwidth are left for further studies). In order to reduce the number of used *bypass-paths*, the edges of these *bypass-paths* should be carefully chosen.

Rule 2*:* Let L^{os} be the set of OSLs. Let i_j and e_j be the edge nodes of a link $l^{os}_j \in L^{os}$. Let l_k be one link adjacent to l^{os}_j. The edge nodes of the *bypass-paths* to be computed are

$$(i_j, e_j) \quad \Leftrightarrow \quad l_k \notin L^{os}, \quad (3a)$$

or

$$(i_j, e_k), (i_k, e_k) \quad \Leftrightarrow \quad l_k = l^{os}_k \in L^{os}. \quad (3b)$$

In this way, there is only one *bypass-path* that bypasses all the adjacent OSLs.

4) ***BYPASS Paths usage****:* Assuming that the routing information updating process is not instantaneous, links that cannot cope with the traffic requirements are only known at the time of path setup. In this way, when a node detects a link *i* with $b^i_r < b_{req}$ it sends the setup message along the *bypass* route which bypasses this link. In order to perform this, *bypass* routes are explicitly sent along with the end-to-end route in the path setup message. Some information about the mechanism used to set up *bypass-paths* can be found in [14]. In order to minimize the setup message size, *bypass-paths* are removed from the setup message when passing the link that bypass. In [13] the SOSP and the OSSP algorithms are evaluated in comparison with other QoS routing algorithms. Best behavior is obtained when the SOSP is the algorithm in use.

III. BBR Mechanism Enhancement

Being the SOSP the best BBR algorithm, a simpler approach for selecting routes in accordance with certain bandwidth constraint is based on including the residual bandwidth in the SOSP algorithm. In this way the SOSP is only modified when the final selection includes more than one path. In this case, the route is not randomly selected but the widest is chosen. We call this algorithm *Widest-Shortest-Obstruct-Sensitive-Path* (WSOSP).

However, the WSOSP is just a simple approach where the number of hops has more weight than the bandwidth capacity in the route selection process. Therefore, in order to balance the path selection process, avoiding those paths that are both widest but too long and shortest but too narrow, a new algorithm is suggested. We call this algorithm *Balanced-Obstruct-Sensitive-Path* (BOSP). The BOSP algorithm is based on extending the shortest path algorithm with the number of OSLs, but unlike previous algorithms based on the BBR mechanism already described in this paper, a new parameter is added to each feasible route between source and destination. This parameter, named F_p, represents the relation between the maximum residual bandwidth and the number of hops along a path *p*, according to

$$F_p = n\left[\max\left(\frac{1}{b^i_r}\right)\right] \qquad i=1..n \quad (4)$$

where *n* is the number of hops and b^i_r is the available residual bandwidth on link *i* in the path *p*. In this way, by using F_p as the cost of each link, the network load and the network occupancy is balanced in the path selection process.

Two main factors contribute to the BOSP complexity. On one hand selecting the shortest path according to *w(l)* by using a binary-heap implementation of the Dijkstra's algorithm introduces a cost of $O(L \cdot \log N)$. On the other hand, assuming that the *bypass-paths* cannot include a network element which is also included in the working path, G(V, E) represents the reduced network where $V < N$ and $E < L$. Therefore, a factor of $O(E \cdot \log V)$ is added because of the *bypass-path* computation. Let M be the variable number of *bypass-paths* that can be computed along a working path. Now, the cost is $O(M(E \cdot \log V)$. Being $\hat{M}$ an upper bound of the number of computed *bypass-paths* along a working path, the BOSP complexity is in order of $O(L \cdot \log N) + O(\hat{M}(E \cdot \log V)$, i.e. effectively to $O(L \cdot \log N) + O(E \cdot \log V)$. Finally, considering that *bypath-paths* are computed based on a reduced network, the complexity is $O(L \cdot \log N)$.

The box enclosed in next page shortly describes the BOSP algorithm. Steps 7 and 8 should be in detail explained. Once a link is defined as OSL, the BBR computes possible routes that bypass this link. If more than one exists, the route that minimizes the number of OSLs is chosen. However, different decision parameters could be used to select the *bypass-path*s, such as to minimize the number of hops or to maximize the residual available bandwidth (left for further studies).

BALANCED OBSTRUCT SENSITIVE PATH ALGORITHM (BOSP)

INPUT:
The input graph $G(N_r,L_r,B_r)$. The *LSP* request is between a source-destination pair *(s,d)* and the bandwidth requirement is b_{req}.

OUTPUT:
An optimized and balanced route from *s* to *d* with enough *bypass-paths* to bypass the routing inaccuracy effects in the *obstruct-sensitive links*.

ALGORITHM:
1. Mark those links that are defined as *OSL* according to Rule 1
2. Compute the weight of a link *l* as
$$w(l)=1 \Leftrightarrow l \in L^{os}, \qquad w(l)=0 \Leftrightarrow l \notin L^{os}$$
3. Apply Dijkstra's algorithm to select the path that minimizes the number of *OSLs* by using *w(l)* as the cost of each link
4. If more than one exists compute the cost F_p of each path
$$F_p = n\left[\max\left(\frac{1}{b_r^i}\right)\right] \qquad i=1..n$$
5. Select the path that minimizes F_p.
6. Determine the edge nodes pair *(i,e)* of the *OSLs* existent in the selected path
7. Compute the *bypass-paths* for each element *(i,e)* according to Rule 2.
8. Decide which *bypass-paths* must be used in accordance with real available resources in the path setup time.
9. Route the traffic from *s* to *d* along the setup path

A. Example for illustrating the enhanced BBR behavior

The topology shown in Fig. 1 is used to test the BBR performance. In order to perform this, an LSP incoming request demanding b_{req} of 4 units of bandwidth between LSR0-LSR7 is supposed. Moreover, the Exponential class based policy, with *f=2* and *Bw=1* [as used in 11] is the triggering policy used.

Table I shows all the different routes between *LSR0* and *LSR7*. We define *ID* as a path identifier; *H* represents the number of hops along the path; b_r^{min} is the minimum residual bandwidth along the path; *N_OSL* is the number of *obstruct-sensitive links*, F_p is the cost and *Algorithm* represents the algorithms that select each path. This paper introduces two new routing algorithms based on the BBR mechanism. In this way, the WSOSP and the BOSP select the paths identified by *b* and *a* respectively. Once the path has been selected, the BBR mechanism computes a *bypass* path for each OSL. When

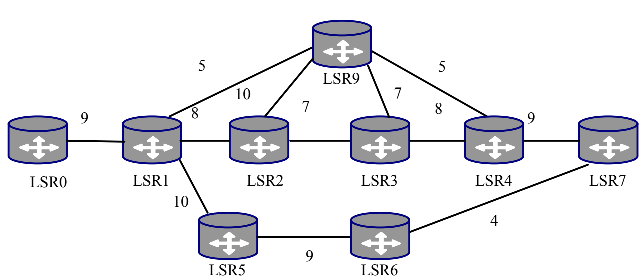

Figure 1. Network topology used in the illustrative example.

TABLE I POSSIBLE PATHS BETWEEN LSR0 AND LSR7

ID	PATH (LSR)	H	b_r^{min}	N_OSL	F_p	Algorithm
a	0-1-2-3-4-7	5	7	1	0.71	BOSP
b	0-1-5-6-7	4	4	1	1	SOSP,WSOSP
c	0-1-2-9-3-4-7	6	7	1	0.85	
d	0-1-9-3-4-7	5	4	2	1.25	
e	0-1-9-2-3-4-7	6	4	2	1.5	
f	0-1-2-9-4-7	5	4	2	1.25	
g	0-1-9-4-7	4	5	2	0.8	WSP

TABLE II POSSIBLE BYPASS PATHS

ID	L^{os}	Edge LSRs	Bypass_path	H	b_r^{min}	F_p	N_OSL
a	2-3	2-3	2-9-3	2	7	0.28	1
b	6-7	6-7	--	-	--	--	--

the BOSP routing algorithm is used, *a* is the route selected and one OSL exists, as shown in Table I.

Then, the BBR mechanism computes the edge nodes of said OSL, that is {LSR2-LSR3}. In order to compute the *bypass-path*, it is possible to use any of the parameters shown in Table II (*H*, b_r^{min}, F and *N_OSL*). In this paper, the *N_OSL* is the parameter used to compute the *bypass-path*s. Therefore, {LSR2-LSR9-LSR3} is the *bypass-path* selected to bypass the OSL. When the WSOSP is implemented the edge nodes to bypass are {LSR6-LSR7}. However, in this case it is not possible to find a path that bypasses this link in the network topology. One approach to address this case could be to find different edges for the OSL (neighbor discovery process), which is currently under study.

IV. PERFORMANCE EVALUATION

To evaluate the performance of the proposed BBR enhancement the ns2 simulator, extended with MPLS features and BBR requirements, has been used. A set of simulations has been carried out to test the WSOSP and BOSP algorithms suggested here, in comparison with the already existing WSP, SSP, SOSP and OSSP algorithms.

A. Simulation scenario

The simulations have been carried out over the network topology shown in Fig. 2. Two link capacities are used, 622 Mbps represented by a light line and 2.4 Gbps represented by a dark line. Every simulation requests 1700 LSPs which arrive following a Poisson distribution where the requested

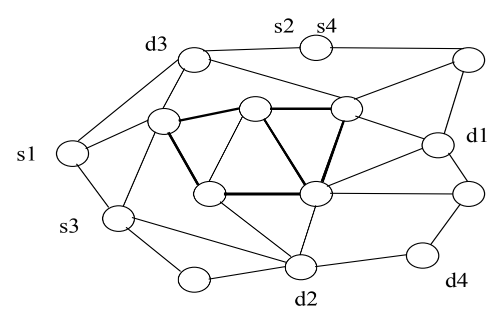

Figure 2. Network topology used in simulations

bandwidth is uniformly distributed between 1 Mb and 5 Mb and the holding time is randomly distributed with a mean of 120 sec. The triggering policies used in this paper are the Threshold and the Exponential class (with $f = 2$). The results presented in this paper have been obtained after repeating 300 sec. of simulation 10 times.

The parameters used to measure the algorithm behavior are the routing inaccuracy and the bandwidth blocking ratio.

Routing Inaccuracy: This parameter represents the total number of paths that has been incorrectly selected. It is defined as

$$\text{routing inaccuracy} = \frac{\text{number of paths incorrectly selected}}{\text{total number of requests}} \quad (5)$$

A path can be incorrectly selected because of two factors. One of them is the LSP rejection when actually there is a route with enough resources to support that demand. The other factor is the blocking of an LSP that initially was routed by the ingress node, but which due to the insufficient bandwidth in an intermediate link it is rejected

Bandwidth Blocking Ratio: In order to obtain the number of rejected LSP demands, we use the bandwidth blocking ratio as

$$\text{bandwidth blocking ratio} = \frac{\sum_{i \in rej\_LSP} \text{bandwidth}_i}{\sum_{i \in tot\_LSP} \text{bandwidth}_i}, \quad (6)$$

where *rej_LSP* are the set of blocked demands and *tot_LSP* are the set of total LSP requested.

B. Simulation results

The aim of BBR is to improve the global network performance in terms of optimizing the path selection. Fig. 3 and Fig. 4 show that this objective has been achieved.

On one hand, Fig. 3 shows the bandwidth blocking ratio tested for both Threshold and Exponential class triggering policies. As it was pursued, the routing algorithms inferred from the BBR mechanism reduce the bandwidth blocking ratio. Note that the best performance is obtained using the BOSP, whereas WSOSP exhibits similar results to SOSP, and in both cases, BOSP and WSOSP perform better than WSP. In the worst conditions (the threshold *tv* of the triggering policy is 90%), the bandwidth blocking ratio obtained by the BOSP (11%) substantially improves those obtained by the WSOSP (13.5%), the SOSP (13.3%) and the SSP (19.3%). Recall that by increasing the threshold value the number of updating messages flooded throughout the network is reduced.

On the other hand, Fig. 4 shows the routing inaccuracy behavior. Theoretically, using a routing algorithm inferred from the BBR mechanism reduces the number of non-optimally selected paths. In fact, Fig. 4 shows that BOSP exhibits better results for both triggering policies. The number of paths incorrectly selected is extremely low, even for large values of the threshold and the base class size *Bw*. Again, in the worst case (*tv* = 90%), the number of paths incorrectly selected by the WSP (10.5%), or SSP (5%) is reduced when applying the BOSP (2%).

Finally, Fig. 5 aims to illustrate the cost of using the BBR mechanism. Here only the routing algorithms inferred from the BBR mechanism are evaluated. Fig. 5 shows that the number of computed *bypass-paths* grows when both the threshold and the base class size values increase, which seems logic, since the number of OSL grows when the amount of flooding messages decrease.

It is important to observe that when the BOSP is applied, not only the bandwidth blocking ratio and the routing inaccuracy decrease but rather the cost decreases as well. This is due to the optimization achieved in the routing process.

V. CONCLUSIONS

This paper describes and evaluates an improvement of the BYPASS Based Routing mechanism, introduced in [13] in order to optimize the bandwidth allocation. Basically, the BBR goal was to bypass those links defined as *obstruct-sensitive links* (OSL), which could be unable to support the traffic attributes required by the incoming LSP.

A new BBR based algorithm named *Balanced-Obstruct-*

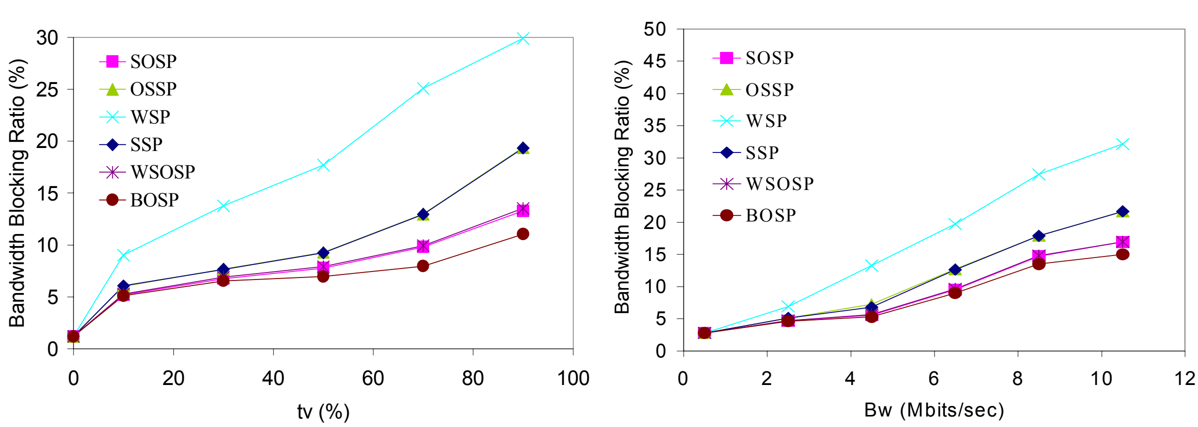

Figure 3. Bandwidth Blocking ratio for the Threshold and the Exponential class triggering policies

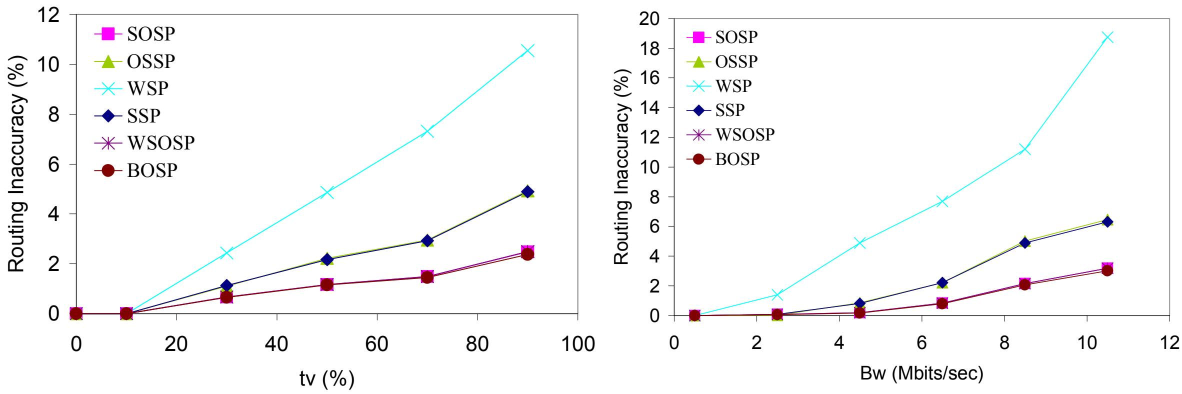

Figure 4. Routing Inaccuracy for the Threshold and the Exponential class triggering policies

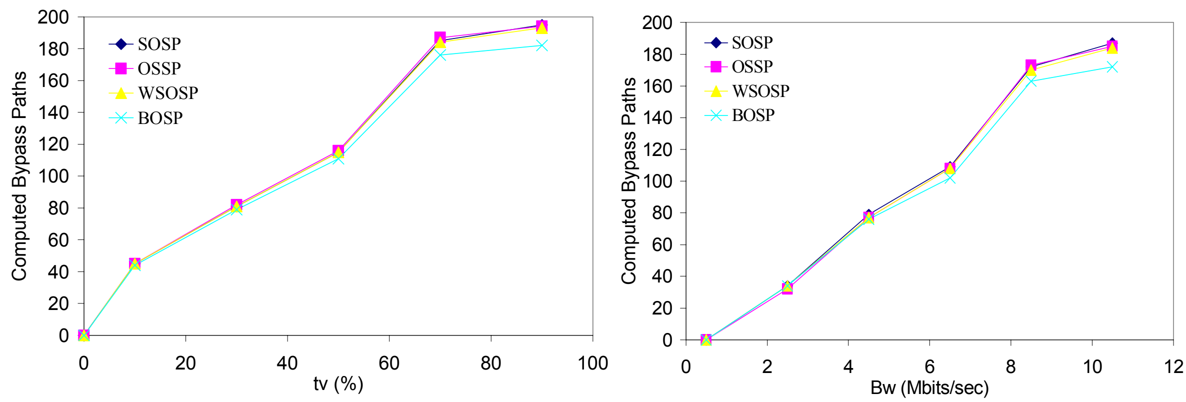

Figure 5. Computed *bypass-paths* for the Threshold and the Exponential class triggering policies

Sensitive (BOSP) is suggested. The main characteristic of the BOSP algorithm is that the path selection is made according to both minimize the number of OSLs and combine the number of hops and the minimum residual available bandwidth. This is done by modifying the cost value of the path.

Simulation results show that the BOSP algorithm substantially improves the performance of both the basic BYPASS Based Routing mechanism and the Safety Based Routing mechanism.

REFERENCES

[1] D.O.Awduche, "Requirements for Traffic Engineering over MPLS", *RFC2702*, September 1999.

[2] E.Rosen, A.Viswanathan, R.Callon, "Multiprotocol Label Switching architecture", *RFC3031*, January 2001.

[3] R.Guerin, A.Orda, D.Williams, "QoS routing mechanisms and OSPF extensions", Proc. 2nd Global Internet Miniconference (joint with Globecom'97), Phoenix, AZ, November 1997.

[4] Z.Wang, J.Crowcroft, "Quality-of-Service routing for supporting multimedia applications", IEEE JSAC, 14(7): 1288-1234, Sept. 1996.

[5] M.Kodialam, T.V.Lakshman, "Minimum interference routing with applications to MPLS Traffic Engineering". Proc. IEEE Infocom 2000.

[6] S.Suri, M.Waldvogel, P.R.Warkhede, "Profile-Based Routing: a new framework for MPLS Traffic Engineering".Proc. QofIS 2001.

[7] Y.Yang, J.K.Muppala, S.T.Chanson, "Quality of service routing algorithms for bandwidth-delay constrained applications", Proc. 9th IEEE ICNP 2001, Riverside, Nov. 11-14, 2001.

[8] G.Apostolopoulos, R.Guerin, S.Kamat, S.Tripathi, "Quality of Service based routing: a performance perspective". Proc SIGCOMM 1998.

[9] R.A.Guerin, A.Orda, "QoS routing in networks with inaccurate information: theory and algorithms", IEEE/ACM Transactions on Networking, Vol.7, nº.3, pp. 350-364, June 1999.

[10] D.H.Lorenz, A.Orda, "QoS routing in networks with uncertain parameters", IEEE/ACM Transactions on Networking, Vol.6, nº.6, pp.768-778, December 1998.

[11] G.Apostolopoulos, R.Guerin, S.Kamat, S.K.Tripathi, "Improving QoS routing performance under inaccurate link state information", Proc. ITC'16, June 1999.

[12] S.Chen, K.Nahrstedt, "Distributed QoS routing with imprecise state information", Proc.7th IEEE International Conference of Computer, Communications and Nettworks, 1998.

[13] X.Masip, S.Sánchez, J.Solé, J.Domingo, "A QoS routing mechanism for reducing the routing inaccuracy effects", Proc. QoS-IP 2003.

[14] X.Masip, S.Sánchez, J.Solé, J.Domingo, "An alternative path fast rerouting in MPLS", Proc. ISCIS XV 2000.