Analysis of Dynamic QoS Routing Algorithms for MPLS Networks

Antonio Capone, Luigi Fratta, Fabio Martignon
Dipartimento Elettronica e Informazione
Politecnico di Milano
P.zza L. Da Vinci 32, 20133 Milano
Email: {capone, fratta, martignon}@elet.polimi.it

***Abstract*— Finding a path in the network for each traffic flow able to guarantee some quality parameters such as bandwidth and delay is the task of QoS routing algorithms developed for new IP networks based on label forwarding techniques as Multiprotocol Label Switching (MPLS). In this paper we focus on Dynamic QoS Routing, i.e. the routing of bandwidth guaranteed flows in a dynamic scenario where new connection requests arrive at the network edge nodes. When more than one path satisfying the bandwidth demand exists, the selection of the path aims at minimizing the blocking probability of future requests. We propose two novel mathematical programming models that assume the knowledge of arrival times and durations of connection requests, and provide theoretical bounds to the performance achievable by on-line routing algorithms. We compare to such bounds the performance of the Min-Hop (MH) algorithm, the Minimum Interference Routing Algorithm (MIRA) and the recently proposed Virtual Flow Deviation (VFD) algorithm. We show that the blocking probability of this new algorithm, in most scenarios, is quite close to the bound.**

I. INTRODUCTION

Recently, substantial effort has been spent to evolve conventional IP routing architecture and protocols by providing them with additional functionalities using the Multi Protocol Label Switching (MPLS) [1]. One of the key aspects of MPLS is the addition of a new connectivity abstraction: explicitly routed point-to-point paths can be established using label based forwarding mechanisms, thus allowing per flow path selection and Quality of Service (QoS) parameters to be taken into account by the routing algorithm. The QoS requirement of a connection can be given as a set of constraints. Such constraints can be expressed, for instance, as bandwidth constraints specifying that the path selected for the connection of the requesting user has sufficient bandwidth to meet the connection requirement (see [2], [3] for a survey of QoS routing optimization problems).

The task of the routing algorithm depends on the service scenario and the information available. In a service scenario with permanent connections, the routing decision point usually knows all connection requests and must select a path for each of them in order to satisfy QoS requirements (static QoS routing). In contrast, in a dynamic scenario new connection requests arrive at the network edge nodes and the routing decision point must select a path at the moment when the request arrives and cannot rearrange the paths of previously routed flows (dynamic QoS routing). Of course, in this case the routing decision point does not know the arrival times and the durations of future connection requests.

In this paper we focus on dynamic QoS routing with bandwidth constraints. When a new connection arrives, the routing algorithm must select a path with residual capacity at least equal to the bandwidth request. If more than one path satisfying the QoS request exists, instead of considering additional constraints or optimization objectives (e.g. minimum delay), we aim at achieving global efficiency in resource utilization minimizing the blocking probability of new connections.

Some algorithms have been proposed in the literature for the problem of dynamic routing of bandwidth guaranteed flows [4], [5], [6], [7]. Here we propose two novel mathematical models that provide a theoretical bound to the performance achievable by on-line QoS routing algorithms.

The first one is an ILP (Integer Linear Programming) model and is based on an extension of the maximum multi-commodity flow problem [8]. It provides an optimal routing configuration capable of accommodating the traffic offered to the network. The model assumes that the information on connection arrival times and durations is available and aims at minimizing the number of rejected connections. Accepted connections are provided a single path which is kept for the whole connection lifetime (no re-routing is allowed).

The ILP model provides a very tight lower bound to the connection rejection rate as a function of the traffic offered to the network. However, due to the complexity of its formulation, the resolution of this model often requires a long computing time and a high memory occupation even with state of the art tools [9].

The second model is based on the extended Erlang formula and on the minimum multi-commodity cut problem [10] (Min-Cut model). Also this model provides a lower bound to the connection rejection rate, but this bound can be not so tight for some network scenarios. However, the memory occupation and computing time of this model are considerably lower than those required by the first model, thus allowing its application to larger network topologies.

We provide numerical results in a set of interesting network scenarios and compare the bounds obtained with the ILP and Min-Cut models with the blocking probability achieved by on-line routing algorithms.

The paper is structured as follows. In Section II we illustrate the ILP model and discuss the problem of setting the objective function parameters, while in Section III we present the Min-Cut model. In Section IV we discuss and compare the performance of on-line algorithms to the theoretical bounds achieved by the mathematical models, under a variety of network scenarios. Finally, Section V concludes the paper.

II. Mathematical Model based on the Maximum Multi-Commodity Flow Problem

Following the notation used in [8], let $G = (N, A)$ be a directed graph, defined by a set of nodes, N, and a set of arcs, A. Every arc $(i, j) \in A$ is characterized by a capacity C_{ij} that represents the maximum flow that can be shipped over that arc in every time slot.

Let us suppose that a set of connections that request guaranteed bandwidth $K = \{1, ..., N_c\}$ is offered to the network. Each of these connections is represented by the notation (S_k, T_k, d_k), where S_k, T_k and d_k represent the connection's source node, destination node and required bandwidth, respectively, with $k \in K$.

Differently from traditional formulations of multi-commodity flow problems [8], our mathematical model takes into account also the information about the arrival time of each connection as well as its duration. More precisely, the $k-th$ connection is characterized not only by the notation previously introduced, but also by its arrival time t^k_{arr} and its duration I^k. The time axis is slotted based on the beginning and ending time of the N_c connections. Note that it is sufficient to use at most $2N_c - 1$ time slots to individuate completely the time location of all the connections. Let $T = \{1, ..., 2N_c - 1\}$ be the set of all such time slots.

To introduce our problem formulation, let us consider the graph $G' = (N', V')$ that is obtained from the original graph $G = (N, V)$ with the following extensions: for each of the N_c connections we add a source node S^k_N and a destination node T^k_N; the source node S^k_N is then connected to the original ingress node of the $k-th$ connection, S_k, using the directed infinite-capacity arc A^k_S. Similarly, the original egress node of the $k-th$ connection, T_k, is connected to the node T^k_N with the directed infinite-capacity arc A^k_T.

Let $SN = \{S^k_N, k \in K\}$ and $TN = \{T^k_N, k \in K\}$ be, respectively, the sets that contain the N_c source nodes and the N_c destination nodes newly added in the extended graph G'. Finally, let $A_{SN} = \{A^k_S, k \in K\}$ and $A_{ND} = \{A^k_T, k \in K\}$ be, respectively, the sets that contain the N_c infinite capacity arcs that connect (for each commodity) the newly added source and destination nodes to the corresponding ingress/egress nodes of the original graph G. Consequently, the extended graph $G' = (N', V')$ is obtained from the graph $G = (N, V)$ by defining $N' = N \cup SN \cup TN$ and $V' = V \cup A_{SN} \cup A_{ND}$.

A. Integer Programming Formulation

Having introduced the above definitions and notation, we can now illustrate our problem formulation. Let us define the following decision variables:

$$x^k_{ijt} = \begin{cases} 1 & \text{if connection } k \text{ is routed on arc } (i,j) \text{ in time slot } t \\ 0 & \text{otherwise} \end{cases}$$

for $(i, j) \in A'$, $k \in K$ and $t \in I^k$. Furthermore we impose that $x^h_{ijt} = 0, \forall t \in T$, when $h \neq k$ and either $(i, j) = S^k_N$ or $(i, j) = T^k_N$.

As our goal is to minimize the connection rejection rate, we can equivalently maximize the number of connections accepted by the network. The problem can be thus formulated in the following way:

$$Maximize \sum_{k \in K} \sum_{(i,j) \in A_{SN}} b_k x^k_{ijt^k_{arr}} \quad (1)$$

$$s.t. \sum_{k \in K} d_k \cdot x^k_{ijt} \leq C_{ij} \forall t \in T, \forall (i, j) \in A \quad (2)$$

$$\sum_{(j,l) \in A'} x^k_{jlt} - \sum_{(i,j) \in A'} x^k_{ijt} = \begin{cases} 1 & \text{if } j \in SN \\ 0 & \text{if } j \in N \\ -1 & \text{if } j \in TN \end{cases} \quad (3)$$

$$\forall k \in K, j \in N', t \in T$$

$$x^k_{ijt} = x^k_{ijt^k_{arr}} \forall k \in K, \forall t \in I^k \quad (4)$$

$$x^k_{ijt} \in \{0, 1\} \quad (5)$$

The objective function (1) is a weighted sum of the connections accepted in the network, where b_k represents the benefit that the network operator has to accept the $k-th$ connection. A discussion on the proper choice of b_k values is proposed in the following Section.

Constraint (2) ensures that, in every time slot, the total flow due to all the connections that pass through each arc (i, j) does not exceed the arc capacity, C_{ij}.

Constraint (3) represents the mass balance equations expressed for each node belonging to the extended graph G', in every time slot $t \in T$. Note that this constraint defines a path for each connection between its source and destination nodes.

Constraint (4) imposes that, once accepted in the network, the connections cannot be aborted or rerouted on different paths for the entire connections lifetime.

Finally, imposing the integrality of the decision variables in (5) makes sure that every connection cannot be split over multiple paths; on the contrary, each connection is routed on a single path between its source and its destination node.

B. Choice of the Objective Function

On-line QoS routing algorithms do not reject an incoming connection that requests d_k bandwidth units if there exists at least one path in the network that has a residual available bandwidth greater than or equal to d_k.

On the contrary, if b_k in the objective function (1) is set to a constant value (e.g. $b_k = 1$), the mathematical model proposed above can reject such connection to accept more connections that temporally follow it. This can happen especially with connections that have larger bandwidth requirements and long duration. These connections are often rejected to admit more connections that require less network resources, especially when the network is working near its saturation condition.

To illustrate this problem let us consider a very simple network scenario in which a single source/destination pair offers connections over a monodirectional single link with capacity equal to 20 bandwidth units. Each connection requires a bandwidth uniformly distributed between 1 and 3 units, and the connections lifetime is exponentially distributed with average equal to 15s.

Fig. 1 compares the rejection probability achieved by on-line routing algorithms to that achieved by the mathematical model with $b_k = 1$. Note that in this network configuration all the routing algorithms achieve the same rejection rate as there is only one path between the source and the destination nodes.

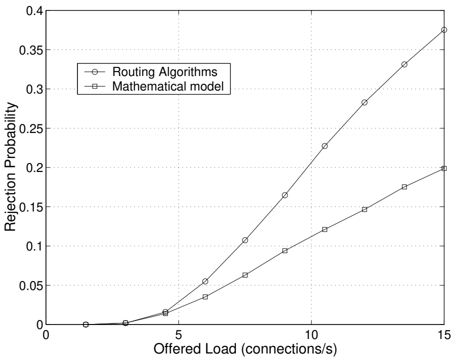


Fig. 1. Connection rejection probability versus the average total load offered to the a single link with $b_k = 1$

The mathematical model, on the contrary, provides a lower rejection rate by selectively rejecting connections having higher bandwidth requests and duration. This behavior is more evident when the offered load increases, as shown again in Fig. 1.

As we want to model as close as possible the behavior of on-line routing algorithms, we propose the following expedient in the choice of the parameter b_k of the objective function:

- the N_c connections are numbered from 1 to N_c according to their arrival time to the network. Connection 1 is hence the first connection offered to the network and N_c is the last.
- the weight b_k is set equal to 2^{N_c-k}.

With this choice, if we consider the $n-th$ connection, the benefit that the network operator has to accept that connection, equal to 2^{N_c-n}, is always greater than the benefit that the operator would have by rejecting such call even to make room for all the connections that temporally follow it, as we have:

$$2^{N_c-n} > \sum_{k=n+1}^{N_c} 2^{N_c-k} \qquad (6)$$

Inequality (6) simply follows from the observation that the summation in (6) is equal to $2^{N_c-n} - 1$. With this choice of the weight b_k the performance achieved by the mathematical model practically overlaps those achieved by on-line routing algorithms already shown in Fig. 1 in the single link scenario. Hence, this choice allows the mathematical formulation to model more closely the behavior of real on-line routing algorithms.

III. Mathematical Model based on the Extended Erlang Formula and the Minimum Multi-Commodity Cut Problem

In this Section we propose a second mathematical model that allows to determine a lower bound to the connection rejection rate. The resolution of this model has computing times and memory occupation considerably lower than the one illustrated in the previous Section. However, in some network scenarios, the bound provided by this model can be not so tight, but quite lower than the one really achievable with on-line routing algorithms.

Let us consider a directed graph $G = (N, A)$ defined by a set of nodes, N, and a set of arcs, A. Every arc $(i, j) \in A$ is characterized by a capacity C_{ij} that represents the maximum flow that can be shipped over that arc. It is further assigned a set of source/destination pairs $K = \{1, ..., N_s\}$, indicated by S_i and T_i, respectively, $i \in K$. Each source S_i wants to ship the amount of flow αf_i toward its destination T_i, and each flow can be split over multiple paths. The problem is to find the maximum α, indicated with α^*, such that all the flow quantities $\alpha^* f_i$, $i \in K$, can be shipped in the network.

Evidently this problem can be solved in polynomial time with linear programming techniques, thus allowing us to determine the maximum multi-commodity flow, indicated with F_{max}, that can traverse the network, $F_{max} = \sum_{i \in K} \alpha^* f_i$. Note that F_{max} represents a lower bound to the capacity of the minimum multi-commodity cut, as discussed in [10].

A. Model Formulation

Having introduced the above notations, we can now illustrate our problem formulation, that is divided in two steps.

In the first step we determine the maximum multi-commodity flow, F_{max}, that can be shipped through the network, with the expression $F_{max} = \sum_{i \in K} \alpha^* f_i$.

In the second step we apply the extended Erlang formula with F_{max} servers to compute the connections rejection probability for the given network scenario. In the following we briefly review the extended Erlang formula.

B. Extended Erlang Formula

Let us consider a network system with C servers, to which N different traffic classes are offered. The connections belonging to the class i request d_i bandwidth units. The connections arrival process is a Poisson process with average equal to λ_i, while the connections duration is distributed according to a generic distribution $f_{\Theta_i}(\theta_i)$, not necessarily negative exponential. Let $\Lambda = \sum_{i=1}^{N} \lambda_i$ be the total load offered to the network.

The state of this system is defined by the number of connections that occupy the servers. If n_i is the number of such connections belonging to the class i, the set of all the possible states is expressed as $S = \{(n_1, ..., n_N)|X \leq C\}$, with X indicating the total occupation of all the servers, calculated as $X = \sum_{i=1}^{N} n_i d_i$.

If we indicate with $A_i = \lambda_i E[\Theta_i]$ the traffic offered to the network by each class, the steady state probability of each state is simply given by the extended Erlang formula:

$$\pi(n_1, ..., n_N) = \frac{1}{G} \prod_{i=1}^{N} \frac{A_i^{n_i}}{n_i!} \tag{7}$$

where G is the normalization constant that ensures that the π sum to 1.

Using the steady state probability calculated with equation (7) we can calculate the loss probability of generic class i, Π_i, as follows:

$$\Pi_i = \sum_{(n_1, ..., n_N) \in B_i} \pi(n_1, ..., n_N) \tag{8}$$

in which B_i is the set of the blocking states for the class i, defined as $B_i = \{(n_1, ..., n_N)|C - d_i < X \leq C\}$. The overall connection rejection probability (p_{rej}) is then given by:

$$p_{rej} = \sum_{i=1}^{N} \frac{\lambda_i \Pi_i}{\Lambda} \tag{9}$$

If we substitute C with the maximum multi-commodity flow value F_{max} in all the above expressions, we can compute the connection rejection rate using equation (9).

IV. Numerical Results

In this section we compare the performance of the Virtual Flow Deviation algorithm with that of the Min-Hop Algorithm and MIRA referring to different network scenarios in order to cover a wide range of possible environments. The performance function we consider is the percentage of rejected calls versus the average total load offered to the network.

We also compare the performance achieved by these routing algorithms with the theoretical bounds provided by the mathematical models that we have presented in the previous Sections.

To perform this evaluation we have considered three different network scenarios. The first scenario is illustrated in Fig. 2. In this network the links are unidirectional with capacity equal to 24 bandwidth units. The network traffic, offered through the source nodes S_1, S_2 and S_3, is unbalanced as the traffic offered by sources S_2 and S_3 is four times that offered by S_1. Each connection requires a 1 bandwidth unit. The lifetime of the connections is assumed to be exponentially distributed with average equal to 15s.

In this simple topology only one path is available to route connections between S_1-T_1 and S_3-T_3, while connections S_2-T_2 can choose between two different paths.

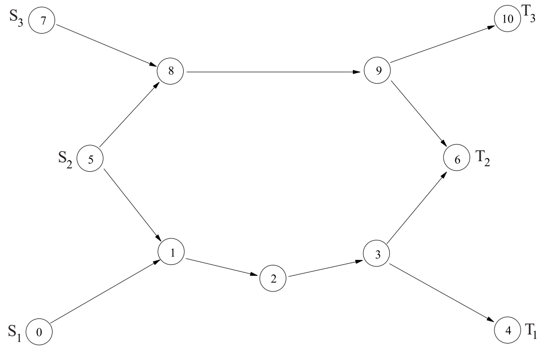


Fig. 2. Network topology with unbalanced offered load: the source/destination pairs S_2-T_2 and S_3-T_3 offer to the network a traffic load which is four times higher than that offered by the pair S_1-T_1

The maximum multi-commodity flow that can be shipped from the source to the destination nodes, F_{max}, calculated as shown in the previous Section, is equal to 48 bandwidth units.

This case shows the main limitation of MIRA that does not consider the information about the total load offered to the network. Since the links (1,2),(2,3) and (8,9) are critical for S_2-T_2, the route selected by MIRA follows the path with the minimum number of critical links (5-8-9-6 in the example). Unfortunately this interferes with the path (7-8-9-10) that carries the high load of S_3-T_3. This choice penalizes the performance as it is shown in Fig. 3.

Fig. 3 shows the connection rejection rate achieved by the on-line routing algorithms (VFD, Min Hop and MIRA) as well as that achieved by the two mathematical models presented in the previous Sections: the ILP model and the Min-Cut model.

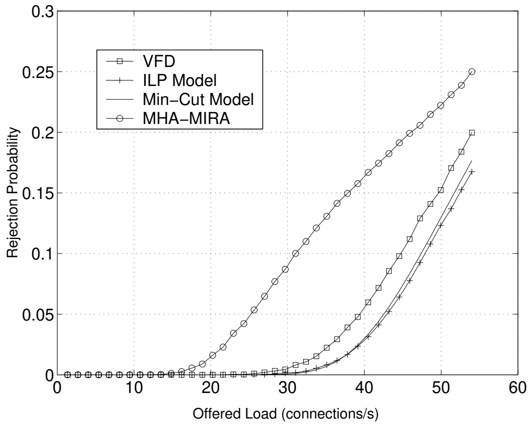


Fig. 3. Connection rejection probability versus the average total load offered to the network of Fig. 2 with link capacity equal to 24 bandwidth units

The performance of MHA and MIRA are exactly the same. In fact MIRA operates for the connections between S_2-T_2 the same path selection of MHA, since the path (5-8-9-6) is shorter than (5-1-2-3-6). VFD achieves the best performance since it exploits the information on the unbalanced load. Although there is still room for improvement in this network scenario, the performance of VFD is very close to the lower bounds provided by the mathematical models.

As a further observation, note that the two models achieve practically the same performance. This results is very important for two reasons: first of all it provides a validation of the two models, at least in this simple network topology; furthermore, it allows us to use the Min-Cut model to provide lower bounds to the performance achievable by QoS routing algorithms in other network scenarios. We noted before that this second model requires much less computation resources than the ILP model, and can be therefore applied even to more complex network configurations.

In the second network scenario, we have considered a simple variation of the above situation, with again the same network topology illustrated in Fig. 2, with all links having capacity equal to 60 bandwidth units. The maximum multi-commodity flow is now equal to 120 bandwidth units.

We have considered two different situations: in the first, all the connections offered to the network always request 1 bandwidth unit; in the second, each connection requires a bandwidth uniformly distributed between 1 and 3 units. Hence, the number of different traffic classes to be considered in equations (7)-(9) in the extended Erlang formula, N, is respectively equal to 1 and 3.

Fig. 4 (a) and (b) show the numerical results of the different routing algorithms together with the bound calculated using equation (9) in the Min-Cut model. As usually, the performance of MHA and MIRA are exactly the same in this network configuration. Evidently, the VFD algorithm achieves better performance than the other routing algorithms in the two situations considered, and its performance is very close to the lower bound calculated using the mathematical model.

Finally, in the last network scenario, we considered the network topology shown in Fig. 5, with 10 nodes and 10 unidirectional links, all having a capacity equal to 60 bandwidth units. The traffic is offered by two source/destination pairs, S_1-T_1 and S_2- T_2.

Also in this network scenario we have considered the two different situations illustrated above, with $N = 1$ and 3 traffic classes offered to the network.

Even in this network configuration, the maximum multi-commodity flow is equal to 120 bandwidth units.

Fig. 6 (a) and (b) show the numerical results of the different routing algorithms together with the bound calculated using the Min-Cut model.

Evidently, the worst performance is achieved by the Min Hop algorithm, which routes all the connections belonging to the S_1-T_1 pair on the path (0,1,3,4,9), thus leading to the saturation of link (3,4).

MIRA, on the contrary, performs in this network situation always the optimal routing choice, as connections belonging to the S_1-T_1 pair are routed on the path (0,1,6,7,8,9) without therefore interfering with the connections belonging to the S_2-T_2 pair.

VFD achieves also in this case a performance higher than MH and slightly worse than that achieved by MIRA. Furthermore, it can be noticed that even in this scenario VFD is very close to the lower bound provided by the mathematical model.

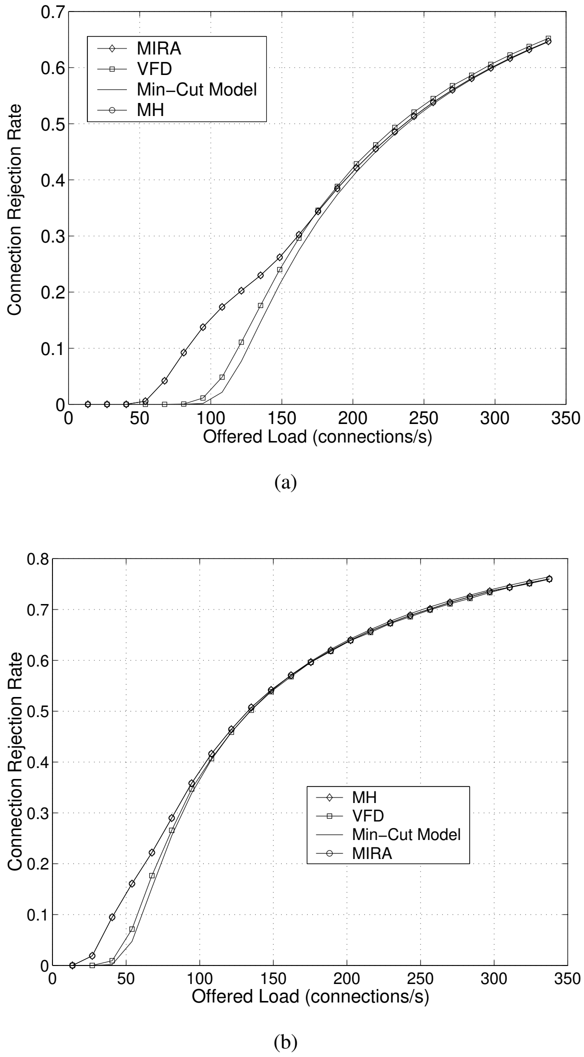


Fig. 4. Connection rejection probability versus the average total load offered to the network of Fig. 2 with link capacity equal to 60 bandwidth units and (a) with $N = 1$ (b) with $N = 3$ traffic classes

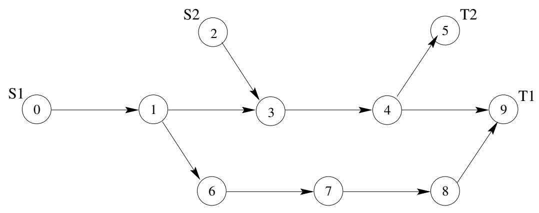
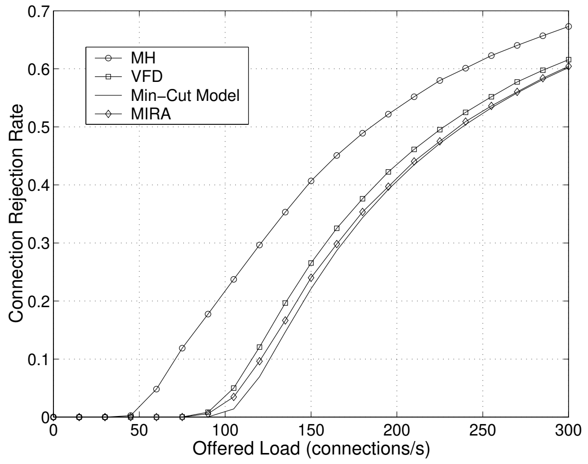


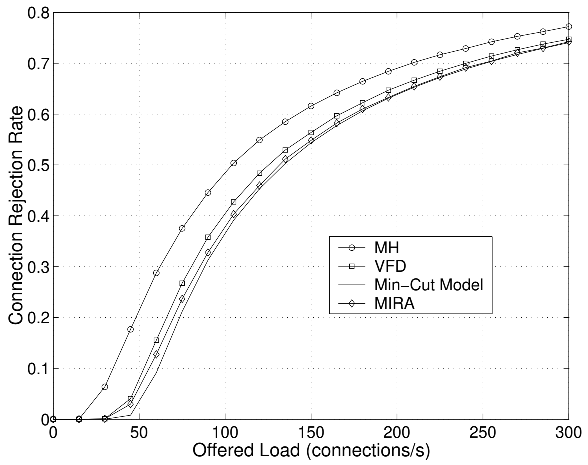
Fig. 5. Network topology with two source/destination pairs

V. Conclusion

We have proposed two novel mathematical models that provide a theoretical bound to the performance achievable by on-line dynamic QoS routing algorithms, expressed as the percentage of rejected calls versus the average total load offered to the network. The first is an ILP model which extends the well known maximum multi-commodity flow problem



(a)



(b)

Fig. 6. Connection rejection probability versus the average total load offered to the network of Fig. 5 (a) with $N = 1$ (b) with $N = 3$ traffic classes

to include connections arrival-times and durations, while the second, which has a much lower complexity, is based on the application of the extended Erlang formula to a link with capacity equal to the residual capacity of the minimum network multi-commodity cut.

We have analyzed the performance of existing QoS routing algorithms, like the Min-Hop Algorithm, the Minimum Interference Routing Algorithm and Virtual Flow Deviation, a new algorithm for explicit-routing of bandwidth guaranteed connections in MPLS networks.

We have shown that VFD not only allows to reduce remarkably the blocking probability in most scenarios with respect to previously proposed schemes, but also it approaches the lower bounds provided by the mathematical models in many network scenarios.

REFERENCES

[1] E.Rosen, A.Viswanathan, and R.Callon. Multiprotocol Label Switching Architecture. In *RFC 3031*, January 2001.

[2] S. Chen and K. Nahrstedt. An Overview of Quality-of-Service Routing for the Next Generation High-Speed Networks: Problems and Solutions, 1998.

[3] D.Banerjee and B.Mukherjee. Wavelength-routed optical networks: linear formulation, resource budgeting tradeoffs, and a reconfiguration study. In *IEEE/ACM Transactions on Networking*, volume Vol 8(5), pages 598–607, Oct 2000.

[4] D.O.Awduche, L.Berger, D.Gain, T.Li, G.Swallow, and V.Srinivasan. Extensions to RSVP for LSP Tunnels. In *Internet Draft draft-ietf-mpls-rsvp-lsp-tunnel-04.txt*, September 1999.

[5] Murali S. Kodialam and T. V. Lakshman. Minimum Interference Routing with Applications to MPLS Traffic Engineering. In *Proceedings of INFOCOM (2)*, pages 884–893, 2000.

[6] A.Capone, L.Fratta, and F.Martignon. Virtual Flow Deviation: Dynamic Routing of Bandwidth Guaranteed Connections. In *Proceedings of 2nd International Workshop on Quality of Service in Multiservice IP Networks, QoS-IP 2003*, Milan, Italy, Feb. 2003.

[7] A.Capone, L.Fratta, and F.Martignon. Dynamic Routing of Bandwidth Guaranteed Connections in MPLS Networks. In *International Journal on Wireless and Optical Communications*, pages 75–86, Vol. 1, No. 1, June 2003, available at http://www.elet.polimi.it/upload/antlab/vfd.pdf.

[8] R.K.Ahuja, T.L.Magnanti, and J.B.Orlin. *Network Flows*. Prentice-Hall, 1993.

[9] *ILOG CPLEX*. Available at http://www.ilog.com/products/cplex/.

[10] Y.Aumann and Y.Rabani. An O(log k) approximate min-cut max-flow theorem and approximation algorithm. In *Manuscript*, 1994.