Routing and admission control with multiconstrained end-to-end quality of service in MPLS networks

Desire Oulai, Steven Chamberland *, Samuel Pierre

Department of Computer Engineering, École Polytechnique de Montréal, C.P. 6079, Succ. Centre-Ville, Montréal, Que., Canada H3C 3A7

Received 23 March 2007; received in revised form 7 December 2007; accepted 17 December 2007
Available online 31 December 2007

Abstract

Multiprotocol label switching (MPLS) networks require dynamic flow admission control to guarantee end-to-end quality of service (QoS) for each Internet protocol (IP) traffic flow. In this paper, we propose to tackle the joint routing and admission control problem for the IP traffic flows in MPLS networks with multiconstrained end-to-end QoS. We propose a mathematical programming model for this problem that includes end-to-end delay and packet loss constraints. These constraints are imposed not only for the new traffic flow, but also for all already admitted flows in the network. Three objective functions are proposed and analyzed: the end-to-end delay, the end-to-end packet loss and the cost of the path used by the new flow. Numerical results show that considering end-to-end delay and packet loss constraints for all flows has a small impact on the flow blocking rate, but improves significantly the end-to-end QoS parameters. Moreover, the proposed approach is able to make the decision to admit or not a new flow in less than 270 ms of computational time.

© 2007 Elsevier B.V. All rights reserved.

Keywords: Multiprotocol label switching (MPLS) networks; Admission control mechanism; Quality of service (QoS); End-to-end QoS constraints; Delay and packet loss constraints

1. Introduction

Multiprotocol label switching (MPLS) networks are typically designed to offer end-to-end quality of service (QoS) for the Internet protocol (IP) traffic flows. The QoS is important for interactive voice and video applications and for specific clients, it is sometimes mandatory to guarantee it. As a result, dynamic admission control is a central mechanism to accept or reject a new flow based on the QoS level requested and the available resources in the network. If there is no adequate admission control mechanism, the network could admit a new flow that overloads links and then downgrades the QoS of several flows.

Traffic routing is also an important mechanism in that context to find a path to the destination router while respecting the QoS constraints. The admission control could use the routing results, but it may be independent from routing. For instance, even if there exists a feasible path, the admission control could reject a flow based on policies. However, most of the admission control mechanisms are routing based.

The admission control could be centralized [4,18] or distributed [1,15]. In the former case, a server gathers information concerning the network state to make the decision. In the latter, the edge routers keep total or partial information and the decision is made by the ingress node. A centralized server may imply a longer setup delay, but the distributed approach is subject to non-optimal decisions. Moreover, the admission control could be reservation based [4] or measurement based [17].

Even though many authors have worked in this area, only a few authors have considered simultaneously several QoS constraints like delay and packet loss. For instance, Cui et al. [5] and Yuan [19] propose to precompute-constrained shortest paths. However, with dynamic traffic flows, these paths may not be optimal for the future requests. Jaffe [9]...
suggested to linearly combine the weights related to each constraint in order to obtain a composite weight for every link. The shortest path to a destination node is found regarding this composite weight. Another approach is the fallback algorithm proposed by Kuipers et al. [12]. This algorithm computes sequentially the shortest path with one QoS parameter while hoping that it will satisfy all the constraints. To our knowledge, no optimal solution has been proposed to satisfy the end-to-end delay and packet loss constraints of all flows exactly without rerouting. We avoid rerouting because it may be service affecting.

In this paper, we propose a mathematical programming model for the joint routing and admission control problem for the IP traffic flows in MPLS networks that includes end-to-end delay and packet loss constraints. These constraints are imposed not only for the new traffic flow, but also for all already admitted flows in the network. Three objective functions are proposed and analyzed: the end-to-end delay, the end-to-end packet loss and the cost of the path used by the new flow. The model, after preprocessing, is solved to optimality.

The rest of the paper is organized as follows. Section 2 presents additional references related to routing and admission control mechanisms. Section 3 presents preliminaries essential for understanding the proposed model. Section 4 presents the mathematical programming model. In Section 5, numerical results are presented and analyzed. Finally, conclusion remarks are presented in Section 6.

2. Additional related works

Routing-based admission control such as constraint-based routing (e.g., with delay or packet loss constraints) is an important area of research. For instance, Kodialam and Lakshman [11] propose the minimum interference routing algorithm (MIRA). The objective is to route the flow request over a path which minimizes the interference with possible future requests. Bagula et al. [3] introduced the least interference optimization algorithm (LIOA). LIOA calculates a cost for each link based on the number of connections through the link and the remaining capacity and then it finds a shortest path. LIOA is less resource-consuming than MIRA and provides less blocking. Capone et al. [4] propose a virtual flow deviation method which allows to split and balance the flows among several paths. Widyono [18] presents an optimal centralized algorithm to find the least cost delay-constrained path. The algorithm (called constrained Bellman-Ford algorithm) performs a breadth-first search to find the optimal path. It is important to mention that end-to-end packet loss constraints are multiplicative constraints that can be transformed into additive constraints. Therefore, the algorithms for delay-constrained routing problems could be adapted to packet loss constrained routing problems.

A more difficult problem is the multiconstrained admission control that considers simultaneously several constraints like delay, jitter, packet loss and bandwidth.

The solutions proposed by Cui et al. [5] and Yuan [19] aimed at precomputing “optimal” constrained shortest paths. However, with dynamic traffic flows, the paths may not be optimal for the future requests. Jaffe [9] suggests to linearly combine the weights related to each constraint in order to obtain a composite weight for every link. The shortest paths are found using those weights. Another approach is the fallback algorithm described in [12]. The principle is to sequentially compute the shortest paths with regard to one QoS measure while hoping that it will satisfy all the constraints.

The admission control can also be done at link (or node) level where each link (or node) has a QoS threshold that cannot be exceeded (e.g., see Cui et al. [6], Nordstrom and Dziong [13] and Spitler and Lee [16]). The complexity of the problem is then reduced but the end-to-end QoS may not be fulfilled.

When doing the admission control for a new request, the service provider has to meet the QoS requirements of the already admitted flows. Indeed, most of the QoS measures such as delay and packet loss are related to the volume of traffic passing on the links. Therefore, the impact of accepting the new flow request has to be evaluated. In that sense, Khan et al. [10] introduce a utility model for optimal routing and admission control. Upon a request, a revenue function is maximized while observing every session with a QoS guarantee. k shortest paths are computed for each connection and the one that minimizes the revenue function is chosen. The algorithm then solves a global optimization problem for each request, which is time-consuming. To get optimal solution of each problem, the authors allow to reroute the flows, which can be service affecting. Ali et al. [1] propose an approach to reserve bandwidth for the already admitted flows to respect delay constraints. Their approach is based on the work by Parekh and Gallager [14]. The main drawback is that only link admission control is considered and this may not be enough for end-to-end delay objectives.

As mentioned before, no authors have considered the joint routing and admission control problem (with end-to-end delay and/or packet loss constrained) without rerouting while guaranteeing end-to-end QoS for all flows in the network.

3. Preliminaries

3.1. The notation

The following notation is used throughout the paper.

3.1.1. Sets

Let $N$ the set of nodes (routers), $M$, the set of unidirectional links, $L$, the set of unidirectional label switched paths (LSPs) and finally, let $T$ be the set of already admitted flows (where the flow $t \in T$ starts at node $O(t) \in N$, terminates at node $D(t) \in N$ and has a traffic request of $x^t$ (in bps), a maximum delay limit of $\beta^t$ (in seconds) and a max-
imum end-to-end packet loss rate limit of $\phi'$. Note that the LSPs are set up between edge nodes (routers) and LSP $(a, b) \in L$ denotes the LSP from the edge node $a \in N$ to edge node $b \in N$.

3.1.2. Constants

Let $y'_{ij}$ be a 0–1 constant such that $y'_{ij} = 1$ if and only if the flow $t \in T$ passes on the link $(i, j) \in M$ and $y''_{ij}$ a 0–1 constant such that $y''_{ij} = 1$ if and only if the LSP $(a, b) \in L$ passes on the link $(i, j) \in M$.

3.1.3. Delay and packet loss functions

Let $d_{ij}(f_{ij})$ be the average delay on the link $(i, j)$, $p_{ij}(f_{ij})$, the average packet loss rate on the link $(i, j)$ and $r_{ij}(f_{ij})$, the average packet transmit rate on the link $(i, j)$, that is, $r_{ij}(f_{ij}) = 1 - p_{ij}(f_{ij})$. In this paper, the $M/M/1/k$ queuing model is used [2]. As a result,

$$d_{ij}(f_{ij}) = \frac{\rho(1 + k\rho^{k+1} - (k + 1)\rho^k)}{\lambda(1 - \rho)(1 - \rho^k)} + a_{ij} \quad (1)$$

$$p_{ij}(f_{ij}) = 1 - r_{ij}(f_{ij}) = \frac{\rho^k(1 - \rho)}{1 - \rho^{k+1}} \quad (2)$$

and

$$\lambda = \frac{f_{ij}}{t} \quad (3)$$

$$\rho = \frac{f_{ij}}{c_{ij}} \quad (4)$$

where, $\lambda$, the mean arrival rate (in packet/s) on the link $(i, j)$; $\rho$, the average utilization of the link $(i, j)$; $k$, the buffer size (in packets); $t$, the mean packet length (in bits); $c_{ij}$, the capacity (in bps) of the link $(i, j)$; $f_{ij}$, the traffic (in bps) on the link $(i, j)$ and finally, $a_{ij}$, the propagation delay on the link $(i, j)$ plus the processing delay at node $i \in N$ (in seconds).

3.1.4. Variables

Let $x_{ab}$ be a 0–1 variable such that $x_{ab} = 1$ if and only if the new flow passes on the LSP $(a, b)$ and $y_{ij}$, a 0–1 variable such that $y_{ij} = 1$ if and only if the new flow passes on the link $(i, j)$.

3.1.5. Objective function

Let $\sum_{(a,b) \in E} f_{ab}(x_{ab})$ be the objective function to minimize where $f_{ab}(x_{ab})$ is a LSP metric representing, for instance, the end-to-end delay for the new flow and the path cost used by the new flow.

3.2. Problem formulation and preprocessing

The joint routing and admission control problem proposed in this paper consists to find a path for the new flow from node $o$ to node $d$ having a traffic request of $x$, with an end-to-end delay limit of $\beta$ and an end-to-end packet loss ratio limit of $\phi$ while considering already admitted flows in the network. If such a path does not exist, the new flow is blocked.

To formulate the mathematical model, preprocessing is necessary. Note that if the new flow does not pass on the link $(i, j)$, the traffic on that link will be

$$F_{ij} = \sum_{t \in T} x_{ij}$$

and the delay on the link $(i, j)$ will be $D_{ij} = d_{ij}(F_{ij})$, the packet loss rate will be $P_{ij} = p_{ij}(F_{ij})$ and the packet transmit rate will be $R_{ij} = 1 - P_{ij}$. Otherwise, the traffic on that link will be

$$F_{ij} = \sum_{t \in T} x_{ij} + \alpha$$

and the delay on the link $(i, j)$ will be $D_{ij} = d_{ij}(F_{ij}) = D_{ij} + \Delta D_{ij}$, the packet loss rate will be $P_{ij} = p_{ij}(F_{ij})$ and the packet transmit rate will be $R_{ij} = 1 - P_{ij}$.

Similarly, if the new flow passes on the LSP $(a, b)$, the end-to-end delay, the transmit rate and the packet loss rate are, respectively, given by the following equations.

$$D_{ab} = \sum_{(i,j) \in M} D_{ij}x_{ij} \quad (7)$$

$$R_{ab} = \prod_{(i,j) \in M} R_{ij} \quad (8)$$

$$P_{ab} = 1 - R_{ab} \quad (9)$$

Otherwise, if the new flow does not pass on the LSP $(a, b)$, we have

$$D_{ab} = \sum_{(i,j) \in M} D_{ij}x_{ij} = D_{ab} + \Delta D_{ab} \quad (10)$$

$$R_{ab} = \prod_{(i,j) \in M} R_{ij}x_{ij} \quad (11)$$

$$P_{ab} = 1 - R_{ab} \quad (12)$$

4. Model formulation

The mathematical model for the joint routing and admission control problem in MPLS networks with end-to-end delay and packet loss constraints, denoted JRAC (joint routing and admission control), can now be given.

$$\text{JRAC} : \min_{\{x_{ab} \in \{0, 1\} \mid L\}} \sum_{(a,b) \in E} f_{ab}(x_{ab}) \quad (13)$$

subject to

$$y_{ij} = \sum_{(a,b) \in L} x_{ab} \forall_{(i,j) \in M} \quad (14)$$

$$0 \leq y_{ij} \leq 1 \forall_{(i,j) \in M} \quad (15)$$

$$\sum_{(a,b) \in L} x_{ab} \leq 2 \quad (16)$$


\[
\sum_{(i,j) \in M} (D_{ij} + y_{ij} \Delta D_{ij}) y_{ij}' \leq \beta \quad \forall t \in T \\
\sum_{(a,b) \in L} D_{ab} x_{ab} \leq \beta \\
\sum_{(i,j) \in M} \left( \ln R_{ij} + y_{ij} \ln \left( \frac{R_{ij}}{R_{ij}'} \right) \right) y_{ij}' \geq \ln(1 - \phi') \quad \forall t \in T \\
\sum_{(a,b) \in L} x_{ab} \ln R_{ab} \geq \ln(1 - \phi) \\
\sum_{(a,b) \in L} x_{ab} - \sum_{(b,da) \in L} x_{ba} \begin{cases} = 1 & \text{if } a = o \\
= -1 & \text{if } a = d \\
= 0 & \text{otherwise} \end{cases} \\
x_{ab} \in \{0, 1\} \quad \forall (a,b) \in L 
\] (17) (18) (19) (20) (21) (22)

Three objective functions are considered and analyzed in this paper.

1. Minimize the end-to-end delay for the new flow and JRAC-D (JRCA delay) denotes the model JRAC when \( f_{ab}(x_{ab}) = D_{ab} x_{ab} \), i.e., the delay on the LSP \((a,b)\) for the new flow.
2. Minimize the end-to-end packet loss for the new flow or, equivalently, maximize the end-to-end path packet transmit rate for the new flow. As a result, since the logarithmic function \( \ln(.) \) is an increasing function, the objective function

\[
\max \prod_{(a,b) \in L, x_{ab}=1} R_{ab} \\
\equiv \max \ln \left( \prod_{(a,b) \in L, x_{ab}=1} R_{ab} \right) \\
\equiv \max \sum_{(a,b) \in L} x_{ab} \ln R_{ab} \\
\equiv -\min \sum_{(a,b) \in L} -x_{ab} \ln R_{ab} 
\]

and JRAC-P (JRCA packet) denotes the model JRAC when \( f_{ab}(x_{ab}) = -x_{ab} \ln R_{ab} \).
3. Minimize the path cost used by the new flow and JRAC-C (JRAC cost) denotes the model JRAC when \( f_{ab}(x_{ab}) = U_{ab} x_{ab} \) where \( U_{ab} \) is the cost of the LSP \((a,b) \in L \) chosen or computed by the network engineer.

Constraints (14) force the variable \( y_{ij} \) to be equal to the number of LSPs used by the new flow passing on the link \((i,j)\) and constraints (15) impose this number be less than or equal to one, i.e., the new flow is allowed to pass on a link at most once. Constraint (16) limits the number of LSPs used by the new flow to at most two. This limit is applied to facilitate the admission control process. Indeed, if two LSPs are used, the controller has just to communicate with the ingress node because any intermediate node that receives a packet will forward it by the direct LSP to the destination node. Moreover, in the next section, it is shown that the blocking rates with limits of two, three and eight LSPs are similar (see Fig. 7). Constraints (17) impose each already admitted flow in the network to respect the end-to-end delay constraints and constraint (18), the new flow to respect the end-to-end delay constraint. Constraints (19) impose each already admitted flow in the network to respect the end-to-end packet loss constraints and constraint (20), the new flow to respect the end-to-end packet loss constraint. These linear constraints are obtained by logarithmic transformations. Indeed, we want for each already admitted flow \( t \in T \) that

\[
\prod_{(i,j) \in M} R_{ij} - \prod_{(i,j) \in M} R_{ij}' \geq 1 - \phi' \\
\equiv \ln \left( \prod_{(i,j) \in M} R_{ij} \right) - \ln \left( \prod_{(i,j) \in M} R_{ij}' \right) \geq \ln(1 - \phi') \\
\equiv \sum_{(i,j) \in M} \left( \ln R_{ij} + y_{ij} \ln \left( \frac{R_{ij}}{R_{ij}'} \right) \right) y_{ij}' \geq \ln(1 - \phi') 
\]

Indeed, the term \( \ln R_{ij} + y_{ij} \ln \frac{R_{ij}}{R_{ij}'} = \ln R_{ij} \) if \( y_{ij} = 0 \) and \( \ln R_{ij} + y_{ij} \ln \frac{R_{ij}}{R_{ij}'} = \ln R_{ij} \) if \( y_{ij} = 1 \). Inequality (23) for all \( t \in T \) are the constraints (19). Constraints (20) can be obtained similarly. Indeed, we want for the new flow that

\[
\prod_{(a,b) \in L, x_{ab}=1} R_{ab} \geq 1 - \phi \equiv \ln \left( \prod_{(a,b) \in L, x_{ab}=1} R_{ab} \right) \geq \ln(1 - \phi) \\
\equiv \sum_{(a,b) \in L} x_{ab} \ln R_{ab} \geq \ln(1 - \phi) 
\]

To enumerate constraints (19) and (20), the preprocessing should verify that \( \phi < 1, \phi' < 1 \) for all \( t \in T, R_{ij} > 0 \) for all \( (i,j) \in M, R_{ij} > 0 \) for all \( (i,j) \in M \) and \( R_{ab} > 0 \) for all \( (a,b) \in L \). Constraints (17)–(20) are central because they permit to meet the QoS requirements for all flows (including the new flow). Constraints (21) are the flow conservation constraints and, finally, constraints (22) are integrality constraints. This way of formulating the multi-constrained JRAC problem is original and, in particular, the linearization of the packet loss constraints.

JRAC is \( \mathcal{NP} \)-hard (transformation from the shortest weight-constrained path problem [7]). However, since the number of integer variables is small, JRAC can be solved to optimality for real-size instances of the problem within a small amount of computational time.

Note that the number of constraints (17) and (19) is \( 2|T| \) where \(|T| \) is the number of already admitted flows. Since this number can be important, it could be computationally expensive to consider all these constraints. However, these constraints can be easily reduced. First note that only traffic flows that have a common link with the new one will be affected. Since the maximum number of LSPs on a path is two, we maintain a table of all LSPs and pairs of adjacent LSPs. For each LSP, the new end-to-end delay limit is set to the minimum of the end-to-end delay limit of each flow using the path composed only by this LSP. Similarly, for each pair of LSPs, the new end-to-end delay limit is set.
to the minimum of the end-to-end delay limit of each flow using the path composed by this pair of LSPs. For each path, we calculate the delay by adding the $D_{ab}$. If the result is less than or equal to the end-to-end delay limit for the path, we do not need to write the constraints (17) for the flows using this path. Similarly, we can reduce the number of constraints (19).

The proposed joint routing and admission control mechanism is used for each flow request while considering the already admitted flows in the network. This per-flow decision is the dynamic characteristic of the mechanism since the requests are arriving randomly.

5. Numerical results

In this section, we evaluate the performance of the proposed routing-based admission control mechanism. All algorithms were programmed in the C language on a Sun Java workstation under Linux with an AMD Opteron 150 CPU and 2 GB of RAM. For solving the model JRAC-D, the CPLEX Mixed Integer Optimizer 9.0 (see [8] for more information about CPLEX) is used. Note that the algorithm used by the CPLEX is the branch-and-bound algorithm. The default settings of CPLEX are used.

The test networks are presented in Table 1. The first column presents the number of nodes $|N|$, the second column presents the number of flow requests $|T|$ and the two last columns present the number of edge nodes and the number of links. These networks are randomly generated and the capacity of each link is set to 100 Mbps. The packet length is set to 1500 bytes and the buffer size to 1200 kbytes (i.e., $k = 800$ packets).

To obtain the numerical results, $|T|$ flow requests are generated randomly and after each one, the proposed mechanism is executed. For each request, we randomly choose a pair of origin-destination edge nodes as well as the end-to-end delay limit in the set $\{50, 100, 150, 200, 250, 300\}$ in ms and the end-to-end packet loss ratio limit in the set $\{0.01, 0.02, 0.03, 0.04, 0.05\}$. Each request has a bandwidth of 1 Mbps.

The metrics of interest to evaluate the performance of our mechanism are the flow blocking rate, the mean end-to-end delay, the mean end-to-end packet loss, the ratio of constraints (17) and (19) violated (when they are not included in the model), the CPU execution time and, finally, the impact of the limit imposed on the number of LSPs that can be used to reach the destination.

5.1. Comparing JRAC-D with the classical approach

In this subsection, the solutions found by solving the model JRAC-D are compared with those found by solving the model without the constraints (17) and (19) which is called the “classical approach”. In fact, because no comparable joint routing and admission control mechanism has considered end-to-end delay and end-to-end packet loss constraints for the already admitted flows, a good approach to evaluate the impact of considering those constraints, is to remove them, find the optimal solution and compare the solutions.

Fig. 1 presents the flow blocking rate for each test network. The blocking rates are a little higher with constraints (17) and (19) (the difference less than 7%). However, if we observe the mean end-to-end delay and the mean end-to-end packet loss for each test network (see Figs. 2 and 3), we conclude that better results are obtained with constraints (17) and (19) assuring end-to-end QoS to be respected for each flow.

| $|N|$ | $|T|$ | Number of edge nodes | Number of links |
|------|------|----------------------|----------------|
| 5    | 400  | 3                    | 10             |
| 10   | 800  | 6                    | 15             |
| 20   | 1600 | 12                   | 30             |
| 30   | 2400 | 18                   | 45             |
| 40   | 3200 | 24                   | 60             |
| 50   | 4000 | 30                   | 75             |
| 60   | 4800 | 36                   | 90             |
| 70   | 5600 | 42                   | 105            |
| 80   | 6400 | 48                   | 120            |
We want to point out with those results that with blocking a small amount of selected flows, it is possible to improve a lot the network performance in term of the end-to-end delay and end-to-end packet loss.

Fig. 4 presents the ratio of constraints violated (when they are not included in the model). The figure shows that up to 52% of the flows do not respect the end-to-end delay limit when constraints (17) are considered. Moreover, Fig. 5 presents the ratio of constraints (19) violated (when they are not included in the model). The maximum ratio is 51%.

Fig. 6 presents the mean CPU execution time for the proposed admission control mechanism as a function of the number of nodes considering that \( |T| \) flows are already admitted in the network (see Table 1). This figure shows that the decision time (to admit or not a new flow in the network) is less than 270 ms and CPLEX takes less than 55 ms to solve the model. As a result, even for networks with 80 nodes and 6400 flow requests, the decision time is relatively acceptable (less than 1 s). The processing time before launching CPLEX is an important part of the CPU time and code optimization should be explored to minimize this part.

Finally, Fig. 7 compares the flow blocking rate as a function of the maximum number of LSPs that can be used to reach the destination (see constraint (16)). This figure shows that the blocking rates with limits of two, three and eight LSPs are similar. Therefore, constraint (16) does not have a negative impact on the flow blocking rate when an LSP is set up between each pair of edge nodes. If the number of LSPs is small, however, from the flow blocking rate perspective, these constraints can have a negative impact on the network performance.

It can be gathered from those figures that JRAC-D provides good results: we have up to 7% of blocked flows but the number of flows that do not respect the end-to-end QoS limit is significantly reduced (up to 52%) as well as the mean end-to-end delay and packet loss (up to 51%).

5.2. Performance of JRAC with different objective functions

In this subsection, the effect of the different objective functions on the numerical results is analyzed. First, Fig. 8 presents the blocking rate of new flows for each test network. JRAC-C leads to more blocking since the link costs are fixed. As a result, some links are congested rapidly so more flows will be blocked. On the other hand, Figs. 9 and 10 present the mean end-to-end delay and mean end-to-end packet loss for each test network. All mean end-to-end delays are less than 25 ms. Sometimes, JRAC-D
provides not the best result, but it is related to the quantity of admitted flows. For the packet loss, JRAC-P gives lower results most of the time.

Fig. 11 presents the mean CPU execution times. JRAC-D provides the highest execution times. This can be explained by the fact that JRAC-D accepts more flows, then the number of constraints to write in the model is more important.

JRAC-D seems to be the best model for the dynamic admission control of flows in MPLS networks. Indeed, it provides lower flow blocking rate than the other models while remaining competitive regarding the end-to-end QoS parameters and the computational time.

6. Conclusions

In this paper, we proposed a mathematical programming model for the joint routing and admission control problem for the traffic flows in MPLS networks that includes end-to-end delay and packet loss constraints. These constraints are imposed not only for the new traffic flow, but also for all already admitted flows in the network. Three objective functions are proposed and analyzed: the end-to-end delay, the end-to-end packet loss and the total cost of the path used by the new flow. The model, after preprocessing, is solved to optimality.

The numerical results show that considering end-to-end delay and packet loss constraints for already admitted flows has a small impact on the flow blocking rate, but reduces significantly the mean end-to-end QoS parameters of each flow. We also demonstrate that minimizing the end-
to-end delay on the path for the new flow is the best objective in a multiconstrained context.

There are several avenues of research open at this point. First, we are currently exploring models with end-to-end jitter in addition to the end-to-end delay and packet loss constraints. Moreover, the processing time before launching CPLEX is an important part of the CPU time and code/algorithm optimization should be explored to minimize this part. Finally, we currently work on efficient heuristic algorithm to find good (quasi-optimal) solutions rapidly.

Acknowledgement

The completion of this research was made possible thanks to Bell Canada’s support through its Bell University Laboratories R&D program.

References