Dipartimento di Informatica
e Sistemistica Antonio Ruberti

Sapienza
Università di Roma

An algorithm for routing optimization in DiffServ-aware MPLS networks

Luigi Atzori
Fabio D'Andreagiovanni
Carlo Mannino
Tatiana Onali

Technical Report n. 4, 2010

An algorithm for routing optimization in DiffServ-aware MPLS networks

Luigi Atzori[1], Fabio D'Andreagiovanni[2], Carlo Mannino[2], Tatiana Onali[1]

[1] Dipartimento di Ingegneria Elettrica ed Elettronica
Università di Cagliari
piazza d'Armi
09123 Cagliari, Italy
E-mail: `(l.atzori,tatiana.onali)@diee.unica.it`

[2] Dipartimento di Informatica e Sistemistica "A. Ruberti"
Sapienza - Università di Roma
via Ariosto, 25
00185 Rome, Italy
E-mail: `(f.dandreagiovanni,mannino)@dis.uniroma1.it`

Abstract

This paper addresses the constrained-based routing problem in DiffServ-aware MPLS networks. We consider a dynamic context in which new requests appear over time, asking for reconfigurations of the previous allocation. In the classical approach, a multi-phase heuristic procedure is adopted: the new requests are evaluated considering available bandwidth; if the bandwidth is not sufficient, preemption and rerouting of one or more connections are performed in sequence. As an alternative, we propose a single-phase approach that simultaneously takes into account both preemption and rerouting. In contrast with the standard approach, we always find an optimal solution when enough bandwidth is available. Otherwise, we apply a heuristic post-processing procedure in order to minimize unsatisfied commodities. The routing problem is modeled as a Multicommodity Flow Problem (MCFP) with side constraints, which is solved by a Column Generation approach. Namely, a sequence of restricted MCFPs is solved, by including new routing paths only if necessary. When new requests are routed, the use of existing paths is preferred in order to reduce preemption. Computational experience on real networks shows that the overall approach is able to (i) obtain a good exploitation of the network resources, (ii) achieve a remarkable acceptance rate and, (iii) hold down the impact of rerouting.

Keywords: DiffServ-aware MPLS networks, routing optimization, Multicommodity Flow Problem.

1 Introduction

During the last decade, the huge growth of Internet has brought a lot of attention on *Traffic Engineering* (TE) [1], that is designing network routes with the aim of reaching a better exploitation of available resources, while avoiding overloading. TE has become essential especially after the emergence of new multimedia applications: services such as voice over IP, video on demand and streaming indeed require precise performance guarantees, that entail advanced planning procedures.

As traditional IP networks can only provide best effort services, over the years the *Internet Engineering Task Force* (IETF) has proposed several architectures and protocols, such as *Differentiated Service* (DiffServ) [12] and *Multi-Protocol Label Switching* (MPLS) [16], that support network traffic management and the provision of Quality of Service (QoS) guarantees. DiffServ allows classification and differentiated management of traffic flows: flows are categorized in a (small) number of traffic classes and, on the basis of their class, are managed according to specific rules. Higher priority traffic can thus receive a preferential treatment. MPLS supports a complete specification of routing paths, through fixed-length labels attached to packets. This enables to find paths that satisfy the QoS constraints. DiffServ and MPLS have established themselves as complementary in provisioning end-to-end QoS and their combined use has lead to the *DiffServ-aware Traffic Engineering* (DS-TE): this is an effective scalable solution to support QoS, while allowing an efficient use of network resources, through a balanced distribution of traffic load on available links, fast-rerouting and bandwidth reservation.

In a MPLS based network, routes (*Label Switching Paths*, LSP) with higher priority are allowed to preempt resources of lower priority LSPs, in case of lack of resources. Preemption entails the termination of lower priority LSPs and the release of the corresponding resources, which are then used to establish the higher priority LSPs. Terminated LSPs have to be rerouted on different paths, in order to avoid service disruption. However, rerouting may require other preemption actions, leading to a cascade effect that can overload network routers and drastically reduce QoS.

Within the DS-TE framework, this paper addresses the constrained-based routing problem in presence of preemption and rerouting. The problem fundamentally consists in satisfying connection requests, by establishing a suitable set of LSPs, that furnishes the requested amount of bandwidth and the desired level of QoS. In particular, our study refers to an online context, where the set of requests is not static but evolves over time.

In the classical approach, a multi-phase heuristic procedure is adopted: the new requests are evaluated considering the available bandwidth; if the bandwidth is not sufficient, preemption and rerouting of one or more connections is performed in sequence. As an alternative, we propose a single-phase approach that simultaneously takes into account both preemption and rerouting. In contrast with the standard approach, we always find an optimal solution when enough bandwidth is available. Otherwise, we apply a heuristic post-processing procedure in order to minimize unsatisfied commodities.

We model the routing problem as a Multicommodity Flow Problem (MCFP) with side constraints, which is solved by a Column Generation approach. Namely, a sequence of restricted MCFPs is solved, by including new routing paths only if necessary. When new requests are routed, the use of existing paths is boosted in order to reduce preemption. Computational experience on real networks shows that the overall approach is able to (i) obtain a good exploitation of the network resources, (ii) achieve a remarkable acceptance rate and, (iii) hold down the impact of rerouting.

The paper is organized as follows. Section II illustrates the basis of the DS-TE and briefly presents significant past works. Section III introduces the routing problem and the corresponding model, whereas Section IV describes the proposed DS-TE network routing algorithm. Finally, Section V presents the experimental results and conclusions are drawn in the last section.

2 Preliminaries

2.1 Past works

In the case of MPLS based architectures, one of the most advanced routing algorithms is the *Minimum Interference Routing Algorithm* (MIRA) [3], which minimizes the number of refused connection requests by selecting paths with the maximum available bandwidth and avoiding link bottlenecks. An improved version of the MIRA, namely the *Least Interference Optimization Algorithm* (LIOA) [4], has been developed to contain the interference of the new connections with the existing ones. In particular, interference over a link is evaluated through a cost function, that takes into account the number of active flows in the link and the difference between the maximum reservable bandwidth and the bandwidth reserved for active connections.

Another algorithm that uses the interference concept is the one proposed in [5]. It is made up of two phases: first, all the information about the requests is exploited to solve a multicommodity flow problem and to preallocate the link bandwidth to each traffic class; then, each request is routed making use of a shortest path algorithm. In [6] a MCFP is also used to solve the routing problem, but, in case of lack of resources, rerouting is used to modify the organization of the used LSPs.

In [7], the authors propose an improved methodology for the targeting of paths that are better candidates to preemption. The selection of the candidates is guided by the minimization of a function that takes into account (i) the priority of the preempted LSP, (ii) the number of preempted LSPs and

(iii) the overall preempted bandwidth. The effectiveness of the method is strongly subject to the choice of the weights that multiply the three distinct objectives in the function. Furthermore, the method does not address the issue of defining the path of the request that needs preemption as well as the issue of defining the rerouting paths of preempted requests.

2.2 DS-TE architecture

DS-TE technology [2] implements Traffic Engineering on a per-class basis, by introducing both end-to-end QoS guarantees and network scalability in a differentiated service environment. Thanks to the adoption of the MPLS, it allows to establish routes for individual flows, i.e. different flows sent between the same end points may follow different routes [8]. More precisely, in a MPLS domain, packets of the incoming traffic flows are marked by the Label Edge Routers (LERs) with a label that completely specify the LSP used for routing. Each LSP is characterized by a set of attributes, such as the requested bandwidth, the priority and the preemption level. In particular, the preemption level indicates which are the LSPs that the considered LSP may preempt to free resources.

DS-TE extends the MPLS Traffic Engineering approach to multiservice environments, where services with different QoS requirements have to be managed in a scalable, flexible and dynamic way. This goal is achieved by introducing the concepts of class type (CT), TE-class and bandwidth constraint (BC) model. A CT is a set of traffic trunks with similar QoS requirements and that are subject to the same constraints on bandwidth utilization. A TE-class is a combination of a CT and a preemption value. It enables traffic trunks belonging to the same CT to be forwarded on different LSPs at different priorities and to preempt bandwidth used by lower priority LSPs. Finally, a BC model defines the portion of bandwidth that can be used in a link to route traffic belonging to each CT. Such kind of bandwidth reservation is operated during the configuration phase of the network.

When a new LSP setup request arrives, it is classified in a TE-Class and is associated to a CT and a preemption level on the basis of its specific QoS requirements. The computation of LSPs is performed through constraint based routing (CBR) algorithms [9], that find paths satisfying a set of constraints, such as the ones on bandwidth, QoS requirements and number of hops. CBRs can be performed either offline or online. In the first case, LSPs are pre-established during the network configuration phase on the basis of expected traffic demand [10]. In the second case, LSP routing is done when a new request arrives. In our study, we will refer to the second case.

3 Problem definition and modeling

In this section we introduce the elements and the modeling assumptions which provide the basis of the optimization model presented in Section 4. We consider the online routing process on DS-TE networks: given a network where a set of LSPs is already established to satisfy former user requests and given a set of new user requests, our goal is to define the LSPs needed to satisfy the new requests and the traffic flow sent on them. Preemption and rerouting are allowed in order to establish higher priority requests in case of resources lack.

We model the DS-TE network as a directed graph $G(V, E)$ where the n-elements set of vertices V and the m-element set of edges $E \subseteq \{(i, j) : i, j \in V, i \neq j\}$ correspond to the sets of routers and communication links, respectively. Each edge $e = (i, j) \in E$ is associated with three non-negative parameters: (i) bandwidth capacity b_e; (ii) delay d_e; (iii) routing cost c_e. A LSP p with ingress router s and egress router t corresponds to a directed path from s to t, i.e. a sequence of l edges $\{e_1 = (s, v_1), e_2 = (v_1, v_2), \ldots, e_l = (v_{l-1}, v_l)\}$. We use the notation $e \in p$ to indicate that an edge e is included in path p. For each path p we define the cost $c_p = \sum_{e \in p} c_e$ and the delay $d_p = \sum_{e \in p} d_e$.

User requests are represented by a set of commodities K. Each commodity $k \in K$ is identified by a quintuple $(s_k, t_k, B_k, D_k, \sigma_k)$, where s_k and t_k are the ingress and egress vertices, respectively, B_k is the requested bandwidth, D_k the maximum delay and σ_k the priority class. For each commodity $k \in K$ we define the set of feasible paths P_k, i.e. the subset of all the directed paths p from s_k to t_k that satisfy the delay constraint $d_p \leq D_k$. Finally, we introduce the set $P = \bigcup_{k \in K} P_k$ of all the commodities paths. A commodity $k \in K$ is satisfied if the sum of the bandwidth used to send traffic flow on each feasible path $p \in P_k$ equals the requested bandwidth B_k of the commodity.

Given the network graph $G(V, E)$, the vector of bandwidth capacity, $b \in Z_+^{|E|}$, the vector of delay $d \in Z_+^{|E|}$, the vector of routing cost $c \in Z_+^{|E|}$ and the set of commodities $k = (s_k, t_k, B_k, D_k, \sigma_k) \in K$ the *DS-TE Network Routing Problem (DS-TE NRP)* is the one of establishing a set of paths P^* and the traffic flow sent on them so as to satisfy all the commodities and minimize the overall routing cost. If not all commodities can be satisfied, MCFP asks for minimizing the number of unserved commodities, according to their priority class.

4 DS-TE network routing

4.1 Optimization procedure

The DS-TE NRP can be naturally modeled as a *Multicommodity Flow Problem* [13]. In particular, we refer to a formulation based on path flows, by introducing a non-negative variable f_p to represent the traffic sent on each delay-feasible path $p \in P_k$ of each commodity $k \in K$. The following formulation is obtained:

$$\min \quad \sum_{k\in K}\sum_{p\in P_k} c_p f_p \tag{1}$$

$$\text{s.t.} \quad \sum_{p\in P_k} f_p = B_k \qquad \forall k \in K \tag{2}$$

$$\sum_{p\in P_k : e\in p, k\in K} f_p \le b_e \qquad \forall e \in E \tag{3}$$

$$f_p \ge 0 \qquad \forall k \in K, p \in P_k \tag{4}$$

The objective function (1) expresses the total cost of routing. Constraint (2) ensures that traffic demand of commodity $k \in K$ is satisfied. Capacity constraint (3) ensures that the sum of all the flows on edge $e \in E$ does not exceed edge capacity. Finally, constraint (4) expresses the non-negativity of flow variables.

We recall that each path $p \in P_k$ satisfies the side constraint $\sum_{e\in p} d_e \le D_k$, that is not explicitly included into the formulation. In [15] MCFP with side constraints is proven to be NP-Hard.

The use of a formulation based on path flows leads to a system of linear inequalities with a huge number of variables, which is impossible to write down even for moderately large size instances. However, this issue can be tackled by adopting a *Column Generation* approach [13], in which variables (i.e. paths) are explicitly inserted in the problem only when needed: initially just a subset of paths $Q \subseteq P$ is considered, thus obtaining a *restricted problem* (R-MCFP). At each iteration, the solution algorithm solves the current R-MCFP to optimality, obtaining a solution f^*. If the solution is optimal also for MCFP (the so called *Master Problem*), the algorithm terminates, otherwise we add a suitable set of paths $\tilde{Q} \subseteq P \backslash Q$ to Q and iterate the procedure. By standard arguments of Linear Programming theory (see, for example, [11]), every path $q \in \tilde{Q} \cap P_k$ corresponds to a variable f_q of the overall problem MCFP with negative reduced cost $\tilde{c}_q$ w.r.t. to the current solution of the restricted problem R-MCFP, that is

$$\tilde{c}_q = \sum_{e\in q}(y_e^* + c_e) - z_k^* < 0 \tag{5}$$

where (z^*, y^*) is the optimal solution to the following linear program (the *dual* of R-MCFP):

$$\begin{aligned} \max \quad & \sum_{k\in K} B_k z_k - \sum_{e\in E} b_e y_e & & (6)\\ \text{s.t.} \quad & z_k - \sum_{e\in p} y_e \le c_p & \forall p \in Q & \\ & y_e \ge 0 & \forall e \in E & \end{aligned}$$

Finding a negative reduced cost path thus corresponds to finding a commodity k and a path $q \in P_k \backslash Q_k$ whose *extended cost* $\sum_{e\in q}(y_e^* + c_e)$ is less than z_k^*. Given a commodity $k \in K$, this can be accomplished by finding a path that is shortest with respect to the extended costs of the edges and satisfies the delay constraint D_k. The column generation problem thus reduces to solve a *Constrained Shortest Path Problem* (CSPP) [14]: if the constrained shortest path satisfies (5) for some $k \in K$, we add the corresponding paths to R-MCFP and iterate, otherwise we can conclude that the current solution f^* of R-MCFP is also optimal for the master problem MCFP. In (Alg. 1), we formalize the algorithm to solve the MCFP associated to DS-TE NRP.

Algorithm 1 MCFP Solver

Input: graph $G(V, E)$, set of commodities K
initial set of paths $Q \subseteq P$
Output: optimal flow vector f^*

1 Solve R-MCFP on Q and obtain primal and dual optimal solution f^* and (z^*, y^*)
while there exists a path with negative reduced cost **do**
  for $k = 1$ to $|K|$ **do**
    2 Solve CSPP for k w.r.t. the extended edge cost $(y_e^* + c_e)$ for all $e \in E$ and obtain the CSP $\tilde{p}$
    if $z_k^* > \sum_{e\in\tilde{p}}(y_e^* + c_e)$ **then**
      $Q := Q \cup \{\tilde{p}\}$
    end if
  end for
  3 Solve R-MCFP on Q and obtain primal and dual optimal solution f^* and (z^*, y^*)
end while

In an online routing framework, the set K dynamically evolvs as old requests are satisfied thus freeing bandwidth resources and new requests are generated. Consider now a network on which a set of commodities K^{old} is currently processed and suppose that a new set of commodities K^{new} arrives and must be processed. The standard approach attempts first to route K^{new} by using the residual bandwidth. If such bandwidth is not sufficient to route all of the new commodities with higher priority class, preemption is used to free bandwidth by interrupting some lower priority commodities in K^{old}; then rerouting is used, when possible, to find alternative paths for the preempted commodities. We remark here that this approach may fail to serve all commodities even if the available bandwidth suffices.

In contrast, we propose an alternative approach that always finds an optimal routing when enough bandwidth is available. In particular, when a set K^{new} is generated, we solve a MCFP with $K = K^{old} \cup K^{new}$. If MCFP is feasible, then all $k \in K$ are satisfied, the algorithm outputs an optimal solution and we are done. Otherwise, no solution satisfying all current commodities exists and we resort to a heuristic approach to find a suitable subset of commodities to serve.

To this end, we define an auxiliary network $G(V, \tilde{E})$ by adding, for each commodity $k = (s_k, t_k, B_k, D_k, \sigma_k) \in K$, the edge $\tilde{e}_k = (s_k, t_k)$ to E and we let (i) $b_{\tilde{e}_k} = B_k$; (ii) $d_{\tilde{e}_k} = 0$; (iii) $c_{\tilde{e}_k} >> max\ \{c_p : p \in P\}$. Observe that by doing this, we actually introduce an artificial path $\tilde{\pi}_k = \{\tilde{e}_k\}$ from s_k to t_k, that is delay feasible for k and provides all the required bandwidth B_k (see Figures 1, 2). Let $\tilde{P} = \bigcup_{k \in K} \{\tilde{\pi}_k\}$ be the set of (artificial) paths of $G(V, \tilde{E})$, and let P be the set of real paths. Since the cost of an artificial path is much higher than that of any path in P, flow is sent on $\tilde{\pi}_k \in \tilde{P}$ only if strictly necessary, that is no more bandwidth can be used on the paths in P. Thus, by using the auxiliary graph $G(V, \tilde{E})$, the MCFP reads as:

$$\begin{aligned} \min \quad & \sum_{p \in P \cup \tilde{P}} c_p f_p && (7) \\ \text{s.t.} \quad & \sum_{p \in P_k \cup \{\tilde{\pi}_k\}} f_p = B_k && \forall k \in K \\ & \sum_{p \in P_k \cup \{\tilde{\pi}_k\} : e \in p, k \in K} f_p \leq b_e && \forall e \in \tilde{E} \\ & f_p \geq 0 && \forall k \in K, p \in P_k \cup \{\tilde{\pi}_k\} \end{aligned}$$

Since we want to contrast rerouting of previously established traffic flows for the commodity in K^{old}, we consider a slightly modified objective function. Denoting by P^{old} the subset of real paths $p \in P$ previously used for routing the old commodities K_{old}, the new objective function becomes:

$$\sum_{p \in P^{old}} \mu c_p f_p + \sum_{p \in P \setminus P^{old}} c_p f_p$$

Parameter μ is chosen in the interval $[0, 1]$ so to scale down the original cost of the paths in P^{old}.

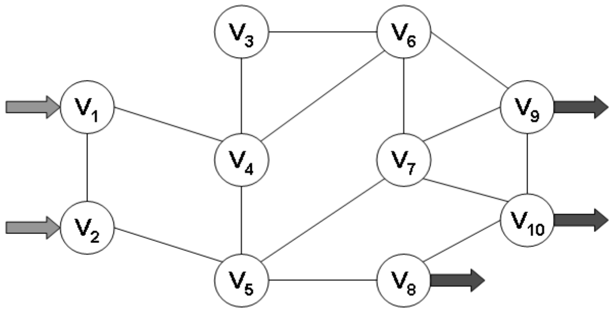


Figure 1: A sample graph $G(V, E)$ associated to 3 commodities with origins-destinations (v_1, v_9), (v_2, v_8), (v_2, v_{10}).

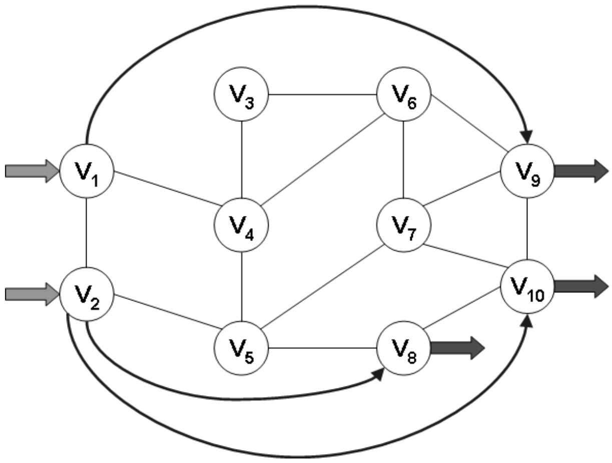


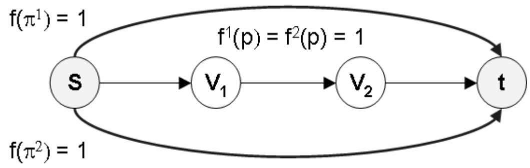
Figure 2: The auxiliary graph of Fig. 1 obtained by adding artificial paths $\tilde{\pi}_1 = \{(v_1, v_9)\}$, $\tilde{\pi}_2 = \{(v_2, v_8)\}$, $\tilde{\pi}_3 = \{(v_2, v_{10})\}$ to $G(V, E)$.

4.2 Post processing

Given an optimal solution f^* to the modified MCFP, every commodity $k \in K$ with a positive flow on the corresponding fictitious path π_k, i.e. $f_{\pi_k} > 0$, is not satisfied. We can then partition K into the set of satisfied commodities K^{sat} and the set of unsatisfied commodities K^{unsat}.

However, it may happen that some $k \in K^{unsat}$ could actually be satisfied, since, due to the linearity of the objective function, other optimal solutions may have a smaller number of non-zero flows over the artificial edges. In order to clarify this concept, we provide an example based on Figure 3: we consider two commodities $1, 2$ that require to send 2 unit of flows from node s to node t through path $p = \{(s, v_1), (v_1, v_2), (v_2, t)\}$. The solution that sends 1 unit of flow on the real path p and 1 unit on the artificial path for both commodities has the same cost of the solution that sends the entire flow of commodity 1 on p and the entire flow of commodity 2 on $\pi_2 = \{(s, t)\}$.

However, in the second solution, commodity 1 turns out to be satisfied as it has a zero flow on its artificial path π_1.



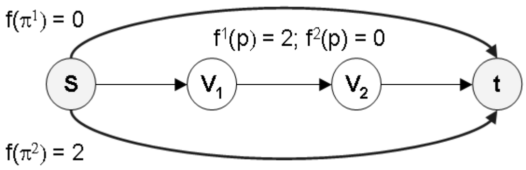


Figure 3: The auxiliary graph of Fig. 4.1 obtained by adding artificial paths $\tilde{\pi}_1 = (v_1, v_9)$, $\tilde{\pi}_2 = (v_2, v_8)$, $\tilde{\pi}_3 = (v_2, v_{10})$ to $G(V, E)$.

We tackle the problem of finding additional satisfiable commodities by the following heuristic post-processing. The basic idea is to solve a sequence of MCFP on the original graph $G(V, E)$, testing the satisfiability of a different set of commodities at each run. First, we order $k \in K^{unsat}$ by decreasing priority class (ties are broken through a FIFO strategy). Let $(k_1, k_2, \ldots)$ be such an ordering. Now, let K be the set of currently satisfied commodities. In the first iteration, $K = K^{sat}$. At the i-th iteration of the post-processing procedure we solve a MCFP on $G(V, E)$ for the commodities in $K \cup \{k_i\}$. If this problem is feasible, i.e. it is possible to satisfy all commodities in K plus commodity k_i, then we let $K = K \cup \{k_i\}$ and iterate. Otherwise we iterate with K unchanged. The final K includes all the commodities that can be actually satisfied.

5 Numerical results

In this section, we evaluate the performance of our methodology on a set of real instances based on a network topology and traffic scenarios provided by the European ISP Tiscali S.p.a. We consider a single network domain with 22 LERs and 6 core routers interconnected by links with capacities ranging in the interval [0.1,1] Gbps. Each link has unitary cost and is characterized by a transmission delay that falls in the range [0,13] ms. Six different CTs are built on the basis of the application requirements. In particular, they convey the traffic of the following services: CT5 - voice over IP and video teleconferencing; CT4 - Internet Protocol Television; CT3 - interactive non-

Table 1: Basic traffic scenario

Class Type	**No. commodities**	**Bandwidth** (kbps)	**Traffic load** (% of network capacity)
CT5	160	[5÷10000]	1.1428
CT4	54	4000	1.5428
CT3	441	20	0.063
CT2	162	0.1	0.0001
CT1	441	1000	3.15
CT0	441	0.1	0.0003
Total	**1699**		**5.8991**

Table 2: Impact of traffic load on link utilization and LSP length

	Link bandwidth utilization		**LSP length**
C_{global}	M_B	V_B	M_P
1	0.0590	0.0010	3.0127
8	0.4719	0.0696	3.4565
9	0.5280	0.0765	3.3970
12	0.6375	0.0860	3.7325
16	0.8866	0.1115	3.7746
17	0.8991	0.1138	3.7906

real time gaming; CT2 - on-line transactions; CT1 - streaming applications; CT0 - best effort traffic [17]. Real traffic patterns have been processed to compute the traffic matrix. We have obtained a basic light load scenario presented in Table 1. The following delay requirements have been considered: $D_k = [1 \div 18]$ ms for CT5, $D_k = [1 \div 40]$ ms for CT4, $D_k = [1 \div 12]$ ms for CT2 and $D_k = [1 \div 6]$ ms for all other commodities.

Multiple tests have been performed by varying both traffic load and the values of parameters affecting the model. In particular, we study the impact of variations in (i) the overall traffic, the (ii) traffic of a particular CT, and (iii) the traffic coming from a specific LER, by respectively multiplying the basic amount of traffic by coefficients C_{global}, C_{CTx} and C_{LERy}. Furthermore, we vary the number of paths N_p that are included in the first restricted problem to begin the column generation procedure. We remark that this initial set may deeply influence the speed of the algorithm. The Constrained Shortest Path Problem that finds new paths to be added to the problem is solved by means of the algorithm presented in [14], based on the use of Lagrangian Relaxation.

In order to evaluate the performance of our approach, we adopt the following indicators:

1. satisfaction ratio, i.e. the ratio between the number of satisfied com-

modities and the total number of commodities. We consider two indicators S_{global} and S_{CTx}, respectively: the first concerns all the commodities whereas the latter refers to the commodities of class CT x;

2. mean (M_B) and variance (V_B) of bandwidth utilization of the links;
3. average length (expressed by the number of hops) M_P of LSPs, used to satisfy the commodities;
4. processing time.

Figure 1 shows the behaviour of indicators S_{global} and S_{CTx} when coefficient C_{global} increases from 1 (basic traffic scenario) to 17 with a step equal to 1. Five paths for each commodity are included in the initial set ($N_p = 5$). As expected, the increase in traffic load causes a reduction of coefficients S_{CTx}, that mainly concerns commodities of CT with low priority. More precisely, when C_{global} is lower than 8, all commodities are satisfied; when $C_{global} = 9$, commodities rejection starts to take place, affecting especially CT0. In the case of CT4 and CT5, S_{CTx} is lower than 1, remaining anyway higher than 0.97, only when C_{global} is higher than 16.

Table II presents the values of link bandwidth utilization and LSP length as a function of the values of C_{global}. It can be noted that, when S_{CT0} and S_{global} start to decrease, the mean of link bandwidth utilization M_B is higher than 0.5, reaching almost a 90% level for $C_{global} > 16$. The length of LSPs tends instead to be stable on the average: the mean is contained in the range (3,4).

We also evaluated the satisfaction ratios S_{global} and S_{CTx} as a function of C_{CTx} and C_{LERy} (Figure 2). Considering C_{CTx}, the results show that, while the increase in lower priority traffic does not affect the values of S_{CTx} for higher priority classes, an increase in the higher priority traffic entails a decrease in the value of S_{CTx}, that is much faster than the decrease of S_{global}. In the case of C_{LERy}, congestion thresholds further decreases since the first links in the backbone DS-TE cannot route all traffic. Note that, when C_{CT5} increases, S_{global} and S_{CT5} start to decrease when M_B equal to 0.35 and 0.587, respectively, while when C_{LER2} increases, congestion thresholds are observed when M_B is equal to 0.1009 (S_{global}) and 0.1043 (S_{CT5}).

6 Conclusions

In this paper, we have proposed a new methodology for traffic engineering in DiffServ-aware MPLS networks. In contrast to standard approaches, where preemption and rerouting of one or more connections is performed in sequence in case of bandwidth lack, we propose a single-phase approach that simultaneously takes into account both preemption and rerouting. We

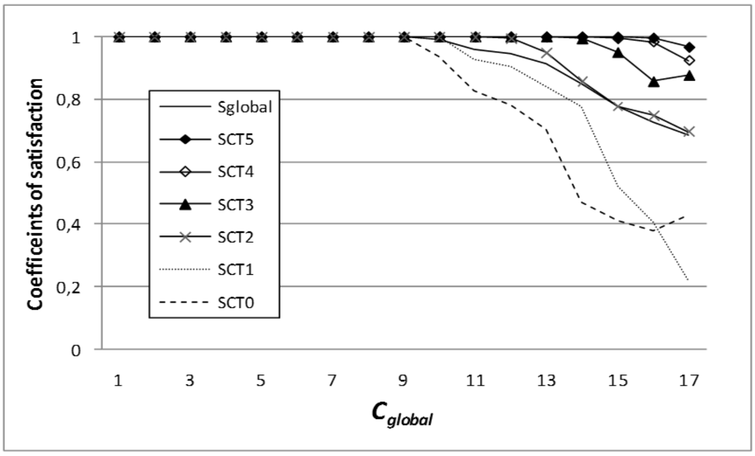


Figure 4: Coefficients of satisfaction as a function of overall traffic load.

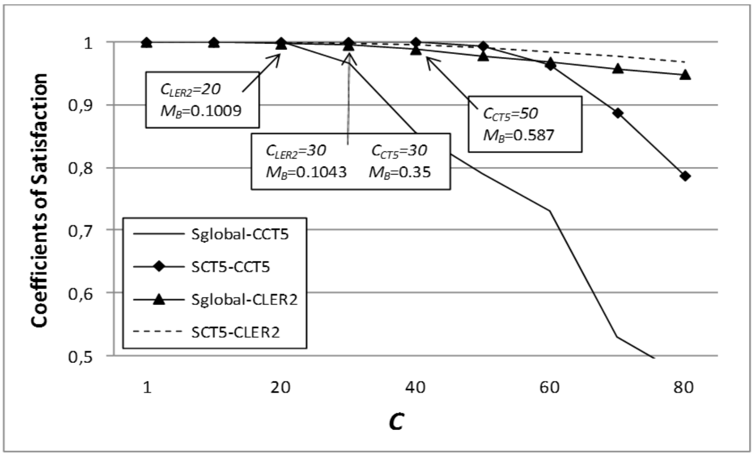


Figure 5: Coefficients of satisfaction as a function of C_{CTx} and C_{LERy}.

always find an optimal solution when enough bandwidth is available. Otherwise, we apply a heuristic post-processing procedure in order to minimize unsatisfied commodities. Tests have shown a good performance of our routing scheme in terms of the satisfaction ratio, link bandwidth utilization and LSPs length.

7 Acknowledgements

This work was partially supported by the MIUR Project IKNOS.

References

[1] D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell and J. McManus, *RFC 2702: Requirements for Traffic Engineering Over MPLS*, RFC Editor, 1999.

[2] F. Le Faucheur, *RFC 4124: Protocol Extensions for Support of Diffserv-aware MPLS Traffic Engineering*, 2005.

[3] K. Kar, M. Kodialam, T.V. Lakshman, *Minimum Interference Routing with Application to MPLS Traffic Engineering*, Proc. IEEE INFOCOM, vol. 2, pp. 884-893, Tel-Aviv, Israel, March 2000.

[4] A.B. Bagula, M. Botha, and A.E Krzesinski, *Online Traffic Engineering: The Least Interference Optimization Algorithm*, IEEE International Conference on Communications, 2004

[5] S. Suri, M. Waldvogel and P. Ramesh Warkhede *Profile-Based Routing: A New Framework for MPLS Traffic Engineering*, Quality of Future Internet Services: Second COST 263 International Workshop, Qofis 2001, Proceedings, Coimbra, Portugal, 2001.

[6] A. Juttner, B. Szviatovszki, A. Szentesi, D. Orincsay, J. Harmatos *On-demand optimization of label switched paths in MPLS network*, Computer Communications and Networks conference, 2000.

[7] J.C. de Oliveira, C. Scoglio, I.F. Akyildiz, and G. Uhl *New Preemption Policies for DiffServ-Aware Traffic Engineering to Minimize Rerouting in MPLS Networks*, IEEE/ACM Transactions on Networking, vol. 12, no. 4, 2004.

[8] D.O. Awduche, *MPLS and Traffic Engineering in IP Networks*, IEEE Communication Magazine, vol. 37, no. 12, 1999.

[9] A. Karaman, *Constraint-Based Routing in Traffic Engineering*, IEEE International Symposium on Computer Networks (ISCN'06), pp. 1-6, 2006.

[10] G. R. Malan and F. Jahanian, *An Extensible Probe Architecture for Network Protocol Performance Measurement*, Proc. ACM SIGCOMM '98, 1998.

[11] D. Bertsimas, J.H. Tsitsiklis, *Introduction to Linear Optimization*, Athena Scientifics, 1997.

[12] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, W. Weiss. *RFC2475: An Architecture for Diffferentiated Service*, RFC Editor, 1998.

[13] W.J. Cook, W.H. Cunningham, W.R. Pulleyblank, A. Schrijver *Combinatorial Optimization*, John Wiley & Sons, 1998.

[14] W.M. Carlyle, J.O. Royset, R.K. Wood, *Lagrangian Relaxation and Enumeration for Solving Constrained Shortest-Path Problems*, Networks, Vol.52, No.4, pp. 256-270, 2008.

[15] K. Holmberg, D. Yuan, *A Multicommodity Network-Flow Problem with Side Constraints on Paths Solved by Column Generation*, INFORMS Journal of Computing, Vol.15, No.1, pp.42-57, 2003.

[16] E. Rosen, A. Viswanathan, R. Callon. *RFC3031: Multiprotocol Label Switching Architecture*, RFC Editor, 2001.

[17] N. Seitz, *ITU-T QoS standards for IP-based networks*, IEEE Communication Magazine, 2003.