SREĆKO KRILE, Ph.D.
E-mail: srecko.krile@unidu.hr
University of Dubrovnik
Department of Electrical Engineering and Computing
Ćira Carića 4, HR-20000 Dubrovnik, Republic of Croatia
DRAGAN PERAKOVIĆ, Ph. D.
E-mail: dragan.perakovic@fpz.hr
VLADIMIR REMENAR
E-mail: vladimir.remenar@fpz.hr
University of Zagreb,
Faculty of Transport and Traffic Sciences
Vukelićeva 4, 10000 Zagreb, Republic of Croatia

POSSIBLE COLLISION AVOIDANCE WITH OFF-LINE ROUTE SELECTION

ABSTRACT

In DiffServ/MPLS networks traffic flows traverse the network simultaneously and there may come to collision of concurrent flows. They are distributed among LSPs (Labeled Switching Paths) related to service classes. In LSP creation the IGP (Interior Gateway Protocol) uses simple on-line routing algorithms based on the shortest path methodology. In highly loaded networks this becomes an insufficient technique. In our approach LSP need not necessarily be the shortest path solution. It can be pre-computed much earlier, possibly during the SLA (Service Level Agreement) negotiation process. In that sense an effective algorithm for collision control is developed. It may find a longer but lightly loaded path, taking care of the collision possibility. It could be a very good solution for collision avoidance and for better load-balancing purpose where links are running close to capacity. The algorithm can be significantly improved through heuristic approach. Heuristic options are compared in test-examples and their application for collision control is explained.

KEYWORDS

collision avoidance, multi-constraint route selection, self-organizing systems

1. INTRODUCTION

MPLS (Multi-Protocol Label Switching) plays a key role in the next generation multi-service IP networks by delivering QoS and TE (traffic engineering) features. Although the MPLS specifications do not directly introduce tools for managing QoS, the optimization of aggregated traffic flow through LSPs (Labeled Switching Path) forms the base for delivering QoS. MPLS allows this by means of Explicit Routed LSP Tunnels [1]. Today we apply on-line explicit routing in the moment of service invocation. The aggre-gated flow comes to LSR (Label Switching Router) and has to be routed through the domain to destination (egress router). With routing protocols such as OSPF (Open Shortest Path First), widely-used IGP protocol, some paths may become congested while others are underutilized.

Therefore, the traffic routing at the moment of service invocation can be appropriate only for under-loaded networks. Highly loaded networks need a prediction of collision probability and this has to be done long before the moment of service utilization. As firm correlation with bandwidth management and traffic engineering is required, the initial (pro-active) routing can be pre-computed off-line, possibly during SLA (Service Level Agreement) negotiation process. Such approach could be a very good solution for the collision avoidance and for better load-balancing purpose where links run close to capacity. QoS provisioning and the quantitative end-to-end guarantees have to be in firm correlation with bandwidth management [2].

This paper looks for an optimal path provisioning technique for new priority traffic (new SLA) in the context of other traffic flows (former contracted SLAs) traversing the network simultaneously; see Figure 1. Figure 2 presents a traffic situation on the path with possible collision. The collision control has to be implemented, possibly during the SLA negotiation process. Such off-line optimization technique takes care of concurrent flows that coexist at the same time in the netwok.

A detailed explanation of such approach is given in Section 2. The collision control technique can be seen as the CEP (Capacity Expansion Problem) without shortages. The mathematical model of CEP, algorithm development and its improved solution are explained in Section 3. The comparison of results for different algorithm options is given in Section 4
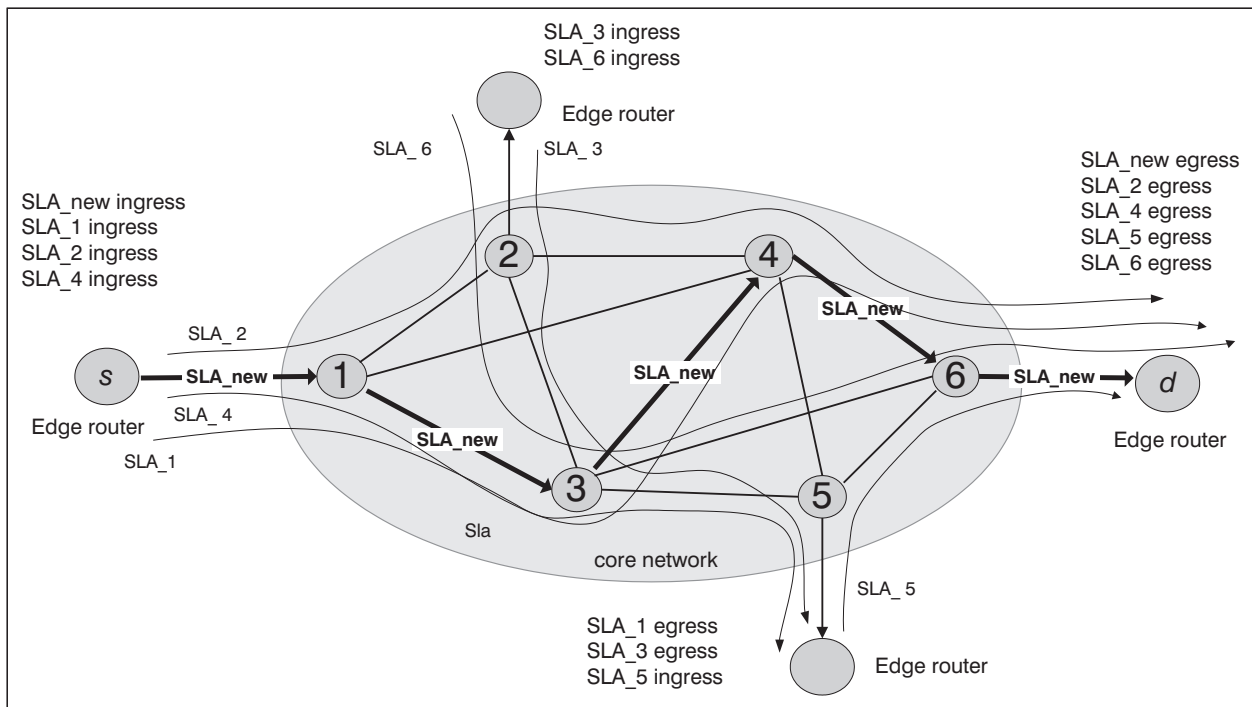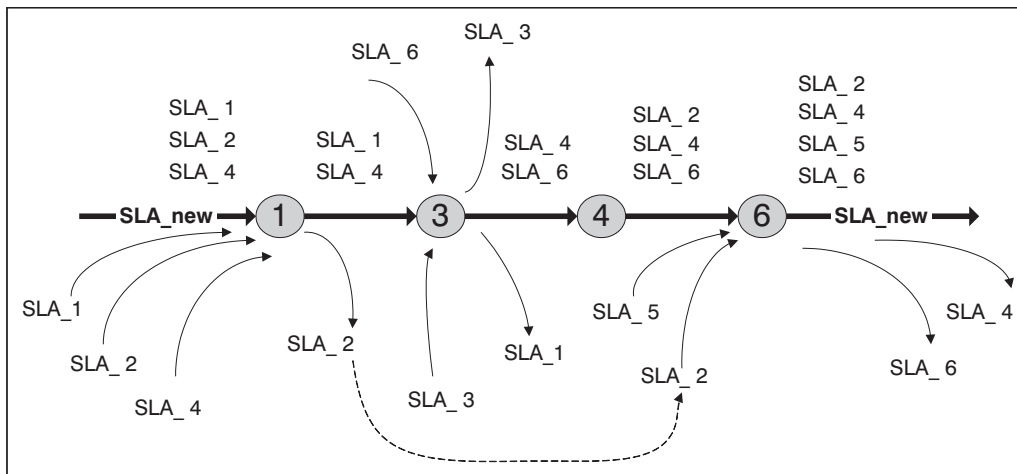
Figure 1 - A number of contracted SLAs

Figure 2 - Simultaneous flows on the path

2. QOS PROVISIONING IN MPLS/DIFFSERV NETWORKS

In DiffServ (service differentiation) paradigm the network operator can ensure the traffic prioritization, especially to quality voice (VoIP) and video calls (premium traffic), the same as for truly differentiated data services. It means that DiffServ network classifies individual flows in a small number of service classes at network edges [3]. Also, it enables "soft" reservation (allocation) of resources and special handling of packets in the core. Combining both, MPLS and DiffServ, provides a scalable QoS solution for the core of the network. The paths of the packets could be specified through their behavior in the queues of different routers.

MPLS uses TE-RSVP, extension to RSVP (Resource Reservation Protocol), to provide explicit routing, [4] and [7]. TE-RSVP is a "soft state" protocol and uses UDP or IP datagrams as the signaling mechanism for LSP setup [5]. The MPLS traffic engineering also extends the MPLS routing capabilities with support for CR (Constraint-based Routing). CR as an extension of explicit routing allows an originating (ingress) router to compute LSP to egress router (with a number of intermediate LSRs), taking care of constraints such as bandwidth and administrative policy [8].

2.1 Intra-domain Routing and Bandwidth Management

All traffic traversing simultaneously a DiffServ/ /MPLS domain is distributed among LSPs between edge routers - LER (Label Edge Router), as the resultant of the intra-domain routing protocol. The aggregated flows of packets are classified in FECs (Forwarding Equivalence Class) and can be routed in relation to class of service (CoS) bits in the packet header. Classification and filtering of the information packet happen only once, at the ingress edge. Such aggregated flow comes to the interior router (LSR) and has to be routed to destination (egress router). LSRs are capable of forwarding traffic in equivalent FEC. Packets of the same FEC are assigned the same label and generally traverse through the same path across the MPLS network. It means that one or more FECs may be mapped to a single LSP. Some of LSPs generally traverse through the same path across the MPLS network. LSPs coexist on the same link with collision probability [12].

In DiffServ networks the classification of traffic flows is performed according to the SLA signed between the customer and the network operator [13]. Each SLA specifies a time period for utilization of defined service class and very good approximation of the "expected capacity" (speed at which traffic may be sent - bandwidth).

The service provider in the domain (e. g. ISP) wants to accept a new SLA with priority traffic flow between edge routers. A *traffic trunk* is defined as a logical pipeline within an LSP, with reservation of certain amount of capacity to serve the traffic associated with a certain SLA. Thus, it is clear that LSP between an ingress/egress pair may carry multiple traffic flows associated with different SLAs. The examples from Figures 1 and 2 show that all traffic flows on the path participate possibly at the same time with collision possibilities. In that sense the network operator (e. g. ISP) has to find the optimal LSPs including new traffic flow (new SLA) but without possible collision in the core network [6]. Each traffic demand can be satisfied on the appropriate or higher QoS level (related to FEC). The main condition is: sufficient network resources must be available for the priority traffic at any moment of service duration, not only at the moment of its invocation [10]. Non-priority traffic uses the best effort class if sufficient bandwidth is available.

2.2 Collision Control during SLA Negotiation Process

During the SLA negotiation process the RM (Resource Manager) module has to determine the main parameters that characterize the required flow (i. e.,

bandwidth, QoS class, ingress and egress IP router addresses). At first the RM can apply any shortest path-based routing algorithm e. g. OSPF (Open Shortest Path First) to get initial LSP. The BB (Bandwidth Broker) will therefore check if there are enough resources on the calculated path to satisfy the requested service class, taking care of all priority flows in the same time. Some of them follow the same path to the common egress router. The correlation with other LSPs is possible and collision can occur [5].

If the optimal routing sequence finds that any link on the path exceeds the allowed capacity (maximal bandwidth) there is collision. Such congested link has to be eliminated from further calculation and the procedure starts again. Alternatively, adding capacity arrangement (short-term) is possible but it can produce significant extra cost for the service operator.

If calculation finds a path without any collision a new SLA can be realized. That flow is assigned to the related LSP and stored in the database of BB. On the contrary the new SLA cannot be accepted or must be re-negotiated. In the moment of service invocation such calculated and stored LSP can be easily distributed to the MPLS network to support explicit routing, leveraging bandwidth reservation and prioritization [11]. In this way the LSP creation should be in co-relation with the SLA creation. With such constraint--based routing technique it is obvious that the LSP need not necessarily be the shortest path solution (as in usual routing techniques today).

3. CEP FOR COLLISION CONTROL AND LOAD BALANCING PURPOSE

The collision control technique explained above can be seen as the capacity expansion problem (CEP) with or without shortages. Any transmission link is capable of serving traffic demands for N different QoS levels (service class) for $i = 1, 2, ..., N$. Each traffic load needs appropriate bandwidth amount on the path, so it looks like bandwidth expansion.

New bandwidth portion on the link can be assigned to appropriate service class up to the given limit (maximal capacity). Used capacity can be dimensioned in two forms: by expansion (or reduction) or by conversion. Expansions/reductions can be done separately for each service class or through conversion (redirected amount) to lower quality class if idle capacity exists. Under special conditions it can be reused to serve the traffic of lower quality level. Diagram from Figure 3 gives the network flow representation of such problem with multiple (N) QoS levels. Links connect M core routers (LSR) on the path. In the CEP model the following notation is used:
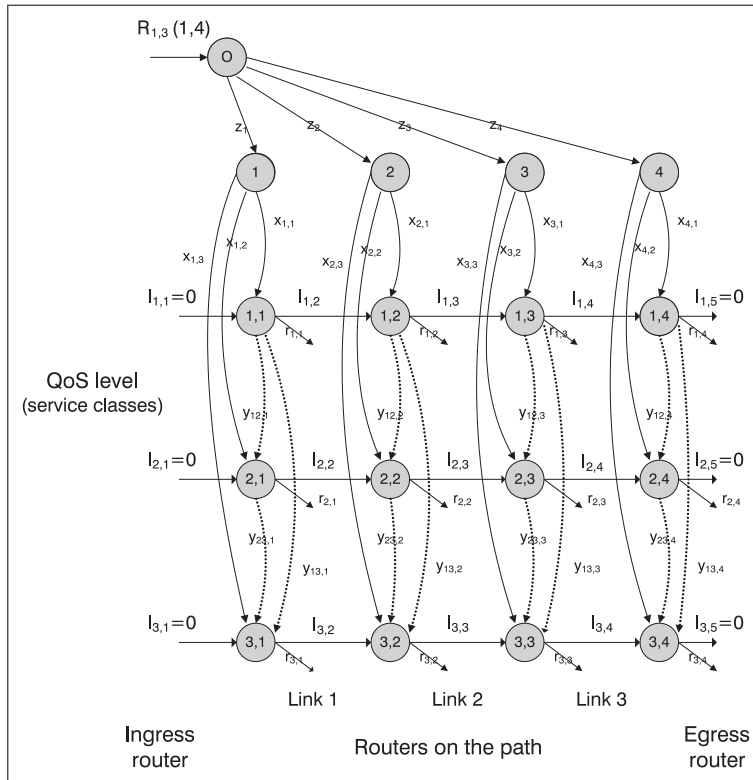
Figure 3 - The network flow representation of the CEP model

i, j and k = QoS level. We differentiate n service classes (QoS levels). The N levels are ranked from $i = 1, 2,..., N$, from higher to lower.

m = the order number of the link on the path, connecting two successive routers, $m = 1, ..., M+1$.

u, v = the order number of capacity points in the sub-problem, $1 \leq u, ..., v \leq M+1$.

$r_{i,m}$ = traffic demand increment for additional capacity. For convenience, the $r_{i,m}$ is assumed to be integer. Traffic demands can be satisfied by expansion/reduction or by converted capacity from any capacity type with higher quality level. The sum of traffic demands on the whole path and for all capacity types has to be positive or zero:

$$\sum_{i=1}^{N} \sum_{m=1}^{M} r_{i,m} \geq 0 \qquad (3.1)$$

It means that no reduction of total capacity on the path is expected, in other words, an increase of capacity toward egress router is presumed.

$I_{i,m}$ = relative amount of idle capacity on link m, connecting two neighboring routers. In relation to capacity of the previous link (neighbor) there are positive and negative values. $I_{i1} = 0$, $I_{i,M+1} = 0$; that means: no adding capacity is necessary on the links towards the edge routers. These links are not a matter of optimization.

$x_{i,m}$ = the amount of adding capacity for each service class on the link m to satisfy the given traffic demands. Possible negative values (decrease).

$L_{i,m}$ = bandwidth constraints for link capacity values on link m and for appropriate service class i ($L_{1,m}$, $L_{2,m}$, ... $L_{N,m}$).

$y_{i,j,m}$ = the amount of capacity for quality level i on link m, redirected to satisfy the traffic of lower quality level j. Traffic demand can also be satisfied by converted capacity from one capacity type to another, partially or in the total amount.

$w_{i,m}$ = weight for the link m and appropriate service class i (QoS level).

$del_{i,m}$ = delay on link m for appropriate service class i. Maximal delay on the path is denoted by DEL_i.

Instead of a nonlinear convex optimization, which can be very complicated and time-consuming (a huge number of constraints), the network optimization methodology is efficiently applied. The main reason for such approach is the possibility of discrete capacity values for a limited number of QoS classes, so the optimization process can be significantly improved. The multi-constrained QoS routing can be formulated as Minimum Cost Multi-Commodity Flow Problem (MCMCF). Such problem (NP-complete) can be easily represented by multi-commodity the single (common) source multiple destination network.

Let $G(V, E)$ denote a network topology, where V is a set of vertices/nodes, representing link capacity states and A, a set of arcs representing traffic flows between routers. Each link on the path is characterized by z-dimensional link weight vector, consisting of z-non-negative QoS weights. The number of QoS mea-

sures (e. g. bandwidth, delay) is denoted by z. In general, there is a multi-constrained problem (MCP) with multi-dimensional link weight vectors for $M+1$ links on the path $\{w_{i,m}, m \in A, i = 1, \ldots, N\}$. The constraints for capacity bounds are denoted by $L_{i,m}$ ($L_{1,m}$ $L_{2,m}$, ... $L_{N,m}$). For a non-additive measure (e. g. bandwidth) the definition of the constrained problem is to find a path from ingress to definite egress node with minimal bandwidth to satisfy all traffic demands. It is equivalent to minimal link weight along the path. The link weight (cost) is the function of used capacity: lower amount of used capacity gives lower weight. If the weight of each link corresponds to the amount of used capacity, the objective is to find the optimal routing policy that minimizes the total cost on the path.

In the context of MCP more constraints can be easily introduced, e. g. maximal delay on the path (end-to-end); [17] and [18]. As it is an additive measure it can be used as criteria to eliminate any unacceptable routing solutions from further calculation.

The definition of the multi-constrained problem is to find path P from ingress to egress node such that:

$$w(P) = \min \sum_{m=1}^{M+1} \sum_{i=1}^{N} w_{i,m}(I_{i,m}, x_{i,m}, y_{i,j,m}) \qquad (3.2)$$

where:

$$I_{i,m} \le L_{i,m} \qquad (3.3)$$

$$\sum_{m_1}^{m_2} del_{i,m} \le DEL_i \qquad (3.4)$$

satisfying condition:

max. delay of $P \le DEL_i$ $\qquad (3.5)$

for $i = 1, \ldots, N$; $m = 1, \ldots, M$

A path meeting the above conditions is said to be feasible. Note that there may be multiple feasible paths between ingress and egress node. Generalizing the concept of the capacity states for each quality level of transmission link m between LSRs in which the capacity states for each service class (QoS level) are known within the defined limits is defined as *a capacity point - α_m*.

$$\alpha_m = (I_{1,m}, I_{2,m}, \ldots, I_{N,m}) \qquad (3.6)$$

$$\alpha_1 = \alpha_{M+1} = (0, 0, \ldots, 0) \qquad (3.7)$$

Formulation (3.6) α_m denotes the vector of capacities $I_{i,m}$ for each service class on link m, and it is called a capacity point. On the flow diagrams (Figure 3) each column represents a capacity point of the link, consisting of N capacity state values (for i-th QoS level). The capacity amount labeled by i is primarily used to serve the traffic demands of that service class but it can be used to satisfy the traffic of lower QoS level j ($j > i$).

The horizontal links (branches) represent the capacity flows between two neighboring routers on the path. An unused capacity (surplus) on the path can be utilized on the next link if maintenance cost of idle ca-

pacity is not too high. On the contrary, shortages (negative values) can produce unsatisfied demands or adding capacity has to be introduced.

Formulation (3.7) implies that idle capacities or capacity shortages are not possible on the edge links (links that connect ingress and egress node with core network). It means that optimization starts and finishes with zero flows (except the new SLA flow) because all prioritized flows (previous SLAs) must be fully satisfied through the network (end-to-end).

Let C_m be the number of capacity point values at router position m (for link between core routers). Only one capacity point for the link that interior router is connected to the edge router: $C_1 = C_{M+1} = 1$. The total number of capacity points is:

$$C_p = \sum_{m=1}^{M+1} C_m \qquad (3.8)$$

The objective function for the CEP problem can be formulated as follows:

$$\min \left(\sum_{m=1}^{M+1} \left\{ \sum_{i=1}^{N} c_{i,m}(x_{i,m}) + h_{i,m}(I_{i,m+1}) + g_{i,j,}(y_{i,j,m}) \right\} \right) \qquad (3.9)$$

so that:

$$I_{i,m+1} = I_{i,m} + x_{i,m} - \sum_{j=i+1}^{N} y_{i,j,m} - r_{i,} \qquad (3.10)$$

$$I_{i,1} = L_{i,M+1} = 0 \qquad (3.11)$$

for $m = 1, 2, \ldots, M+1; i = 1, 2, \ldots, N; j = i+1, \ldots, N$.

In the objective function the total cost (weight) on the path includes some different variable costs. The most important expansion cost is denoted by $c_{i,m}$ ($x_{i,m}$). The expansion costs for each service class can be differentiated. The idle capacity cost $h_{i,m}$ ($I_{i,m+1}$) as a penalty cost to force the usage of minimum link capacity (prevention of unused/idle capacity) can be taken into account. Also, we can introduce the facility conversion cost $g_{i,j,m}$ ($y_{i,j,m}$) that can control non-effective usage of link capacity (e. g. usage of higher service class capacity instead). All cost functions are assumed to be concave and non-decreasing (reflecting economies of scale). The objective function is necessarily a non-linear cost.

3.1 Algorithm Development

Let value $d_{u,v}(\alpha_u, \alpha_{v+1})$ represent the minimum expansion cost between two capacity points of the neighboring links denoted by α_u and, α_{v+1}. The network optimization can be divided in two steps. In the first step the minimal expansion weights $d_{u,v}$ between capacity points of the neighboring links are calculated. The calculation of each weight value between any couple of capacity points is called: capacity expansion sub-problem (CES). The objective function is very

similar to relation (3.9). It requires repeated solving of a certain Single Location Expansion Problem (SLEP).

Many expansion solutions exist, depending on D_i polarity, the measure of capacity change; see details in [16]. The number of all possible $d_{u,v}$ values depends on the total number of capacity points. A lot of expansion solutions are not acceptable and they are not part of the optimal sequence. It is the key for algorithm improvement.

In the second step of network optimization, the shortest path in the network with former calculated weights between node pairs (capacity points) is looked for. The problem can be seen as the shortest path problem for an acyclic network. Then the Dijkstra's algorithm or any similar algorithm can be applied. It is obvious that the optimal routing sequence need not be the shortest path solution.

3.2 Improvement of CEP Algorithm

The key for algorithm improvement lies in the fact that extreme flow theory enables separation of these extreme flows which can be included in the optimal expansion solution from those which cannot be. The most of the computational effort is spent on computing of the sub-problem values. Any of them, if they cannot be part of the optimal sequence, are set to infinity. It can be shown that a feasible flow in the network given in Figure 3 corresponds to an extreme point solution of CEP if and only it is not part of any cycle (loop) with positive flows, in which all flows satisfy the given properties [9]. One may observe that the absence of cycles with positive flows implies that each node has at most one incoming flow from the source node. This result holds for all the single source networks. This means that the optimal solution of $d_{u,v}$ has at most one expansion (or reduction) for each facility (service class). Using a network flow theory the properties of extreme point solution are identified. These properties are used to develop an efficient search for the link costs $d_{u,v}$.

4. TESTING RESULTS AND COMPARISON OF DIFFERENT ALGORITHM OPTIONS

The proposed algorithm is tested on many numerical test-examples, looking for optimal routing sequence on the path and their minimal cost. Between edge routers there are maximum M core routers (LSR) and the path consists of maximum $M+1$ links. The traffic demands (former contracted SLAs) are given in the relative amount (increments) for each interior router on the path successively. The demands overlap in time and are defined for each capacity type

(service class); see Figure 6. The results obtained by the improved algorithm are compared with the results obtained by referent algorithm that calculates all the possible expansion solutions for each CES. The number of CES is the measure of the complexity for the CEP-problem.
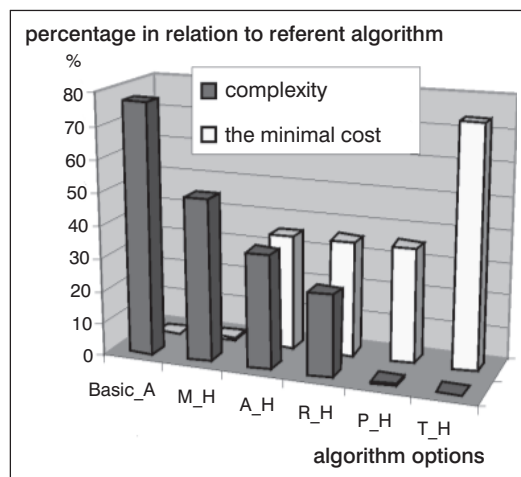
Figure 4 - Comparison of algorithm options
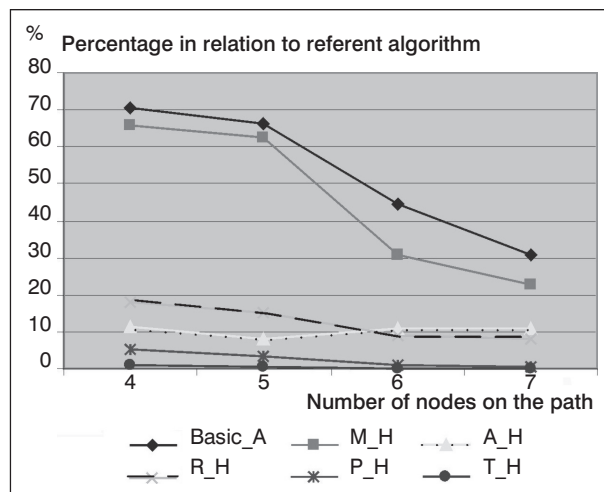
Figure 5 - Algorithm complexity influenced by value *M*

For each test-example the total number of capacity points is known, as a resultant of traffic demands. In real application the definite granularity of capacity values through discrete values of traffic demands (only integer values for $r_{i,m}$) can be applied. The lowest step of the possible capacity change ($step\_I_i$) defines value C_p and has strong influence on the algorithm complexity. So CEP requires the computation effort of $O(N^2 M C_p^2)$ with quadratic influence of N and C_p, but a linear one *of M*.

For all numerical test-examples the improved algorithm (denoted with Basic_A) can obtain the best possible result (near-optimal expansion sequence), same as referent algorithm (with full number of CES). Also, for each test-example calculated with Basic_A

the number of acceptable sub-problems can be checked, satisfying the basic and additional properties
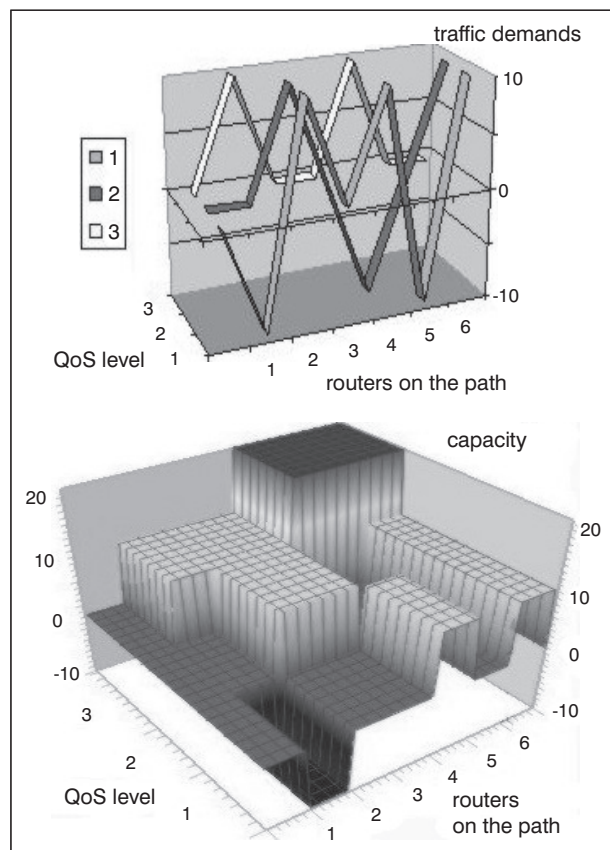
Figure 6 - Optimal usage of the capacity
in relation to given traffic demands

of the optimal flow. For $N=3$ and $M=6$ the savings in percents are on average near 20% that is proportionally reflected on the computation time savings; see Figure 4.

In real situation some limitations on the capacity state values can be introduced, considering heuristic algorithm options:

a) Only one negative capacity value in the capacity point. Such option is denoted by M_H (*Minimal-shortage Heuristic option*);

b) Total sum of the link capacity values (for all quality levels) is positive A_H (*Acceptable Heuristic option*);

c) Total sum is positive but only one value can be negative. Such option is denoted by R_H (*Real Heuristic option*);

d) Algorithm option that allows only non-negative capacity state values is denoted by P_H (*Positive Heuristic option*);

e) Only null capacity values are allowed. A trivial heuristic option (denoted with T_H) allows only zero values in capacity point (only one capacity point).

The efficiency of algorithm in the above mentioned options are compared. Figure 4 shows the average values of results for test-examples with $N=3$ and $M=6$. Also, their complexity is compared. Only for a few test-examples all the algorithm options can find the best expansion sequence, providing the minimal cost no matter which algorithm option is used; see the test-example from Table 1 and Figure 6. For the most

Table 1 - Results of numerical test-example from Figure 6

Traffic demands (increment)				Algorithm option	The best result found	Number of capacity points	Number of acceptable sub-problems satisfying properties	Computational savings in perc. (%)
Routers on the path	$r_{1,m}$	$r_{2,m}$	$r_{3,m}$					
1	-10	0	10	Full approach	9 487,58	839	30 133	-
2	10	10	0	Basic_A	9 487,58	839	25 885	14,10 (85,90)
3	0	0	0	M_H	9 487,58	590	17 869	40,70 (59,30
4	10	-10	10	A_H	9 487,58	424	12 249	59,35 (40,65)
5	-10	0	0	R_H	9 487,58	394	11 146	53,01 (36,99)
6	10	10	0	P_H	9 487,58	91	963	96,80 (3,20)
				T_H	9 487,58	7	6	99,99 (0,01)

$c_{i,m}(x_{i,m}) = f_i{}^{m-1}(A_i+B_i x_{i,m}{}^{ai})$, $A_1=3000$, $B_1=25$, $a_1 = 0.9$, $A_2=1000$, $B_2=20$, $a_2=0.85$, $A_3=2000$, $B_3=30$, $a_3=0.95$

For negative expansions $(x_{i,m} < 0)$ $c_{i,m}(x_{i,m}) = - f_i{}^{m-1}(B_i abs(x_{i,m})^{ai})$.

$h_{i,m}(I_{i,m+1}) = f_i{}^{m-1}H_i I_{i,m+1}$, $H_i = 400$ $(I_{i,m} > 0)$ for $i = 1, 2, 3$. Shortages $(I_{i,m} < 0)$ are not allowed;

$g_{i,t}(y_{i,j,m}) = f_i{}^{m-1}G_i y_{i,j,m}$, $G_i = 100$ for $y_{i,j,m} > 0$, $i < j$. Conversions in opposite direction $(y_{i j,m} < 0)$ are not allowed;

For all positions and for all quality levels it is the same value $f_i = 0.9$. All cost values are the same regardless of the position on the path. In this test-example expansion limits $L_{i,m}$ are satisfied.

Optimal usage of the capacity (expansion sequence):

$y_{1,3,1} = 10$, $x_{1,2} = 10$, $x_{2,2} = 10$, $x_{1,4} = 20$, $y_{1,3,4} = 10$, $x_{2,4} = -10$, $x_{1,5} = -10$, $x_{1,6} = 10$, $x_{2,6} = 10$

Minimal cost: 9 487,58

test-examples the algorithm option M_H can obtain the best result with average saving of 50%. For other algorithm options the significant reduction of complexity is obvious but there is deterioration of the result. The final results are still within acceptable limits (see Figure 4). The trivial algorithm option (T_H) shows very small complexity but significant deterioration of result makes this heuristic option unacceptable for application (for the most cases).

A very good fact for all algorithm options is that the increase of algorithm complexity is not caused by value *M*. Figure 5 makes it obvious that savings are higher if the number of nodes (routers on the path) rises.

5. CONCLUSION

The proposed algorithm (with different heuristic options) can be efficiently incorporated in the optimal LSP creation. The collision control can be a part of explicit intra-domain routing in DiffServ/MPLS networks. So we can make the firm correlation with the bandwidth management and admission control. LSP creation can start much earlier, possibly during the SLA negotiation process.

The collision probabilities on the path with algorithm of very low complexity can be checked first (e. g. P_H algorithm option). It means that only if there is collision possibility we need optimization by a more complex algorithm (e. g. A_H). The most complex algorithm option (Basic_A) can yield the best possible result, so that one may be sure if collision on the path could appear or not. In case of collision the appearance of a new SLA cannot be accepted or adding capacity arrangement should be done. It means that SLA re-negotiation has to be performed and the customer has to change the service parameters: e. g. bandwidth (data speed), max. delay on the path, period of service utilization etc.

Dr. sc. **SREĆKO KRILE**
E-mail: srecko.krile@unidu.hr
Sveučilište u Dubrovniku
Odjel za elektrotehniku i računarstvo
Ćira Carića 4, 20000 Dubrovnik, Republika Hrvatska
Dr. sc. **DRAGAN PERAKOVIĆ**
E-mail: dragan.perakovic@fpz.hr
VLADIMIR REMENAR, dipl. ing.
E-mail: vladimir.remenar@fpz.hr
Sveučilište u Zagrebu, Fakultet prometnih znanosti
Vukelićeva 4, 10000 Zagreb, Republika Hrvatska

MOGUĆNOST IZBJEGAVANJA SUKOBA KROZ PRETHODNO ODREĐIVANJE PUTA

U DiffServ/MPLS mrežama moguć je sukob istovremenih tokova. Svi prometni tokovi su podijeljeni između LSP-ova (Label Switching Path) koji su pridruženi pojedinim klasama usluga. Pri osiguranju kvalitete usluga (QoS) za određivanje LSP-a danas se koriste jednostavni algoritmi usmjeravanja (kao što su OSPFS ili IS-IS), a koji se temelje na određivanju najkraćeg puta kroz mrežu u trenutku pokretanja usluge. Za preopterećene mreže takva tehnika postaje nedostatna. Zato se mogu pronaći duži ali manje opterećeni putovi, vodeći računa o mogućnostima sukoba. U našem pristupu LSP ne mora biti nužno najkraći put. On može biti proračunat mnogo ranije, možda već u trenutku pregovaranja za kreiranje SLA (Service Level Agreement), pazeći da ne dođe do sukoba simultanih (istovremenih) tokova na putu kroz mrežu. U tom smislu razvijen je efikasan algoritam za kontrolu sukoba temeljen na tehnici mrežne optimizacije. Osim za detekciju zagušenja pogodan je i za balansiranje mreže kada su prijenosni putovi blizu zagušenja. Značajna poboljšanja su moguća primjenom heurističkih metoda što je pokazano na numeričkim test-primjerima. U radu je dana usporedba dobivenih rezultata te primjena ovog algoritma i njegovih heurističkih opcija.

KLJUČNE RIJEČI

izbjegavanje sukoba, višekriterijsko određivanje puta, samo--organizirajući sustavi

LITERATURE

[1.] **Morrow, M. and Sayeed, A.**: ''*MPLS and Next-Generation Networks: Foundations for NGN and Enterprise Virtualization*'', Cisco Press, 2006

[2] **Haddadou, K., Ghamri-Doudane, & al.**: "*Designing scalable on-demand policy-based resource allocation in IP networks*", IEEE Communications Mag., Vol. 44, No. 3, 2006, pp. 142-149

[3] **Degrande, N., Van Hoey, G., La Vallee-Poussin, P. and Van Busch, S.**: "*Inter-area Traffic Engineering in a Differentiated Services Network*", Journal of Network and Systems Management (JNSM), Vol. 11, No. 4, 2003

[4] **D'Arienzo, M., Pescape, A. and Ventre, G.**: "*Dynamic Service Management in Heterogeneous Networks*", JNSM , Vol. 12, No. 3, 2004. pp. 349-370

[5] **Botaba, R., Szeto, W.** and **Iraqi, Y.**: "*DORA: Efficient Routing for MPLS Traffic Engineering*", JNSM, Vol. 10, No. 3, 2002, pp. 309-325

[6] **Lima, S., Carvalho, P. and Freitas, V.**: "*Distributed Admission Control for QoS and SLS Management*", JNSM, Vol. 12, No. 3, 2004, pp. 397-426

[7] **Le Faucheur, F. et al.**: "*Multi-protocol Label Switching (MPLS) Support of Differentiated Services*", Technical Report RFC 3270, IETF, 2002

[8] **Kagklis, D., Tsakiris, C., Liampotis, N.**: "*Quality of Service: A Mechanism for explicit activation of IP Services Based on RSVP*", Journal of Electrical Engineering, Vol. 54, No. 9-10, Bratislava, 2003, pp. 250-254

[9] **Luss, H.**: "*A Heuristic for Capacity Expansion Planning with Multiple Facility Types*", Naval Res. Log. Quart., Vol. 33(04), 1986, pp. 685-701

[10] **Cheng, Yu., Farha, R., Tizghadam, A. et al.**: "*Virtual Network Approach to Scalable IP Service Deployment and Efficient Resource Management*", IEEE Comm. Mag., Vol. 43, No. 10, 2005, pp. 76-84

[11] **Giordano, S., Salsano, S., Ventre, G.**: "*Advanced QoS Provisioning in IP Networks*": The European Premium IP Projects, IEEE Comm. Mag., Vol. 41, No. 1, 2003, pp. 30-36.

[12] **Sarangan, V., Jyh-Cheng C.**: "*Comparative study of protocols for dynamic service negotiation in the next-generation Internet*", IEEE Comm. Mag., Vol. 44, No. 3, 2006, pp. 151-156

[13] **Guichard, J., Le Faucheur, F. and Vasseur, J. P.**: ''*Definitive MPLS Designs*'', Cisco Press, 2005. pp. 253- 264

[14] **Bhatnagar, S.; Ganguly, S.; Nath, B.**: "*Creating multi-point-to-point LSPs for traffic engineering*", IEEE Comm. Mag., Vol. 43., No. 1, 2005, pp. 95- 100

[15] **Zangwill, W. I.**: "*Minimum Concave Cost Flows in Certain Networks*", Mgmt. Sci., Vol. 14, 1968, pp. 429-450

[16] **Krile S., Kuzumilovic D.**, "*The Application of Bandwidth Optimization Technique in SLA Negotiation Process*", Proc. of 11th CAMAD'06 (Int. Workshop on Computer-Aided Modeling, Analysis and Design of Comm. Links and Networks), Trento, 2006, pp. 115-121.