Sub-flow assignment model of multicast flows using multiple p2mp LSPs

Fernando Solano, Ramón Fabregat
Institut d'Informàtica i Aplicacions, Universitat de Girona
Av. Lluis Santaló s/n. EPS P4, (17071) Girona, España.
{fsolanod, ramon}@eia.udg.es

and

Yezid Donoso
Departamento de Ingeniería de Sistemas y Computación, Universidad del Norte
Km. 5 vía Pto. Colombia, Barranquilla, Colombia.
ydonoso@uninorte.edu.co

Abstract

In previous work, a multi-objective traffic engineering scheme (MHDB-S model) using different distribution trees to multicast several flows is proposed. Because the flow assignment can not be mapped directly into MPLS architecture, in this paper, we propose a liner system equation to create multiple point-2-multipoint LSPs based on the optimum sub-flow values obtained with our MHDB-S model.

Keywords: Multiobjective Optimization, Multicast, MPLS, Sub-flow assignment

Resumen

En trabajos previos, se ha propuesto un esquema de ingeniería de tráfico multiobjetivo (modelo MHDB-S) para realizar multicast de diversos flujos, usando diferentes árboles de distribución. Como la asignación de flujos no puede ser mapeada directamente en la arquitectura MPLS, se propone un sistema de ecuaciones lineales para crear múltiples LSPs punto-multipunto basándonos en los valores de subflujo óptimos obtenidos con nuestro modelo MHDB-S.

Palabras claves: Optimización multiobjetivo, Multicast, MPLS, Asignación de subflujos.

1.Introduction

Traffic engineering is concerned with optimizing the performance of operational networks. The main objective is to reduce congestion in hot spots and to improve resource utilization. This can be achieved by setting up explicit routes through the physical network in such way that traffic distribution is balanced across several traffic trunks [1]. Current configurations in computer networks provide an opportunity for dispersing traffic over multiple paths to decrease congestion and achieve the aggregated end-to-end bandwidth requirement.

This load balancing technique can be achieved by a multicommodity network flow formulation [2], [3] and [4], which leads to the traffic being shared over multiple routes between the ingress node and the egress nodes in order to avoid link saturation and hence the possibility of congestion. Several advantages of using multipath routing are discussed in [5]: links do not get overused and therefore do not get congested, and that multipath has the potential to aggregate bandwidth allowing a network to support more data transfer than it is possible with any one path, etc.

In previous work [6] we proposed a multi-objective traffic engineering scheme (MHDB-S model) to multicast several flows. The aim of [6] is to combine the following weighting objectives into a single aggregated metric: the maximum link utilization, the hop count, the total bandwidth consumption, and the total end-to-end delay. Moreover, our proposal solves the traffic split ratio for multiple trees.

In unicast transmission, the split ratio is fed to the routers which divide the traffic of the same pair of ingress-egress nodes into multiple paths i.e. each flow is split into multiple sub-flows. In multicast transmission, the load balancing consists of traffic being split (using the multipath approach) across multiple trees, between the ingress node and the set of egress nodes.

The proposed MHDB-S model can be applied to MPLS networks by allowing multiple explicit trees to be established in order to transport several multicast flows (fig. 1). With this load balancing technique, each flow is split between multiple trees [7] depending on the solution obtained.

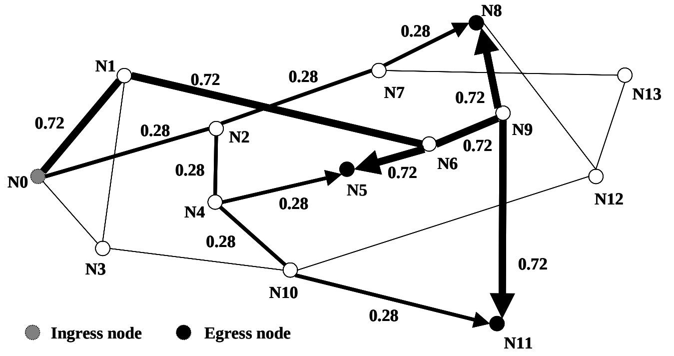


Fig. 1. Flow from ingress node N0 to egress nodes {N5, N8, N11} is split into two sub-flows, and each one is sent along different trees: {(0,1), (1,6), (6,5), (6,9), (9,8), (9,11)} and {(0,2), (2,7), (7,8), (2,4), (4,5), (4,10), (10,11)}. The sub-flow fraction along each tree is 0.72 and 0.28 respectively. Note that the total flow coming from each egress node is 1.

In this paper, we focus on the specific problem of sub-flow assignment. The aim of this is to obtain an efficient solution to formulate p2mp LSPs given a set of optimum sub-flow values.

The rest of this paper is organized as follows. In section 2, we describe some related studies. In section 3, we explain the multi-objective scheme for static multicast routing (MHDB-S model) [6] that multicast several flows and solve the traffic split ratio for multicast trees. The sub-flow assignment problem is analyzed in section 4. In section 5, we propose a linear equation system for creating multiple p2mp LSPs based on the optimum sub-flow values obtained with the MHDB-S model. The problem related to the lack of labels in MLS networks is analyzed in section 6. Finally, in section 7, we give our conclusions and suggestions for further study.

2.Related work

2.1.Multipath routing: splitting flows

Several papers, [8] [9].[10].[11] and [12], address the splitting multipath problem of unicast traffic, motivated by its importance in complete traffic engineering solutions. Traffic splitting is executed for every packet in the packet-forwarding path. A simple method to partition the input traffic is on a per-packet basis, for example in a round-robin fashion. However, this method suffers from the possibility of excessive packet reordering and is not recommended in practice.

[13] tries to balance the load among multiple LSPs according to the loading for each path. In MPLS networks [14] multiple paths can be used to forward packets belonging to the same "forwarding equivalent class" (FEC) by explicit routing. The distribution of the load in a set of alternate paths is determined by the amount of number space in a hash computation allocated to each path. Effective use of load balancing requires good traffic distribution schemes. In [15], the performance of several hashing schemes for distributing traffic over multiple links while preserving the order of packets within a flow is studied. Although hashing-based load balancing schemes have been proposed in the past, [15] is the first comprehensive study of their performance using real traffic traces.

In [8], Rost and Balakrishnan propose a multi-path transmission between sources and destinations. The current configurations in computer networks provide an opportunity for dispersing traffic over multiple paths to decrease congestion. In [8] dispersion involves (1) splitting, and (2) forwarding the resulting portions of aggregate traffic along alternate paths. The authors concentrate on (1): methods that allow a network node to subdivide aggregate traffic, and they offer a number of traffic splitting policies which divide traffic aggregates according to the desired fractions of the aggregate rate. Their methods are based on semi-consistent hashing of packets to hash regions as well as prefix-based classification.

2.2. Support of multicasting in MPLS networks

In MPLS, unicast and multicast packets have already been assigned to different type code in the link-layer header. Therefore, MPLS routers know whether a packet is from a unicast or a multicast flow. In the case of unicast forwarding the event of an incoming flow leads to the forwarding of exactly one flow. The packet duplication mechanism that is implemented in IP routers to support the IP multicast can be used to duplicate MPLS packets. MPLS routers at the bifurcation of a multicast routing tree duplicate packets and send copies of the same packet on different outgoing links. Although MPLS natively supports multicasting in its design, the MPLS community has focused its efforts mainly on the label switching of unicast IP traffic, leaving the sections on multicasting in the main MPLS documents ([14] and [16]) virtually empty, to be addressed in future studies. Based on this, there are some proposals for supporting multicasting in MPLS networks.

A framework for MPLS multicast traffic engineering proposed by Ooms et al. [17] gives an overview of the applications of MPLS techniques to IP multicast. Another proposal explains how to distribute labels for unidirectional multicast trees [18] and for bi-directional trees' label distribution [19].

To provide MPLS Traffic Engineering [18] for a point-to-multipoint (p2mp) application in an efficient manner in a large scale environment, p2mp TE mechanisms are required. Existing MPLS point-to-point (p2p) mechanisms have to be enhanced to support the p2mp TE LSP setup. [20] presents a set of requirements for p2mp TE extensions to MPLS.

In MPLS working group meeting in Seoul (march 1 2004) two different solution drafts ([21] and [22] for TE p2mp LSPs are presented but the chairs and the meeting strongly encourage the authors for both need to get together and converge on a single solution. The computation of p2mp TE paths is implementation dependent and is beyond the scope of those solutions. Path information can be computed by some off-line or on-line algorithms, e.g. the MHDB-S model presented in the next section.

[21] describes a solution for p2mp TE which extends [23] and [24] in order to establish, maintain, and teardown a p2mp TE LSP. In this case, a p2mp TE LSP is established by setting up multiple standard p2p TE LSPs from a sender node and all the downstream branch nodes along the p2p TE LSP to one of the leaf nodes of the p2mp TE LSP. The calculation for a p2mp requires three major pieces of information. The first is the route from the ingress LSR of a p2mp path to each of the egress LSRs, the second is the traffic engineering related parameters, and the third is the branch capability information.

[22] describes how RSVP-TE can be used for p2mp TE. It relies on the semantics of RSVP that RSVP-TE inherits for building a p2mp TE tree. P2p TE LSPs are set up between ingress LSR and egress LSRs. These p2p TE LSPs are appropriately merged by the network using RSVP semantics to result in a p2mp TE LSP.

Various traffic engineering solutions using programming techniques to balance loads by multiple routes have been designed and analyzed in different studies (see [6] and [25] for a detailed explanation of these proposals). It should be pointed out that several proposals can be applied to MPLS networks. In [6], we show that the multi-objective model produces a better result than various one-objective models. In [26], we present an enhanced model (MHDB-D) for multicasting dynamic groups, and in [25] and [27] we present two heuristics algorithms to solve the previous models.

2.3. The lack of labels problem.

A general problem of supporting multicasting in MPLS networks is the lack of labels. The MPLS architecture allows aggregation. Aggregation reduces the number of labels that are needed to handle a particular set of flows, and may also reduce the amount of label distribution control traffic needed [14]. Addition of new LSPs increases the label space and hence the lookup delay. So reducing the number of used labels is a desirable characteristic for any algorithm that adds LSPs to flows.

As pointed out in [14], the label based forwarding mechanism of MPLS can also be used to route along **mp2p** LSPs. In [28] and [29], aggregation algorithms that merge p2p LSPs into a minimal number of mp2p LSPs are considered. In this case, labels assigned to different incoming links are merged into one label assigned to an outgoing link. If two p2p LSPs follow the same path from an intermediate node to the egress node, these aggregation algorithms allocate the same label to the two p2p LSPs and thus reduce the number of used labels. In [30], an algorithm reducing the number of MPLS labels to N (number of nodes) + M (number of links) without increasing any link load is presented. For differentiated services with K traffic classes with different load constraints, their bound increases to K(N+M). Their stack-depth is only one, justifying implementations of MPLS with limited stack-depth.

The label stack was introduced into MPLS framework to allow multiple LSPs to be aggregated into a single LSP tunnel [14]. In [31], a comprehensive study of label size versus stack depth trade-off for MPLS routing protocols on lines and trees is undertaken. They show that in addition to LSP tunneling, label stacks can also be used to dramatically reduce the number of labels required for setting up LSPs in a network. Their protocols have numerous practical applications that include implementation of multicast trees, and virtual private networks using MPLS as the underlying signaling mechanism.

To reduce the number of used labels for multicast traffic, another label aggregation algorithm is presented in [32]. In this case, if two **p2mp** LSPs follow the same tree from an ingress node to the egress node set, the aggregation algorithm allocates the same labels to the two p2mp LSPs. Ingress nodes have a new table (named Tree Node Table) saving node information of the p2mp LSP and label allocation is executed by using this table.

Aggregated multicast is a scheme to reduce multicast state [33]. The key idea is that, instead of constructing a tree for each flow, there can be multiple multicast flows share a single aggregated tree to reduce multicast state and, hence, tree maintenance overhead and the number of used labels. Data packets from different flows are multiplexed in the same distribution tree, called aggregated tree. Each data packet of each group is encapsulated and travels through the aggregated tree.

3. Optimization model

The network is modeled as a directed graph $G=(N,E)$, where *N* is the set of nodes and *E* is the set of links. The set of links is $E \subseteq N \times N$. We use *n* to denote the number of network nodes, i.e., $n=|N|$. Among the nodes, we have a source $s \in N$ (ingress node) and some destinations T (the set of egress nodes). Let be $t \in T$ any egress node. Let $(i,j) \in E$ be the link from node *i* to node *j*. Let $f \in F$ be any multicast flow, where *F* is the flow set and T_f is the egress node subset to the multicast flow *f*. We use |*F*| to denote the number of flows. Note that $T = \cup T_f$.

Let X_{ij}^{tf} be the fraction of flow *f* to egress node *t* assigned to link *(i,j)*; note that these variables include the egress node *t*. Including the egress node variables allows us to control the bandwidth consumption in each link with the destination of the set of egress nodes. Therefore, it is possible to maintain the constraint of flow equilibrium to the intermediate nodes exactly. The problem solution, X_{ij}^{tf} variables, provides optimum flow values.

Let c_{ij} be the capacity of each link *(i,j)*. Let bw_f be the traffic demand of a flow *f* from the ingress node *s* to T_f. The binary variables Y_{ij}^{tf} represent whether link *(i,j)* is used (1) or not (0) for the multicast tree rooted at the ingress node *s* and reaching the egress node subset T_f. Let v_{ij} be the propagation delay of link *(i,j)*. Let *m* be the number of variables in the multi-objective function. Let $connection_{ij}$ be the indicator of whether there is a link between nodes *i* and *j*.

The problem of minimizing |*F*| multicast flows from ingress node *s* to the egress nodes of each subset T_f is formulated as follows:

Minimize

$$r_1.\alpha + r_2 \sum_{f\in F} \sum_{t\in T_f} \sum_{(i,j)\in E} Y_{ij}^{tf} + r_3 \sum_{f\in F} \sum_{(i,j)\in E} bw_f \max_{t\in T_f}\left(X_{ij}^{tf}\right) + r_4 \sum_{f\in F} \sum_{t\in T_f} \sum_{(i,j)\in E} v_{ij} Y_{ij}^{tf} \quad \text{(MHDB-S model)} \qquad (1)$$

Subject to

$$\sum_{(i,j)\in E} X_{ij}^{tf} - \sum_{(j,i)\in E} X_{ji}^{tf} = 1 \quad ,t\in T_f, f\in F, i=s \qquad (2)$$

$$\sum_{(i,j)\in E} X_{ij}^{tf} - \sum_{(j,i)\in E} X_{ji}^{tf} = -1 \quad ,i,t\in T_f, f\in F \qquad (3)$$

$$\sum_{(i,j)\in E} X_{ij}^{tf} - \sum_{(j,i)\in E} X_{ji}^{tf} = 0 \quad ,t\in T_f, f\in F, i\neq s, i\notin T_f \qquad (4)$$

$$\sum_{f\in F} bw_f.\max_{t\in T_f}(X_{ij}^{tf}) \le c_{ij}.\alpha \quad ,\alpha\ge 0,(i,j)\in E \qquad (5)$$

$$\sum_{j\in N}\sum_{t\in T_f} Y_{ij}^{tf} \le \left\lceil \frac{bw_f}{\left\lfloor \sum_{j\in N} c_{ij} \Big/ \sum_{j\in N} connection_{ij} \right\rfloor} \right\rceil \quad ,i\in N, f\in F \qquad (6)$$

where

$$X_{ij}^{tf} \in \Re, 0 \le X_{ij}^{tf} \le 1 \qquad (7)$$

$$Y_{ij}^{tf} = \left\lceil X_{ij}^{tf} \right\rceil = \begin{cases} 0 & ,X_{ij}^{tf} = 0 \\ 1 & ,0 < X_{ij}^{tf} \le 1 \end{cases} \qquad (8)$$

$$\sum_{i=1}^{m} r_i = 1, \quad r_i \in \Re, \quad r_i \ge 0, m > 0 \qquad (9)$$

The Multi-objective function (MHDB model) (1) defines a function and generates a single aggregated metric through a combination of weighting objectives. The main objective consists in minimizing the maximum link utilization (MLU), which is represented as α in equation (1). In this case, the solution obtained may report long routes. In order to eliminate these routes and to minimize hop count (HC), the term $\sum_{f\in F} \sum_{t\in T_f} \sum_{(i,j)\in E} Y_{ij}^{tf}$ is added. In order to minimize the total bandwidth consumption (BC) over all links, the term $\sum_{f\in F} \sum_{t(i,j)\in E} bw_f \max_{t\in T_f}\left(X_{ij}^{tf}\right)$ is also added. This is included so that, if there is more than one solution with the best maximum link utilization, the solution with the minimum resource utilization is chosen. Though several sub-flows of the flow *f* in the link *(i,j)* with destinations to different egress nodes are sent, in multicast IP specification just one sub-flow will be sent, that is, only the maximum value of X_{ij}^{tf} for $t\in T_f$ needs to be considered. Furthermore, in order to minimize the total end-to-end propagation delay (DL) over all the links, the term $\sum_{f\in F} \sum_{t\in T_f} \sum_{(i,j)\in E} v_{ij} Y_{ij}^{tf}$ is also added.

Constraints (2), (3) and (4) are flow conservation constraints. Constraint (2) ensures that the total flow emerging from ingress node to any egress node *t* at flow *f* is 1. Constraint (3) ensures that the total flow coming from an egress node *t* at flow *f* is 1. Constraint (4) ensures that for any intermediate node different from the ingress node (i ≠ s) and egress nodes $(i\notin T)$, the sum of their output flows to the egress node *t* minus the input flows with destination egress node *t* at flow *f* is 0.

Constraint (5) is the maximum link utilization constraint. In a unicast connection, the total amount of bandwidth consumed by all the flows with the destination of egress node *t* must not exceed the maximum utilization (α) per link capacity c_{ij}, that is, $\sum_{f\in F} bw_f \sum_{t\in T} X_{ij}^{tf} \le c_{ij}\mathrm{A},(i,j)\in E$. Nevertheless, in constraint (5) only the maximum value of X_{ij}^{tf} for $t\in T_f$ needs to be considered.

Constraint (6) limits the maximum number of sub-flows (MSF) in each node by means of the capacity of each link and the traffic demand. This formulation represents the amount of necessary links for a particular traffic demand. Without this constraint, the model could suffer from scalability problems, i.e. the label space used by LSPs would be too high.

Expression (7) shows that the X_{ij}^{tf} variables must be real numbers between 0 and 1. These variables form multiple trees transport multicast flow. The demand between the ingress node and the egress node *t* may be split over multiple routes. When the problem is solved without load balancing, this variable will only be able to take values 0 and 1, which will show, respectively, whether or not the link *(i,j)* is used to carry information to egress node *t*.

Expression (8) calculates Y_{ij}^{tf} as a function of X_{ij}^{tf} .

Finally, expression (9) shows that the weighting coefficients, r_i, assigned to the objectives are normalized. These values are calculated by solving the optimization problem.

4. Sub-flow Assignment to p2mp LSPs problem

In this section, we detail the problem of creating multiple p2mp LSPs based on the optimum sub-flow values X_{ij}^{tf} obtained with solutions to MHDB-S model (1). Remember that X_{ij}^{tf} is the fraction of flow *f* to destination node *t* assigned to link *(i,j)*. First at all, because the following system applies only to one flow *f*, the index *f* will be omitted when it does not cause confusions. It means that, for example, X_{ij}^{t} is the subset of links in X_{ij}^{tf} that transmits the flow *f*.

To explain the problem, the MHDB-S model have been applied to the topology of fig. 2, with a single flow *f*, where s=N1 and T={ N5, N6}. In this case, a possible sub-flow solution (X_{ij}^{t}) obtained is shown in fig. 3. The simplest solution (fig. 4), to create LSPs based on the optimum sub-flow values, is to send each sub-flow (0,4 and 0,6 fraction) to the group separately, and in this case each sub-flow is assigned to one p2mp LPS. In fig. 4, each packet represents a 0,2 fraction of the flow. With this assignment, sub-flows X_{12}^{5} and X_{12}^{6} are different and the maximum link utilization constraint (5) could be violated. Moreover, the network is inefficiently used because multicast node capabilities are not considered. Only ingress node multicast capabilities are considered when applying the multipath approach, which permits that the flow is balanced across several links.

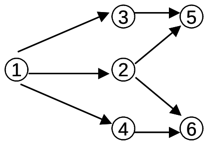


Fig. 2. Physical network topology.

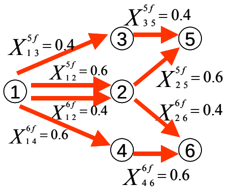


Fig. 3. MDDB-S solution

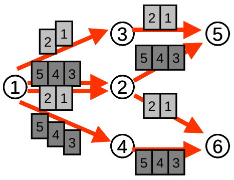


Fig. 4. Simplest p2mp assignment.

A second approach considers that one sub-flow is included in the other, i.e. min(X_{12}^{5}, X_{12}^{6}) $\subseteq$ max(X_{12}^{5}, X_{12}^{6}), in the example $X_{12}^{5} \subseteq X_{12}^{6}$. If both sub-flows X_{12}^{5} and X_{12}^{6} are sent over the link (1,2) to each member of the group separately (fig. 5), a part of the same flow is being transmitted over the same link and the network is also inefficiently used.

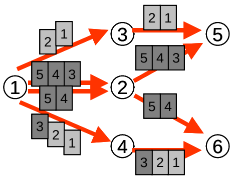


Fig. 5. p2mp assignment: unicast transmission

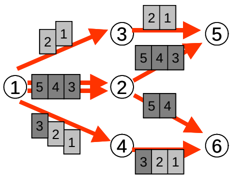


Fig. 6. p2mp assignment: multicast transmission

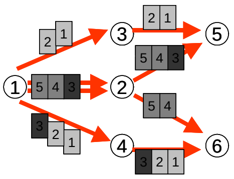


Fig. 7. p2mp assignment: sub-flow assignment

Moreover, if a node has multicast capabilities, it is not necessary to transmit all the sub-flows over the link. In particular, if N2 has multicast capabilities, only the max (X_{12}^5, X_{12}^6) must be transmitted over link (1,2) (fig. 6). However, this solution presents a problem in the forwarding mechanism. Some incoming packets at node 2 must be forwarded exactly once (packet 3), but other packets of the same sub-flow must be forwarded by different output links (packets 4 and 5). To solve this, the ingress node must split this sub-flow in several sub-flows (fig. 7).

5. Sub-flow assignment based on a linear equation system

In this section, a linear system equation to split a sub-flow in several sub-flows is proposed. Solution is the set of desired multicast trees for the set of X_{ij}^{tf} values. For a unique flow *f* and given values X_{ij}^t , the δ solution of the presented system is a set of encoded *p2mp* LSPs, in which each, *p2mp*$_k$, sends a fixed fraction, c_{min} , of the whole bandwidth to all destinations in T_f . Let $\delta_{ij}^{p2mp_k}$ be a natural number, possibly 0, indicating the number of destinations that link (i,j) at *p2mp*$_k$ broadcasts.

To compute the c_{min} value, it should be taken into consideration that a low value can result in many equation with many equal p2mp LSPs and, in contrast, a high value can result in a unsolvable problem because some fractions of X_{ij}^t could not be assigned to p2mp LSP. In the other hand, it can be seen that the set of found p2mp LSPs, and thus δ can be regarded as a linear combination of X_{ij}^t values. Therefore, an optimal c_{min} value must divide all X_{ij}^t and difference among them, hence $c_{min}=m.c.d.^*(X_{ij}^t)$, where $m.c.d.^*$ is the maximum common divisor operator used with real numbers between 0 and 1. In the example of figure 3, the c_{min} value is 20%.

The following three equation sets model p2mp LSPs in general.

$$\forall k, \forall j \ \sum_{j \in N} \delta_{sj}^{p2mp_k} = |T_f| \tag{10}$$

$$\forall k, \forall (i,j) \in E, \forall m \ \delta_{ij}^{p2mp_k} = \begin{cases} \sum_{m \in N} \delta_{jm}^{p2mp_k} + 1, & if \ j \in T_f \\ \sum_{m \in N} \delta_{jm}^{p2mp_k}, & otherwise \end{cases} \tag{11}$$

$$\forall k, \forall t \in T_f \ \sum_{i \in N} \delta_{it}^{p2mp_k} = 1 \tag{12}$$

And for this problem in particular:

$$\forall (i,j) \in E \ c_{min} \sum_k \delta_{ij}^{p2mp_k} = \sum_{t \in T_f} X_{ij}^t \tag{13}$$

Note that, the set $\{(i,j) | \delta_{ij}^{p2mp_k} \geqslant 1\}$ for a given *k* is the set of links that conforms the tree *p2mp*$_k$.

The equation set (10) suggests that the number of reached destinations from a source node *s* is equal to $|T_f|$. This is clear because all p2mp LSPs reach exactly all destinations. Another obvious consequence is that $|p2mp|=1/c_{min}$, in other words, the number of constructed p2mp LSPs is the inverse of the fraction sent by each p2mp LSP. For the analyzed example at figure 3, the number of p2mp LSPs to be constructed is 5.

The conservation flow law seen at (2), (3) and (4) can be traduced as the set regarded on (11). It means that the amount of destinations a node *j* must forward packets to is the same amount after (i.e. *i* node) and before (i.e. *m* nodes), or one less if j is a destination.

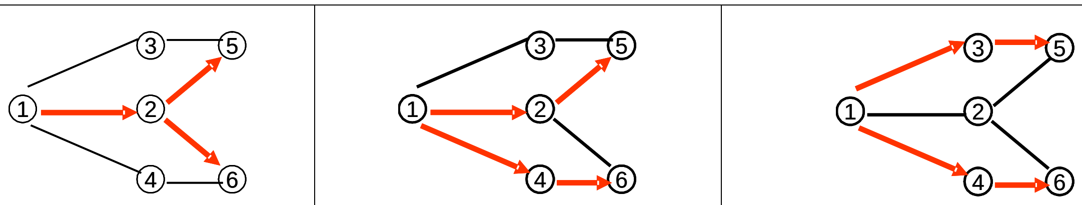
Inside a p2mp LSP, no destination node can receive flow information by two links at the same time. This can be expressed with (12) where *t* is a leaf node in the p2mp LSPs, i.e. it does not reroute traffic to another point in the network.

By looking at a single link *(i,j)*, the total bandwidth consumed in a link for a single destination should be equal to the amount consumed by all p2mp LSPs to that destination. In general, this holds for all destinations at the same time by (13).

For a given set of X_{ij}^{t} values, the δ solution set of the presented system is a set of encoded p2mp LSPs, which are conformed by those δ values greater or equal to 1. Note that the equation system can resolve into equal p2mp LSPs. If two p2mp LSPs A and B are the same (i.e. contains the same links, no more, no less), these p2mp LSPs can be merged by adding their fractions. Solving the example described on figure 3, the solution (in table 1) shows that 2 pairs of p2mp LSPs (1 $\wedge$ 4 and 3 $\wedge$ 5) can be merged. Therefore, we have a set of 3 p2mp LSPs in our solution (figure 8); one transmitting 20% of the total bandwidth and two transmitting 40% of it.

$\{\delta_{12}^{1\wedge4}=2,\delta_{25}^{1\wedge4}=1,\delta_{26}^{1\wedge4}=1\}$	$\{\delta_{12}^{2}=1,\delta_{25}^{2}=1,\delta_{14}^{2}=1,\delta_{46}^{2}=1\}$	$\{\delta_{13}^{3\wedge5}=1,\delta_{35}^{3\wedge5}=1,\delta_{14}^{3\wedge5}=1,,\delta_{46}^{3\wedge5}=1\}$

Table 1: δ solution. Super indexes $a \wedge b$ indicates that the values were assigned initially to two p2mp LSPs (a and b).



a. First tree. (40%)

b. Second tree. (20%)

c. Third tree. (40%)

Fig. 8. The p2mp LSPs.

We analyze the performance of the sub-flow assignment algorithm when sub-flow assignment and merging LSPs is considered. Over the 14-node NSF (National Science Foundation) network (fig. 1), two flows with the same source, s=N0, are transmitted. The egress nodes subsets are T1={N5, N8, N11} and T2={N8, N11, N13} respectively. The transmission rates are 256 Kbps, 512 Kbps, 1 Mbps, 1.5 Mbps, 2 Mbps and 2.5 Mbps for each flow.

		0.25	**0.5**	**1**	**1.5**	**2**	**2.5**
A	**MHDB-S model:** (multicast + multipath)	p2mp LSPs are not considered (see section 4)					
B	**MHDB-S +** **Sub-flow assignment**	c_{min} = 33 % f1 = 3 f2 = 3			c_{min} = 10 % f1=10 f2=10		
C	**MHDB-S +** **Sub-flow assignment +** **merging LSPs**	f1 = 2 f2 = 2			f1 = 8 f2 = 5	f1 = 9 f2 = 6	

Table 2: Total number of p2mp LPS obtained to flow 1 and flow 2

Table 2 shows the number of p2mp LSPs used when different schemes for multicasting several flows are considered. Comparing B and C, the merging LSPs scheme reduces the number of p2mp LSPs.

7. Conclusions

In this paper, considering a multiobjective traffic engineering scheme using different distribution trees to multicast several multicast flows, we propose a sub-flow assignation method based on a linear equation system to create multiple p2mp LSPs. The comparison of different routing schemes shows that the number of LSPs found with the sub-flow assignation method is more than the number of flows considered. However, the merging LSPs scheme reduces these values.

In the future we plan to demonstrate the usefulness of sub-flow assignment to p2mp LSPs with a variety of network scenarios. Despite merging LSPs reduces the number of used label, label aggregation algorithms will be also considered because they reduce even more the labels needed. Moreover, label stacking mechanism will be also analyzed to reduce more this value.

Acknowledgments

This work was partially supported by the MCyT under the project TIC2003-05567. The work of Yezid Donoso was supported by the Universidad del Norte (Colombia) under contract G01 02-MP03-CCBFPD-0001-2001. The work of Fernando Solano was supported by the Ministry of Universities, Research and Information Society (DURSI) of the Government of Catalonia under contract 2004FI-00693.

References

[1] Changhoon Kim, Yanghee Choi, Yongho Seok, Youngseok Lee. “A constrained Multipath Traffic Engineering Scheme for MPLS Networks”. ICC 2002.

[2] R. Ahuja, T. Magnanti, J. Orlin. “Network flows: Theory, algorithms and applications”. Prentice-Hall NJ USA 1993.

[3] M. Bazaraa, J. Jarvis, H. Sherali. “Linear Programming and Network Flows”. John Wiley & Sons, 2nd ed, USA, 1990.

[4] M. Bazaraa, H. Sherali, C. M. Shetty. “Nonlinear Programming, Theory and Algorithms”. John Wiley & Sons, 2nd ed, USA, 1993.

[5] J.C. Chen, S.H. Chan. “Multipath routing for video unicast over bandwidth-limited networks". GLOBECOM 2001

[6] Y. Donoso, R. Fabregat, L. Fàbrega. “Multi-Objective scheme over multi-tree routing in multicast MPLS networks”. ACM/IFIP LANC’03.

[7] Y. Wang, Z. Wang, L. Zhang. “Internet Traffic Engineering without Full Mesh Overlaying”. INFOCOM 2001.

[8] S. Rost, H. Balakrishnan. “Rate-Aware Splitting of Aggregate Traffic”. Submitted http://web mit edu/stanrost/www/research/publications html

[9] C. Cetinkaya, E. Knightly. “Opportunistic Traffic Scheduling Over Multiple Network Paths”. INFOCOM 2004.

[10] A. Sridharan; R. Guerin, C. Diot. “Achieving near-optimal traffic engineering solutions for current OSPF/IS-IS networks”. INFOCOM 2003.

[11] S. Vutukury, J. J. Garcia-Luna-Aceves. “A traffic engineering approach based on minimum-delay routing”. Ninth International Conference on Computer Communications and Networks, October 2000, pp 42 - 47.

[12] H. Cho, J. Lee, B. Kim. “Multi-path Constraint-based Routing Algorithms for MPLS Traffic Engineering”. ICC '03. Volume: 3 pp 1963 - 1967.

[13] C. Villamizar. “MPLS optimized multipath (MPLS-OMP)”. Internet draft. 1998.

[14] E. Rosen, A. Viswanathan, R. Callon. “Multiprotocol Label Switching Architecture”. RFC 3031. January 2001.

[15] Z. Cao, Z. Wang, E. Zegura. “Performance of hashing-based schemes for Internet load balancing”. INFOCOM 2000.

[16] L. Andersson, P. Doolan, N. Feldman, A. Fredette, B. Thomas, “LDP Specification”. RFC 3036. January 2001.

[17] D. Ooms, B. Sales, W. Livens, A. Acharya, F. Griffoul, F. Ansari. “Overview of IP Multicast in a Multi-Protocol Label” RFC 3353. August 2002.

[18] D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell, J. McManus. “Requirements for Traffic Engineering over MPLS”. RFC 2702. September 1999.

[19] J. Cui, J. Kim, A. Fei, M. Faloutsos, M. Gerla. “Scalable QoS multicast provisioning in Diff-Serv-Supported MPLS networks”. GLOBECOM 2002.

[20] S. Yasukawa, D. Papadimitriou, J. Ph. Vasseur, Y. Kamite, R. Aggarwal, A. Kullberg. “Requirements for point-to-multipoint extension to RSVP-TE”. draft-yasukawa-mpls-p2mp-requirement-02.txt. March 2004.

[21] S. Yasukawa, A. Kullberg, L. Berger. “Extended RSVP-TE for point-to-multipoint LSP tunnels. Draft-yasukawa-mpls-rsvp-p2mp-04.txt. February 2004.

[22] R. Aggarwal, L. Wei, G. Apostolopoulos, K. Kompella, J. Drake. “Establishing point to multipoint MPLS TE LSPs”. draft-raggarwa-mpls-p2mp-te-02.txt. January 2004.

[23] D. Awduche , L. Berger, D. Gan, T. Li, V. Srinivasan, G. Swallow. “RSVP-TE: Extensions to RSVP for LSP Tunnels”. RFC 3209. December 2001

[24] L. Berger. “Generalized Multi-Protocol Label Switching (GMPLS) Signaling Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Extensions”. January 2003

[25] Y. Donoso, R. Fabregat and J. L. Marzo. “Multi-objective optimization algorithm for multicast routing with traffic engineering”. IEEE ICN’04.

[26] Y. Donoso, R. Fabregat. J. L. Marzo. “Multi-Objective optimization scheme for dynamic multicast groups”. IEEE ISCC’04.

[27] Y. Donoso, R. Fabregat, J. L. Marzo. “Multi-Objective Optimization Model and Heuristic Algorithm for

Dynamic Multicast Routing”. IEEE & VDE Networks 2004.
[28] H. Saito, Y. Miyao, M. Yoshida. “Traffic engineering using multiple multipoint-to-point LSPs”. INFOCOM 2000.
[29] S. Bhatnagar, S. Ganguly, B. Nath. “Label space reduction in multipoint-to-point LSPs for traffic engineering”. ECUMN2002.
[30] D. Applegate, M. Thorup. “Load optimal MPLS routing with N+M labels”. INFOCOM’03.
[31] A. Gupta, A. Kumar, R. Rastogi. “Exploring the trade-off between label size and stack depth in MPLS Routing. INFOCOM’03.
[32] Y.K. Oh, D.K. Kim, H.J. Yoen, M.S. Do, J. Lee. “Scalable MPLS multicast using label aggregation in Internet broadcasting systems”. ICT'03.
[33] A. Fei, J.H. Cui, M. Gerla, M. Faloutsos. “Aggregated multicast: an approach to reduce multicast state”. Sixth Globel Internet Symposium (GI2001), Nov. 2001.