Instituto de Engenharia de Sistemas e Computadores de Coimbra
Institute of Systems Engineering and Computers
INESC - Coimbra

João Clímaco, José Craveirinha, Carlos Gomes da Silva and Lúcia Martins

A bi-criteria minimum spanning tree model for broadcasting in MPLS networks

No. 9 2008

ISSN: 1645-2631

Instituto de Engenharia de Sistemas e Computadores de Coimbra
INESC - Coimbra
Rua Antero de Quental, 199; 3000-033 Coimbra; Portugal
www.inescc.pt

A bi-criteria minimum spanning tree model for broadcasting in MPLS networks

João Clímaco(1,4), José Craveirinha(2,4), Carlos Gomes da Silva(3,4) and Lúcia Martins(2,4)
(1) Faculdade de Economia da Universidade de Coimbra
Avenida Dias da Silva, 165, 3004-512 Coimbra, Portugal
Email: jclimaco@fe.uc.pt
(2) Departamento de Engenharia Electrotécnica e Computadores
Faculdade de Ciências e Tecnologia da Universidade de Coimbra
Polo II da Universidade de Coimbra, Pinhal de Marrocos, 3030-290 Coimbra, Portugal
Email: jcrav@deec.uc.pt; lucia@deec.uc.pt
(3) Escola Superior de Tecnologia e Gestão de Leiria, Instituto Politécnico de Leiria
Morro do Lena, Alto Vieiro, 2401-951 Leiria, Portugal
Email: cgsilva@estg.ipleiria.pt
(4) Insituto de Engenharia de Sistemas e Computadores - Coimbra
Rua Antero de Quental 199, 3000-033 Coimbra, Portugal

July 31, 2008

Abstract

The MPLS platform enables the implementation of advanced multipath and multicast routing schemes. In this work we address a new bi-criteria minimum spanning tree model intended for routing broadcast messages in MPLS networks. The aim of the model is to obtain spanning trees that are compromise solutions with respect to two important traffic engineering metrics: load balancing cost and average delay bound. An exact solution to the formulated bi-criteria optimization problem is based on an algorithm previously proposed that enables the computation of the set of supported non-dominated spanning trees. An application model and a set of experiments on randomly generated Internet type topologies will also be presented.

Key-Words: minimum spanning trees, multicriteria models, communication networks, Internet

1 Introduction

Routing problems in modern multiservice communication networks involve the calculation of paths satisfying various technical constraints (usually QoS - Quality of Service - related constraints) and seeking simultaneously to "optimise" relevant metrics. Therefore there are advantages in developing multicriteria routing models in this area, which depend on the features of the network functionalities and the adopted routing framework. In particular, the MPLS platform for Internet enables the implementation of advanced routing schemes, namely explicit routes satisfying QoS requirements and is prepared for dealing with multipath routing, including multicast (connection between a node and a group of network nodes) and broadcast (connection between a node and all network nodes) connections. MPLS is a modern Internet technology based on the forwarding of packets using a specific label switching technique. Among other advanced routing mechanisms in MPLS the utilization of explicit-routes is characterized by the fact the path corresponding to the technical concept of label switched path (LSP) followed by each node-to-node packet stream of a certain type, is entirely determined by the ingress router (corresponding to the originating node).

An overview on multicriteria routing models, including multipath routing can be seen in Clímaco et al. (2007). An example of the numerous literature on the application of Steiner tree-based models to multicriteria multicast routing using meta-heuristics can be seen in Pinto and Barán (2005). Cerulli et al. (2006) mentions possible applications of minimum labelling spanning tree problems in communication networks.

In this work we address a new bi-criteria minimum spanning tree model, intended for routing broadcast messages in MPLS networks. The aim of the model is to obtain the set of spanning trees that constitute non-dominated solutions (hence possible best compromise solutions) with respect to two important traffic engineering metrics (objective functions of the model) and, in a second stage, to select one of those trees in an automated manner. The envisaged application is the broadcast of signalling information from a MPLS router to all remaining routers, for example information concerning critical parameters (such as bandwidth available in the links or failure in adjacent links) in the context of state-dependent dynamic routing methods or concerning any routing related information that has to be broadcasted from a centralized management system to all MPLS routers of a given autonomous area network. An exact resolution approach to the formulated problem, is provided by the algorithm proposed by two of the authors in Gomes da Silva and Clímaco (2007) that enables the computation of the set of supported non-dominated solutions of the bi-criteria minimum spanning tree problem, in ordered manner. The first objective function of the problem is a ‘load balancing’ cost function that is the sum of the load balancing costs associated with the arcs (edges or links) in the spanning tree, the costs being defined from a piece-wise linear function of the available bandwidth in the arcs. This metric seeks to achieve a balanced distribution of traffic throughout the whole network in order to minimise the impact of the establishment of the computed tree in all the remaining traffic flows in the network. The second objective function is the sum of estimated average delay bounds in the tree arcs assuming certain packet scheduling mechanisms in the MPLS routers (nodes of the network representation) and its aim is to minimise the average delay bound in the arcs of the tree associated with packet broadcasting throughout the network.

The paper is organized as follows. Firstly, in Section 2, we will describe the bi-criteria broadcast model and review the main features of the used bi-criteria minimum spanning tree algorithm. Secondly, in Section 3, a number of computational experiments will be performed with an application model focusing on broadcast routing in MPLS type networks, to show the effectiveness of the proposed procedure. The network topology generation platform used is the "GT-ITM Georgia Tech Internetwork Topology Models" software (http://www.cc.gatech.edu/fac/Ellen.Zegura/graphs.html) which enables the generation and analysis of a significant variety of randomly generated Internet network topologies, following certain probabilistic laws. In the developed application model the available bandwidths in the arcs are randomly generated according to predefined empirical probabilistic distributions and the delay bounds are given from a model for delay bound estimation (Rosenbaum et al., 2005) in MPLS networks with "latency-rate servers" (Stiliadis and Varma, 1998), that is network nodes that use specific packet scheduler mechanisms, dedicated to multi-service MPLS networks with QoS based routing. Finally concluding remarks and future work will be outlined.

2 The bi-criteria broadcast routing method

2.1 The bi-criteria broadcast model

The proposed bi-criteria broadcast model is envisaged as a new form of routing certain types of signalling information from a MPLS router to all other MPLS routers of a given autonomous area network. A first type of information could concern critical parameters in the context of state-dependent dynamic routing methods, that is methods where paths (corresponding to LSPs) between MPLS routers may change in time, seeking an adaptation to changing network conditions. Typical examples of such parameters are available bandwidths on the links or information on failure states in the links. A second type of signalling information could be related to the availability of new network services, a functionality usually designated as "service advertisement" or any routing related information to be broadcasted from a management system as foreseen in Awduche et al. (2002).

The first objective function of the bi-criteria broadcast model is a *load cost function*, an additive metric, such that the load balancing cost associated with an arc is a monotonically increasing function of the bandwidth occupied in the arc. This cost is modelled through a piecewise linear function of the occupied bandwidth which has been used in previous multicriteria routing models in multiservice communication networks, including MPLS networks, namely Knowles et al. (2000), Erbas and Erbas (2003) and Craveirinha et al. (2007).

The minimization of the load balancing cost aims at minimizing the negative impact on the network traffic flows (other than the flow associated with the broadcasted service) resulting from the utilization of the arcs of the spanning tree. This stems from the fact that the minimization of such cost tends to give a balanced distribution of the traffic throughout the network, hence leading to a better overall nework traffic carrying capacity.

The load balancing cost of an arc a_j with total capacity c_j and available bandwidth $0 \leq b_j \leq c_j$ is given by:

$$\phi_j = \begin{cases} \theta_j & , \quad 0 \leq \theta'_j \leq 0.5 \\ 2\theta_j - \frac{1}{2}c_j & , \quad 0.5 < \theta'_j \leq 0.6 \\ 5\theta_j - \frac{23}{10}c_j & , \quad 0.6 < \theta'_j \leq 0.7 \\ 15\theta_j - \frac{93}{10}c_j & , \quad 0.7 < \theta'_j \leq 0.8 \\ 60\theta_j - \frac{453}{10}c_j & , \quad 0.8 < \theta'_j \leq 0.9 \\ 300\theta_j - \frac{2613}{10}c_j & , \quad 0.9 < \theta'_j \leq 1 \end{cases} \tag{1}$$

where $\theta_j = c_j - b_j$ is the occupied bandwidth and θ'_j denotes the relative occupancy of a_j, $\theta'_j = \frac{\theta_j}{c_j}$.

The second objective function is an upper-bound on the average delay experienced by the packets on all arcs of the spanning tree. This bound was obtained by an adaptation of the model by Stiliadis and Varma (1998) that gives a bound on the path delay in LSPs in MPLS networks using "latency rate routers". These are routers (corresponding to nodes of the network representation) that use certain packet scheduler queuing mechanisms such as Weighted Fair Queuing. This model can be used, as shown in Rosenbaum et al. (2005), in the context of a deterministic network routing modelling approach by including the relevant parameters to characterize the traffic flows, through an aggregate traffic specification (including the maximal packet size, the sustainable and peak rates and the burst tolerance). Taking into account that the number of arcs of the spanning trees is constant ($n-1$, where n is the number of nodes) the bound given by (Eq. 1) in Rosenbaum et al. (2005) can be adapted to the present context by omitting the fixed term and the constant terms associated with the arcs, leading to the simplified expression for the delay bound, in ms, on each arc j:

$$d_j = \frac{M_j^{\max}}{c_{j \times 10^3}} + \frac{l_j}{200} \tag{2}$$

where $M_j^{\max}$ is the maximal packet size of all LSPs that use link a_j (in bits) and l_j is the link length (in Km). The maximization of the delay bound function with coefficients (Eq. 2) aims at obtaining the best spanning trees in terms of average delay bound per link.

Therefore, the following bi-criteria spanning tree optimization problem is formulated:

$$\begin{aligned} \min \Phi(T) &= \sum_{j:a_j \in T} \phi_j \\ \min D(T) &= \sum_{j:a_j \in T} d_j \end{aligned} \tag{3}$$

where T is a spanning tree and optional constraints may be easily included depending on the nature of the broadcast service (for example in QoS type services there will be a minimal guaranteed bandwidth in all arcs).

The calculation of non-dominated trees for this problem will enable the obtainment of a solution that represents a trade-off between the impact on the global network traffic performance resulting from the utilization of a tree and the minimization of an average delay bound on the tree arcs, depending on the network current state of occupation of all its arcs (given by θ_j).

2.2 Review of bi-criteria spanning tree algorithms

In Gomes da Silva and Clímaco (2007) it is proposed a procedure for the bi-criteria minimum spanning tree problem (BMST), which is based on the extension of the Kruskal's algorithm (see Ahuja, 1993) and the weighted-sum approach. It must be emphasised that this procedure enables the computation of the set of supported non-dominated solutions in ordered manner. The procedure is based on a systematic detection of arcs which must be replaced in one non-dominated solution to obtain the new adjacent one, in the criterion space. The main features of this procedure are the re-use of computations made in previous iterations and the fact that it avoids the repetitive resolution of single criterion problems. In the problem (Eq. 3) two criteria are considered, cost and delay. In general terms, in this section, we designate $Z_1 \equiv \Phi\left(T\right)$ and $Z_2 \equiv D\left(T\right)$, given in Eq. 3, as first and second cost functions.

Supported non-dominated solutions optimize weighted sum functions. In the bi-criteria case, these functions are $f_\lambda(T) = \lambda Z_1\left(T\right) + \left(1-\lambda\right) Z_2\left(T\right)$, with $0 \leq \lambda \leq 1$. When the entire interval $[0,1]$ for λ is analyzed, then all the non-dominated supported solutions are found. Let G^λ be the graph G such that each arc a_j has the cost $p_j^\lambda = \lambda c_j^1 + \left(1-\lambda\right) c_j^2$, with $c_j^1 = \phi_j$ and $c_j^2 = d_j$.

We are interested in the computation of such solutions but in an ordered manner.

A spanning tree is composed of $n-1$ arcs, and following the Kruskal's algorithm, one has to consider the arcs according to non-decreasing costs and select the $n-1$ arcs that enable the drawing of a spanning tree on the given graph.

In the bi-criteria case, the cost of each arc a_j is p_j^λ, which depends on the value of λ. Thus, the minimum spanning tree of $f_\lambda(T)$ also depends on the value of λ. As λ changes in the interval $[0,1]$ some arcs also change their relative position, whimum spanning tree if and only if every non-tree arc a_j has a cost greater or equal than the cost of any arc in the unique path of T that links the nodes concerning a_j.ch may lead to a different minimum spanning tree.

According to the path optimality condition a spanning tree T is a mini

Suppose that for a given λ, λ^k, the corresponding minimal spanning tree is T^{λ^k} and the associated supported non-dominated solution is $\left(Z_1\left(T^{\lambda^k}\right), Z_2\left(T^{\lambda^k}\right)\right)$. This tree remains optimal for every λ which observes the path optimality condition.

Let $P^{\lambda^k}(a_j)$ be the path in T^{λ^k} which links the nodes concerning the non-tree edge a_j. For the non-tree edge a_j the maximum value of λ is given by the solution of the linear problem:

$$\begin{array}{l} \max\ \lambda \\ s.t.: \\ p_j^\lambda \geq p_i^\lambda, \forall a_i \in P^{\lambda^k}(a_j) \\ 0 \leq \lambda \leq 1, \lambda > \lambda^k \end{array} \tag{4}$$

In order to preserve the optimality of the spanning tree, the path optimality condition must be observed by every non-tree edge. Thus the overall maximum value is given by $\lambda^k_{\max} = \min\limits_{a_j \notin T^{\lambda^k}} \left\{\lambda^{a_j}_{\max}\right\}$.

From these results it is easy to identify the candidate arcs to enter and leave the current solution in each iteration. For details see Gomes da Silva and Clímaco (2007).

Note that, if $\lambda = \lambda^k_{\max}$ the current tree is still the optimal tree, but there exists at least another tree with the same weighted cost. Finding all the supported non-dominated solutions with the same weighted cost as T^{λ^k}, requires replacing in the tree every combinations of possible pairs of arcs.

If $\lambda = \lambda^k_{\max} + \varepsilon$ (ε is a small value) the current tree is not the optimal of the problem $\min\{f_\lambda(T) : T \text{ is a spanning tree on } G\}$, since at least one non-tree arc has a smaller weight compared with at least one arc of the associated path in the tree. An optimal tree in this case is the one which has the minimum for criterion $Z_1\left(T\right)$ (note that λ is increasing towards the value 1), among all the efficient MST obtained with $\lambda = \lambda^k_{\max}$.

So, in summary, the procedure starts with the computation of an initial MST, which optimizes criterion Z_2 (chosen arbitrarily). The maximum value of λ which maintains the MST is computed as well as the candidates arcs to enter the tree. If the maximum of λ is greater than or equal to 1 the procedure stops. Otherwise, the leaving arcs are identified for each entering candidate. Supported non-dominated trees generated by removing and inserting identified arcs are used to update the list of solutions. The tree corresponding to the lowest value of criterion Z_1 is the new reference tree and the above steps are repeated.

The formalization of the procedure that generates the supported non-dominated solutions of a bi-critira MST can be seen in Gomes da Silva and Clímaco (2007).

An alternative, more general procedure for computing the set of supported non-dominated solutions consists of using iteratively weighted-sum functions until no new solutions are found (Hamacher and Ruhe, 1994). The method requires the optimization of several single criterion functions, some of them without producing any new efficient solution, and the obtained solutions are not obtained in an ordered manner.

3 Application model and results

In order to assess the performance of the proposed broadcast model, it was applied to two sets of networks with randomly generated topologies. In both sets the topologies were generated with the GT-ITM Georgia Tech Internetwork Topology models software. This software (in the site http://www.cc.gatech.edu/fac/Ellen.Zegura/graphs.html) generates Internet type topologies with different structures, randomly generated, which satisfy certain laws that define the probability of occurring an arc between any two given nodes. These probability laws are typically a negative exponential function of the Euclidean distance between the two nodes and also depend on some calibrating parameters. In the present context we wanted to obtain a certain specified approximate average node degree δ and consider a maximal Euclidean distance $L_{\max}$. Hence, we used as an adequate probability distribution, the Doar-Leslie model (Doar and Leslie, 1993). This distribution was calibrated for each given number of nodes, n, to obtain approximately the designed node degrees $\delta_1 = 2.7$ and $\delta_2 = 4$, common typical values in these networks. In each set of experiments the number of nodes was 14, 30, 50, 100 and 150. In the first set of experiments (A) the networks had arcs with equal capacities $c_j = 155,52Mb/s$ and the parameter $M_j^{\max} = 672$ bits (a value also used in Rosenbaum et al. (2005)). The maximal distance in the Doar-Leslie model was $L_{\max} = 35$ Km for $n = 14, 30$, and 50 and $L_{\max} = 1060$ Km for $n = 100$ and 150.

The available bandwidths b_j in all arcs of the defined networks were randomly generated in sets I_i according to the following empirical statistical distributions:

Distribution 1: I_0-20%; I_1-20%; I_2-20%; I_3-20%; I_4-20%

Distribution 2: I_0-50%; I_1-20%; I_2-15%; I_3-10%; I_4-5%

Distribution 3: I_0-41%; I_1-6%; I_2-6%; I_3-6%; I_4-41%

where I_i are five discrete sets of values with equal amplitudes:

$I_i = \{0, 52 + 2k : k = 15i, \ldots, 15(i+1) - 1\}, i = 0, 1, 2, 3;$

$I_4 = \{0, 52 + 2k : k = 60, \ldots, 75\}$

For each topology and a given distribution, 15 instances were generated in this manner and for each n and δ, five network topologies were considered. Therefore, in this set of experiments, 1125 instances of the bi-criteria optimization problem were solved.

The second set of experiments, B, were designed in perfectly analogous form with arc capacities $c_j = 4$ Mb/s and $M_j^{\max} = 3072$ bits. The empirical distributions, for the b_j in terms of the percentage of values assigned to each set I_i are the same, while the intervals constructed with a similar principle as in the other set of experiments A are:

$$I_i = \{0, 18 + 0.0515k : k = 15i, \ldots, 15(i+1) - 1\}, i = 0, 1, 2, 3, 4$$

In Fig. 1 we present the average, minimum and maximum of the number of non-dominated supported solutions obtained by the algorithm for $\delta_1 = 2.7$ and different distributions as a function of the number of nodes, for all instances of the set of experiments A.

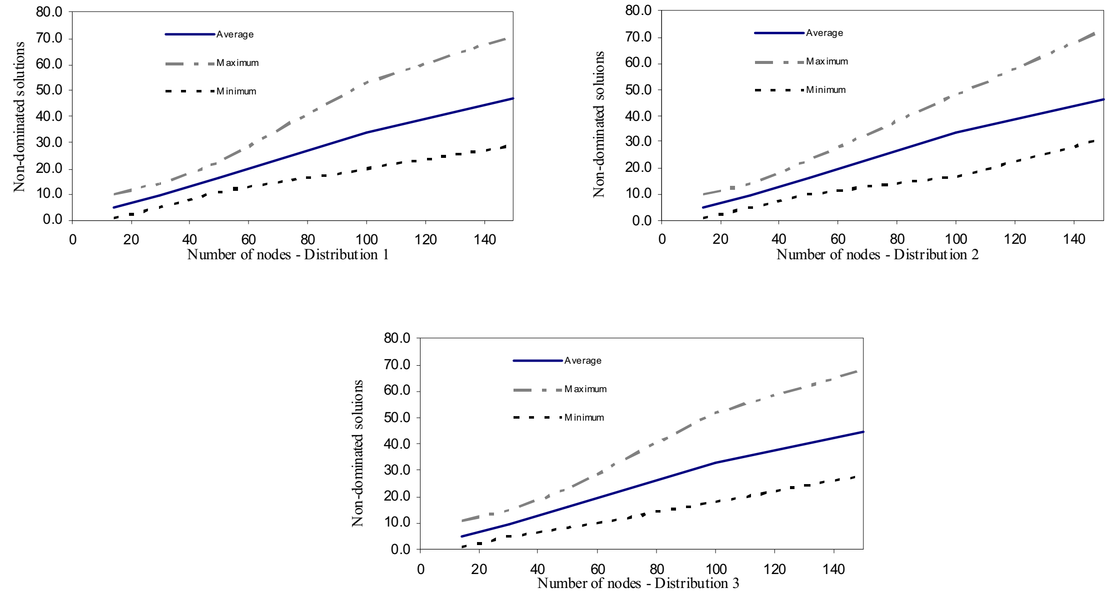

Fig. 1: Non-dominated solutions (δ_1=2.7)

As expected, the number of solutions tends to increase with the number of nodes. The variation between the minimum and the maximum tends to increase for values of $n \geq 50$. The average, minimum and maximum values of CPU times (in seconds) are displayed in Fig. 2, also for the three distributions. All these values tend to increase with n, the major difference to be observed in the results being the significantly higher values of maximum CPU for Distribution 1 (as compared to Distributions 2 and 3) which corresponds to an uniform empirical distribution of the b_j in all the defined sets of values I_i.

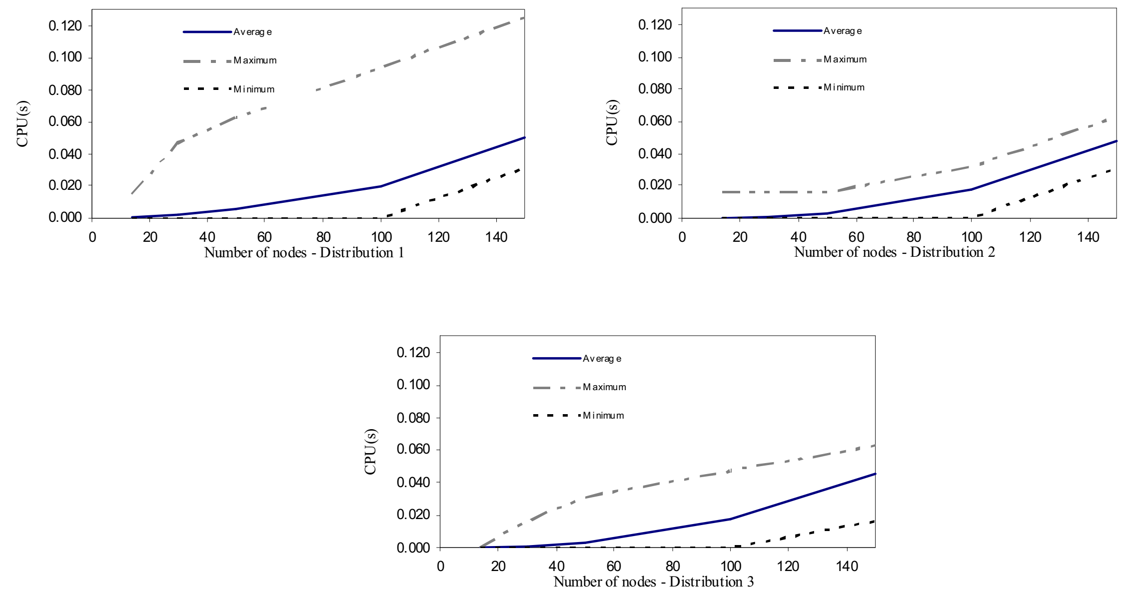

Fig. 2: CPU(s) (δ_1=2.7)

In any case, taking into account that each point in the graphics corresponds to a value (average, minimum or maximum) averaged over all 75 instances of the problem for fixed n and δ we may conclude of the very great efficiency of the algorithm. This makes it adequate not only for off-line broadcast routing methods but also for on-line dynamic broadcast routing for networks within the considered topological ranges.

Similar graphics were obtained for the set of experiments for networks with a higher average node degree, $\delta_2 = 4$, that are shown in Fig. 3 and Fig. 4. The major difference by comparison with the results for $\delta_1 = 2.7$ is the significant increase in the number of obtained non-dominated solutions, for all the distributions, that is the expected consequence of the increase in the number of possible solutions associated with the increase in the network density. Similar comments apply to the CPU times, that nevertheless still remain in levels compatible with the practical applications mentioned above.

Similar graphics were obtained for the various experiments B. Since the results follow the same pattern and trends as for experiments A, the graphics are not presented.

Concerning the selection of a solution in the set of non-dominated supported solutions Γ_N it must be remarked that it has to be executed in an automated manner, in a very short time. For this purpose we propose a two-stage procedure.

The first stage assumes that a certain maximal delay per path D'_M is admissible, depending on the nature of the broadcast service. For example in QoS real-time services with guaranteed QoS parameters and stringent transmission requirements, this constraint on delay results from technical specifications while for other services it can be viewed just as an acceptable QoS target (not necessarily mandatory). This bound can then be used for "filtering" the solutions, possible in conjunction with other QoS constraints, for example related to "bottleneck bandwidth" in the arcs.

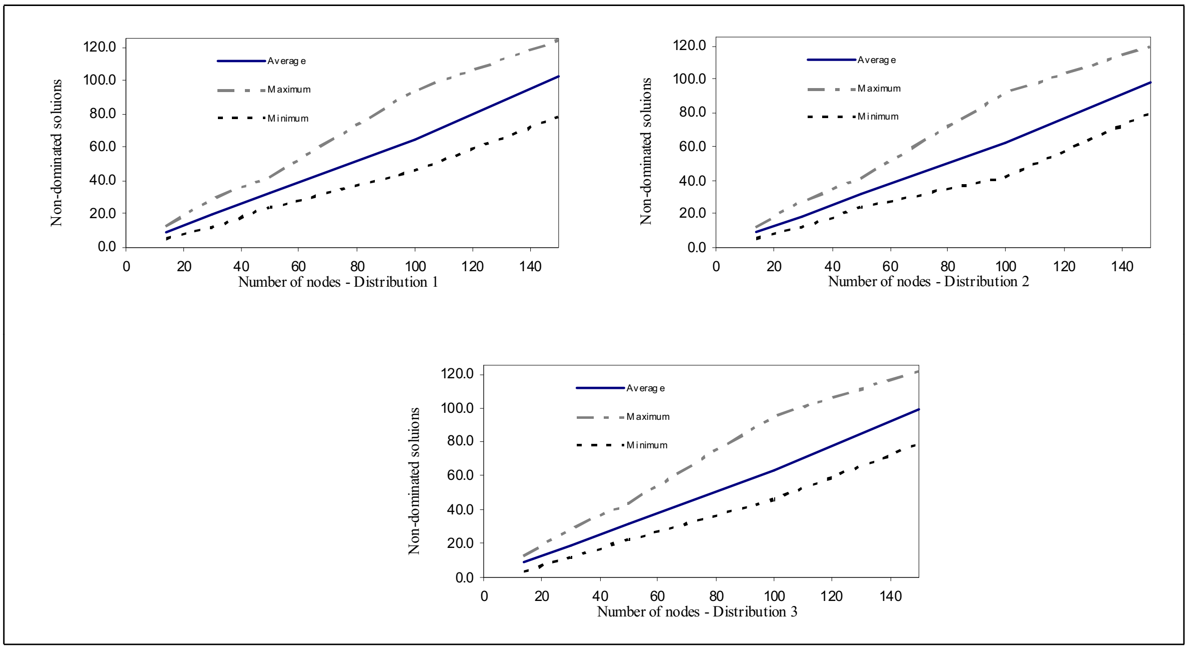

Fig. 3: Non-dominated solutions (δ_2=4)

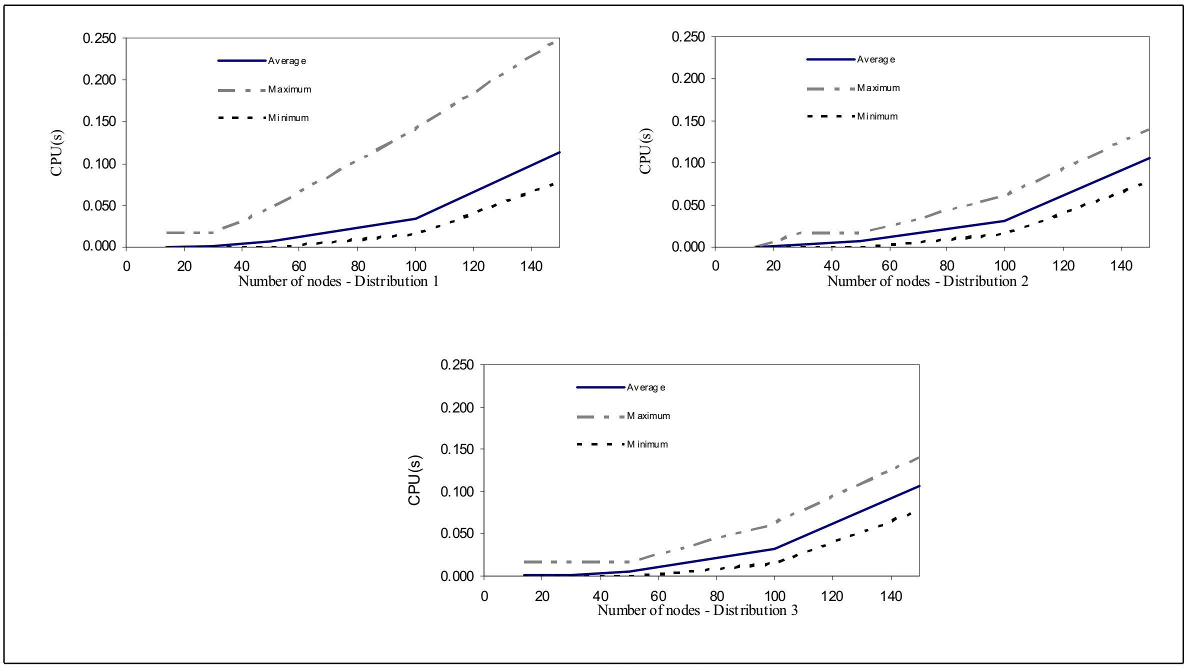

Fig. 4: CPU(s) (δ_2=4)

Hence, in this first stage, we would have to calculate, for each solution T_k and for a given root node, s, the maximal delay path $p_k^M \subset T_k$ and consider for further analysis just the "filtered" set Γ' of solutions ($\Gamma' \subseteq \Gamma_N$) such that:

$$d'_k = d\left(p_k^M\right) \leq D'_M\left(1-\epsilon\right), 0 \leq \epsilon < 1 \tag{5}$$

where $p_k^M = \arg\max\limits_{p \subseteq T_k} \left\{ d'(p) = \sum\limits_{a_j \in p} d_j \right\}$ and ϵ gives the degree of "tightening" of this filtering mechanism, that can be viewed as a floating barrier on path delay.

The second stage involves the selection of a solution in the set Γ' when $|\Gamma'| > 1$. For this purpose we will adapt to this model a method proposed in Clímaco et al. (2006), that is based on a reference point-like approach, combined with the use of preference thresholds. The first step of the selection procedure involves the calculation of preference thresholds in the form of requested and acceptable values for the values of each objective function. From these thresholds, denoted by the index "acc" (for acceptable) and index "req" (for requested), we may define priority regions in the objective function space as illustrated in Fig. 5.

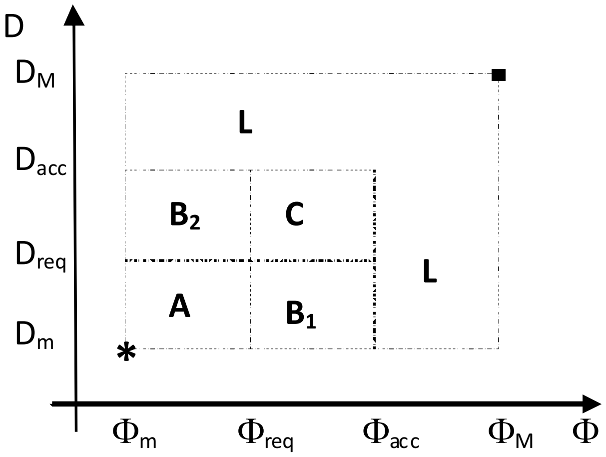

Fig. 5: Priority regions

In Fig. 5, A is the first priority region where the requested values for both functions are satisfied. B_1 and B_2 are second priority regions where only one of the requested values is satisfied while the acceptable value for the other function is guaranteed. A further preference order between B_1 and B_2 is established by giving preference to solutions with less delay bound, that is solutions found in B_1 are given preference over solutions in B_2. C is the third priority region where only the acceptable values for the two functions are satisfied. L is a last choice preference region, only considered when no solution in Γ' were found in A, B_1, B_2 or C.

The preference thresholds are obtained as follows. Denoting by $\Phi_m = \min_T \Phi(T)$, $D_m = \min_T D(T)$, $T_1^{\Phi} = \arg\min_T \Phi(T)$, $T_1^{D} = \arg\min_T D(T)$, $\Phi_M = \Phi\left(T_1^D\right)$, $D_M = D\left(T_1^{\Phi}\right)$, $\overline{\Phi} = \frac{\Phi_m+\Phi_M}{2}$, $\overline{D} = \frac{D_m+D_M}{2}$

$$\begin{array}{ll} \Phi_{acc} = \frac{\overline{\phi}+\Phi_M}{2} & D_{acc} = \frac{\overline{D}+D_M}{2} \\ \Phi_{req} = \frac{\Phi_m+\overline{\phi}}{2} & D_{req} = \frac{\overline{D}+D_m}{2} \end{array} \tag{6}$$

Concerning the selection of a solution in the higher priority region with at least one solution, $S \in \{A, B_1, B_2,\ C, D\}$, when the number of solutions in S is greather than one this is achieved by using a weighted Chebyshev distance to a reference point, where the weights reflect the form of the region where the solutions are located. This leads, at this point, to a reference point based procedure, by considering the "left bottom corner" of S as reference point. Note that this point coincides with the ideal solution (Φ_m, D_m) if $S = A$.

Usually, reference type approaches minimise the distance to candidate solutions to a certain point by using a certain metric, recurring to a scalarizing function Wierzbicki (1980). In our model, following Clímaco et al. (2006), we used a weighted Chevychev metric with weights proportional to the size of rectangle S. Denote by (Z_1^*, Z_2^*) the coordinates of the "left bottom corner" of S, by (M_1, M_2) the coordinates of "right top corner" of S and let w_i denote the weight with dimension free values (see Fig. 6):

$$w_i = \frac{1}{M_i - Z_i^*}, i = 1, 2 \tag{7}$$

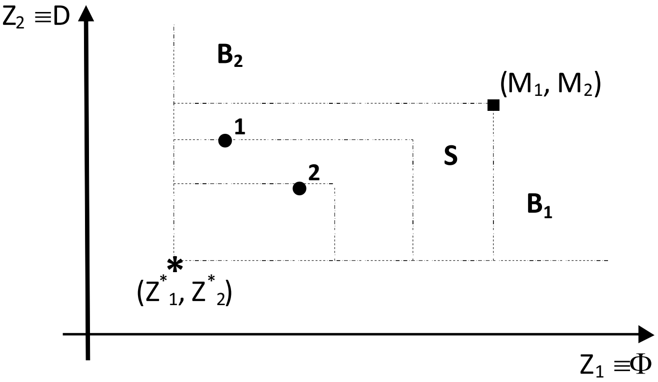

Fig. 6: Ranking solutions

Therefore the chosen solution, T^*, is given by $T^* = \arg \min_{T \in S^c \cap \Gamma'} \max \{ w_i \left| Z_i(T) - Z_i^* \right| \}$, where S^c is the set of solutions which correspond to points in S.

In the example illustrated in Fig. 6 this procedure would lead to the selection of solution corresponding to point 2.

To illustrate the application of the first stage procedure, Table 1 shows the values of the two objective functions for the 13 solutions obtained in an instance of the bi-criteria problem in the set of experiments A of the experiments together with the maximal path delay d'_k from a given root node (Eq. 5) for every solution.

Considering that $D'_M = 10$ ms and choosing $\epsilon = 15\%$ only the solutions for which $d'_k \leq 8.5$ ms would be acceptable at this stage, a subset $\Gamma' = \{2, 5, 8, 9, 11, 12, 13\}$ would be obtained, by filtering the initial solution set. The final solution would then be selected in Γ' by following the method described above (second stage of the described procedure). After calculating the preference thresholds ($\Phi_{acc} = 262.1925; \Phi_{req} = 120.5195; D_{acc} = 24.632; D_{req} = 24.232$), Table 2, shows the preference regions for the solutions in Γ'.

As there are two solutions in region B_1, the weighted Chevychev metric is used. Finally, the selection procedure will choose solution 8.

It must be remarked that by changing the parameter ϵ in Eq. 5 one may "modulate" the candidate solutions set Γ', which enables widening or tightening the compromise solutions set to be analyzed at the final stage. This possibility enhances the flexibility of the proposed broadcast automated approach.

Solution	Φ	D	d'_k
1	49.683	24.832	10.241
2	49.9405	24.797	8.115
3	50.713	24.702	11.717
4	54.1695	24.597	11.717
5	59.5315	24.482	7.662
6	68.344	24.357	11.024
7	89.967	24.287	11.024
8	127.31	24.212	8.430
9	144.744	24.192	8.225
10	179.004	24.157	9.821
11	276.599	24.072	8.115
12	304.093	24.052	8.115
13	333.029	24.032	8.115

Table 1: Non-dominated solutions and maximal delay

Solution	Preference Region
2	L
5	B_2
8	B_1
9	B_1
11	L
12	L
13	L

Table 2: Non-dominated in preference regions

4 Conclusions and further work

A new bi-criteria minimum spanning tree model designed for routing broadcast messages in MPLS networks was presented. The model enables the calculation of compromise solutions with respect to two important traffic engineering metrics: load balancing cost (which seeks the minimization of the impact of the computed tree on the existing network flows) and a delay bound (which seeks the minimization of the average maximal delay per arc tree). The formulated bi-criteria optimization problem is solved by a very efficient algorithm which enables to compute the supported non-dominated solutions in very short times for networks with a wide range of number of nodes and for average node degrees typical of core and metro-core MPLS networks of great dimension. A simple procedure based on preference thresholds and reference points enables a fast, flexible and automated selection of a final solution. These conclusions concerning the effectiveness of the proposed approach were suggested by an experimental study considering a large number of randomly generated networks, representing typical Internet type network structures, with a number of nodes varying from 14 to 150.

The application of the used bi-criteria minimum spanning tree algorithm is limited to undirected networks, hence limiting the present applicability of the resolution approach to MPLS networks with fully symmetric directed arcs. This is a limitation of the present approach that justifies further work concerning the extension of the algorithm to generic types of directed networks.

References

[1] Ahuja, R., Magnanti, T. and Orlin, J. (1993). *Network flows - theory, algorithms and applications*, Prentice Hall.

[2] Awduche, D., Chiu, A., Elwalid, A., Widjaja, I., Xiao, X. (2002). Overview and Principles of Internet Traffic Engineering, *IETF Network Working Group*, RFC3272.

[3] Cerulli, R., Fink, A., Gentili, M. and Voss, S. (2006).Extensions of the minimum labelling spanning tree problem. *Journal of Telecommunications and Information Technology,* 4, pp. 39 - 45.

[4] Clímaco, J., Craveirinha, J., Pascoal, M. (2006). An automated reference point-like approach for multicriteria shortest path problems. *Journal of Systems Science and Systems Engineering,* 15 (3), pp. 314-329

[5] Clímaco, J., Craveirinha, J., and Pascoal, M. (2007). Multicriteria routing models in telecommunication networks – overview and a case study. In *Advances in multiple criteria decision making and human systems management: knowledge and wisdom.* In Y. Shi, D. Olson, and A. Stam (editors), edited in honor of Milan Zeleny, chapter 1, pp. 17–46. IOS Press.

[6] Craveirinha, J., Clímaco, J., Pascoal, M. and Martins, L. (2007). Traffic splitting in MPLS networks - a hierachical multicriteria approach. *Journal of Telecommunications and Information Technology*, 4, pp. 3-10.

[7] Doar, M., Leslie, I. (1993). How bad is naive multicast routing ? INFOCOM (1), San Francisco, USA, pp. 82-89.

[8] Erbas, S., Erbas, C. (2003). A multiobjective off-line routing model of MPLS networks". In Proceedings 18th Int. Teletraf. Congr. Berlin, Germany.

[9] Gomes da Silva, C. and Clímaco, J. (2007). A note on the computation of ordered supported non-dominated solutions in the bi-criteria minimum spanning tree problems. *Journal of Telecommunications and Information Technology*, 4, pp. 11-15.

[10] Hamacher, H. and Ruhe, G. (1994). On spanning tree problems with multiple objectives. *Annals of Operations Research*, 52, pp. 209-230.

[11] Knowles, J., Oates, M., Corne, D. (2000). Advanced multiobjective evolutionary algorithms applied to two problems in telecommunications. British Telecom Technology Journal, 18 (4), pp. 51-65.

[12] Pinto, D. and Barán, B. (2005). Solving multiobjective multicast routing problem with a new ant colony optimization approach. In *Proceedings of the 3rd international IFIP/ACM Latin American conference on Networking*, pp. 11-19, Cali, Columbia.

[13] Rosenbaum, G., Chun Tung Chou, Jha, S. and Medhi, D. (2005). Dynamic routing of restorable QoS connections in MPLS networks, *Proc. of 30th Annual IEEE Conference on Local Computer Networks, (LCN2005)*, Sydney, Australia.

[14] Stiliadis, D. and Varma, A. (1998). Lattency-rate servers: a general model for analysis of traffic scheduling algorithms, *Transactions on Networking*, 6, pp. 611-624.

[15] Wierzbicki, A. (1980). The use of reference objectives in multiobjective optimization. In G. Fandel and T. Gal (eds), MCDM Theory and Applications, Proceedings, *Lecture notes in Economics and Mathematical Systems*, 177, pp. 468-487, Springer Verlag.