Minimizing the Energy Consumption of Carrier Grade Ethernet with Multiple Spanning Trees

Antonio Capone♣, Daniele Corti♣, Brunilde Sansò♢
♣ Dipartimento di Elettronica e Informazione, *Politecnico di Milano*, Italy
♢ Département de génie électrique of the *École Polytechnique de Montréal*, Canada

***Abstract*—This paper addresses the problem of energy consumption in the Metropolitan Area Networks that employ Carrier Grade Ethernet. The objective is to minimize the energy consumption of nodes and links, managing the traffic through the Multiple Spanning Tree Protocol (MSTP), where the spanning trees are appropriately chosen from a given set and matched with the traffic demands. We present a modelling framework, a resolution approach and comment on the different features that affect energy performance.**

I. Introduction

Minimizing energy consumption in the area of Telecommunication Networks has become an increasingly important issue as the Internet continues its grow. Significant energy could be saved by applying energy efficiency concepts. However, even though there has been remarkable results in the area of energy efficient wireless devices, the concept of energy efficiency is just beginning to surface in wireline networking (for an overview see the pioneering work of [1]).

In this paper we deal exclusively with the energy efficiency of Metro Networks and, in particular, Carrier Grade Ethernet. Simplicity and the high bandwidth available at low cost made Ethernet an attractive choice in various networking deployments. Nowadays Ethernet is no longer confined to be used in LANs, but its high speed versions are also employed in server interconnections, MANs, WANs and Carrier Networks. However, Metro Networks have different requirements from the early LAN. As such, traditional Ethernet did not have the characteristics required in Carrier Grade networks and needed to be enhanced if it has to meet these requirements. Therefore, the past few years have seen significant innovations around Ethernet standards in order to meet recent requirements.

Most of the new standards developed for Carrier Grade networks rely on Spanning Tree Protocols (STP). The STP is responsible for building a loop-free logical topology over the physical one using a shortest path approach and ensuring connectivity among all nodes.

978-1-4577-0351-5/11/$26.00

An evolution of STP, the Multiple Spanning Tree Protocol (MSTP), guarantees better performances in terms of network utilization, recovery time in case of failures, load balancing and Qos functionality. It allows a switch to participate in multiple spanning trees, providing a flexible way to implement traffic engineering in Metro Ethernet. In these networks, traffic flows are defined on a per VLAN basis (the simplest VLAN that can be considered is an E-Line VLAN , i.e., a point-to-point VLAN that carry a single commodity) and routing of the traffic flows is done through the network based on Spanning Trees (STs), so that when a VLAN is assigned to a ST Instance (STI), its traffic flow is routed between its end nodes through the unique path defined by the assigned STI.

In this paper, we exploit the specific features of the Multiple Spanning Tree Protocol to reduce the energy consumption of Carrier Ethernet deployment. To the best of our knowledge, this is the first time such an approach is taken for Carrier Ethernet. The reminder of the paper is described as follows. Section II gives a brief review of previous related work on energy consumption and Spanning Tree design. The proposed optimization model is explained in Section III. In Section IV the solution approach is presented. Finally, some computational results are listed in Section V. Conclusions and a brief discussion for future works are presented in Section VI.

II. Related work

There has been several approaches to reduce power consumption. For instance, it has been proposed to adaptively vary the link rate in order to match the offered load or utilization [2]. Other attempts focus on the possibility of introducing a low power "sleep" mode for idle links in order to obtain energy savings [3]. In [4] a protocol that coordinates how routers go into power saving mode without degrading quality of service nor network connectivity is proposed. For an overall survey on other methods introduced in green networking, the reader is referred to [5].

The IEEE 802.1s standard that deals with the Spanning Trees (STs) does not provide any criterion on how to map STs to VLANs, despite the importance

*

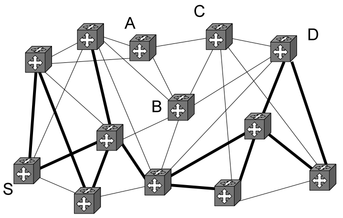


Figure 1. An example of how the algorithm should led to use the fewest possible of nodes and arcs avoiding to route traffic in a portion of the network (in this case the one that includes nodes A, B and C).

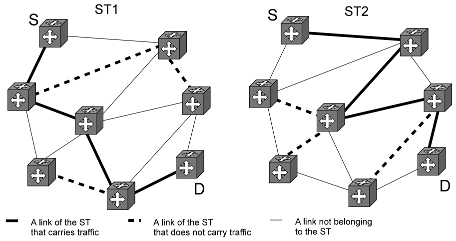


Figure 2. A VLAN carrying a commodity with source node S and destination node D matched with two different Spanning Tree results in two different paths between source and destination.

of choosing the appropriate ones to facilitate network management. These points are crucial for the good performance of the MSTP. As a result, traffic engineering of Ethernet using Multiple Spanning Trees is a widely researched topic.

In [6], [7] and [8] the authors claim the importance of an opportune cost metric in the tree construction algorithm. In [9] and [10] the authors propose a technique to determine the MSTP parameters configuration that minimizes the impact of network failures and optimizes load balancing. This objective is achieved minimizing the Worst case Link Load (WLL) and the Average Link Load (ALL). Some other algorithms are presented in [11] and [12], having as objective the load balancing.

This article proposes to use the IEEE 802.1s Multiple Spanning Tree Protocol in order to minimize the energy consumption of a given network, finding the best subset of spanning trees and the best mapping of the traffic demands to them. In the next section, this approach will be introduced.

III. Energy Optimization

A. *System description*

Consider an Ethernet Carrier network composed by switches connected through point-to-point links. The network could be represented by a graph consisting of nodes and edges. Since Ethernet uses full-duplex links, traffic in both directions must be considered separately. For this reason, edges must be distinguished from arcs, where an arc is equivalent to a directed edge. Each edge has a known full-duplex capacity and an energy consumption, as well as each node has an energy consumption. The network supports a set of VLANs, each of which carries a single commodity characterized by its origin node, its destination node and its traffic demand.

The model presented in this article aims at finding the best Spanning Trees among a given set of Spanning Trees covering the network and map them to the VLANs that must be routed, minimizing the energy spent by the network itself. The idea of the energy management scheme is that nodes and arcs that carry no traffic can be switched off (in other words, are put in sleep mode), while ensuring full connectivity of the network. This leads to use the fewest possible number of nodes and arcs. For example, suppose that the links in bold in the network of Figure 1 are the ones through which the traffic is routed. If a portion of traffic is sent between node S and node D, this flow, considering the minimization of the energy consumption, should be assigned to a Spanning Tree whose paths between the source and the destination node belongs to the portion of the network which has already been employed (therefore avoiding nodes A, B and C).

The traffic flows are routed over the network over paths defined by the Spanning Trees the VLANs have been assigned to. Assigning a VLAN to a Spanning Tree means determining the route that should be taken by the traffic of that VLAN. Assigning it to a different Spanning Tree means to route it over a different path. An example is given in Figure 2. Note that the links that do not carry traffic (even the ones belonging to the Spanning Tree matched with the considered VLAN) can be switched off.

B. *Notational description*

After a brief description of the modelling philosophy, additional notation is required before presenting the formulation of the model. The notation is divided in model parameters and decision variables:

Model parameters

N	the set of nodes, each of which represents a switch.
E	the set of edges, each of which represents an undirected link between two nodes.
$G(N,E)$	the graph composed by nodes and edges representing a network.
A	the set of arcs, which are directed links between nodes.
Q	the set of VLANs to be routed.
S	the set of Spanning Trees.
$o_q \in N$	the origin node of the VLAN $q \in Q$.
$b_q \in N$	the destination node of the VLAN $q \in Q$.
$d^q \in \mathbb{R}^+$	the traffic demand of the VLAN $q \in Q$.

*

$c_{\{ij\}} \in \mathbb{R}^+$	the full-duplex capacity of the edge $\{i,j\} \in E$.
$\varepsilon_{\{ij\}} \in \mathbb{R}^+$	the energy consumption of the edge $\{i,j\} \in E$.
$\varepsilon_i \in \mathbb{R}^+$	the energy consumption of the node $i \in N$.
$K \in \mathbb{N}$	the maximum number of Spanning Trees that can be mapped with the VLAN.
a_{ij}^{sq}	$= 1$ if arc $(i,j) \in A$ is in the path of the Spanning Tree $s \in S$ for the VLAN $q \in Q$, and 0 otherwise.

The last parameter is as crucial as complex. It is a binary parameter that defines if a VLAN could be routed over a link accordingly to the Spanning Tree it is has been matched to.

Decision variables

The problem of limiting the energy consumption of the network routing the VLAN through the best Spanning Trees while continuing to ensure full connectivity and the management of all requests can be formulated using four decision variables.

The first variables state which STs have been chosen in the solution among the set S: λ_s is equal to 1 if Spanning Tree $s \in S$ is chosen in the solution, and it is 0 otherwise.

The mapping between VLANs and STs is defined by the second binary variables: ϕ_s^q is equal to 1 if VLAN $q \in Q$ is assigned to Spanning Tree $s \in S$, and it is 0 otherwise.

Finally, the last two groups of variables are those that determine whether a node or an edge respectively are switched on or not: $x_{\{ij\}}$ is equal to 1 if edge $\{i,j\} \in E$ is switched on, and it is 0 otherwise; y_i is equal to 1 if node $i \in N$ is switched on, and it is 0 otherwise.

C. The energy management optimization model

Objective function

$$\min z_1 = \sum_{i \in N} y_i \varepsilon_i + \sum_{\{i,j\} \in A} x_{\{ij\}} \varepsilon_{\{ij\}} + \\ +\beta \left\{ \sum_{i \in N} (1 - y_i) \varepsilon_i + \sum_{\{i,j\} \in A} (1 - x_{\{ij\}}) \varepsilon_{\{ij\}} \right\} \quad (1)$$

The term z_1 is composed of two parts. The first is the energy consumption of the switched on nodes and edges belonging to the network. The second is the consumption of the elements in sleep mode. It is weighted by the parameter β, that is set to 0.1 according to Energy Efficient Ethernet (EEE) estimates.

Constraints

$$\sum_{s \in S} \sum_{q \in Q} a_{ij}^{sq} \phi_s^q d^q \leq c_{\{ij\}} x_{\{ij\}} \qquad \forall (i,j) \in A \quad (2)$$

Constraints (2) are the capacity constraints for each arc (i,j). If the edge $\{i,j\}$ is switched off, no traffic can be routed on it.

$$\phi_s^q \leq \lambda_s \qquad \forall s \in S, q \in Q \quad (3)$$

Constraints (3) guarantee that a single VLAN q can be assigned to a Spanning Tree s only if s has been chosen in the solution.

$$\sum_{s \in S} \phi_s^q = 1 \qquad \forall q \in Q \quad (4)$$

Constraints (4) guarantee that each VLAN q is assigned to one Spanning Tree s.

$$\sum_{s \in S} \lambda_s \leq K \quad (5)$$

Constraint (5) guarantees that no more than a fixed number K of trees are in the solution.

$$\sum_{j \in N} x_{\{ij\}} \leq y_i 2K \qquad \forall i \in N \quad (6)$$

Constraints (6) guarantee that a node i can be switched off only if there are no active links incoming or outgoing from that node.

IV. Solution approach

The resolution approach is closely related to the choice of network instances and Spanning Tree choices. The networks considered have been randomly generated through an instance generator implemented in the C++ programming language. Having fixed the number of nodes and edges, the program is able to generate a network randomly choosing the edges from a full-mesh grid, maintaining these edges in the network with a predetermined probability p and discarding the other. In these networks, all the links capacity were set to 100 Gbps. Concerning traffic demands, they too have been generated through the instance generator. Origin and destination have been randomly selected among all nodes and their bandwidth requests have been randomly generated between 0.1 Gbps and 10 Gbps.

To present computational results we have chosen a Cisco CRS-3 16-Slot Single-Shelf System [13]. This device is compatible with the Cisco CRS-3 1-Port 100 Gigabit Ethernet Interface Modules, each of which is responsible for the ingress and the egress packet processing, so that in order to build up a 100 Gbps full-duplex link, we must employ two modules (one at each end).

Regarding the energy consumption parameters, they have been estimated from the device datasheets. The maximum power consumption of a router when chassis is fully configured with line cards and with traffic running is estimated to be 12320 Watts, while the energy consumption of a single line card is 150 Watts (therefore the

*

energy consumption of a full-duplex link is considered to be 300 Watts). Finally, the power consumption in the low-power mode is assumed to be 10% of that in the active mode, in line with the estimates provided by different manufacturers during the standardization process of Energy Efficient Ethernet (EEE) [3].

Algorithm IV.1: ST GENERATION(E, N, s)

```
$Tree \leftarrow \emptyset$
$S \leftarrow \emptyset$
for each $(i,j) \in E$
  do $cost(i,j) \leftarrow Uniform[0,1]$
while $cont \leq s$
  do
    while $card(Tree) < card(N) - 1$
      do
        $BestEdge \leftarrow$ MinCost($E$)
        $E \leftarrow E \backslash BestEdge$
        NoLoop($BestEdge$)
    if ($\nexists\, T \in Forest : T = Tree$) and
    criterion($Tree$)
      then
        $S \leftarrow Forest \cup Tree$
        Update($cost$)
```

In addition to data on network topologies and traffic matrices, the model needs to have as input a large number of Spanning Trees for the considered networks. These were obtained through the Algorithm ST GENERATION, which then adds them to the set S.

Kruskal's algorithm has been used in order to find the Spanning Trees which are the elements of the set S. Each Spanning Tree has been generated by assigning random costs to the links with a uniform distribution, so that Kruskal's algorithm could give at each cycle a different Spanning Tree as output.

The function MinCost(E) provides the minimum cost link in the set E. The procedure NoLoop($BestEdge$) determines if the edge given as input will create a loop and adds it to the set $Tree$ if not. The procedure is defined by the Algorithm BestEdge.

In Algorithm IV.2 variable $comp[i]$ indicates which component of the temporary Tree the node i belongs to. Its default value is zero. Given an Edge (i,j) four cases must be considered:

- neither node i or j belongs to the tree ($comp[i] = comp[j] = 0$). In this case the edge (i,j) will not create loops and can be added to the set $Tree$;
- node i (or j) does not belong to the tree, while node j (or i) has already been added to the tree. Again the edge is added without creating loops;
- both nodes i and j already belong to the set $Tree$ but in two disconnected components ($comp[i] \neq comp[j]$). In this case the edge can be added and the two components are merged.
- both nodes i and j already belong to the set $Tree$ in the same connected component ($comp[i] = comp[j] \neq 0$). In this case the edge cannot be added, otherwise it would create a loop.

Algorithm IV.2: NoLoop($BestEdge$)

```
if $comp[i] = 0$
  then
    if $comp[j] = 0$
      then
        $comp[i] \leftarrow NewComp$
        $comp[j] \leftarrow comp[i]$
        $Tree \leftarrow Tree \cup (i,j)$
      else
        $comp[i] \leftarrow comp[j]$
        $Tree \leftarrow Tree \cup (i,j)$
  else
    if $comp[j] = 0$
      then
        $comp[j] \leftarrow comp[i]$
        $Tree \leftarrow Tree \cup (i,j)$
    else if $comp[j] \neq comp[i]$
      then
        for each $n \in N$ :
        $comp[n] = comp[j]$
          do $comp[n] \leftarrow comp[i]$
        $Tree \leftarrow Tree \cup (i,j)$
```

The function criterion($Tree$) provides an additional criterion to decide whether to add or not the considered tree to the set S. If not mentioned, it's default value is 1.

Finally, function Update($cost$) assigns new costs to the links, according to the uniform distribution. When a tree is complete, it is added to the set S, and also saved into a parameter T such that $T[i,j,s] = 1$ if edge i,j belongs to the Spanning Tree s.

Other ways to generate trees have also been considered. In the first alternative, function criterion($Tree$) has been calibrated in such a way that only the trees which involve as few edges as possible could be added to the set S. In this case, the function has been set to 1 only when the number of edges employed to route traffic through the considered tree was below a threshold. Another option concerns the function to update the edge costs. In this case, at the end of each cycle a fixed value is added to the cost of every edge. This value is related to the number of times each edge has been chosen to belong to a tree, in order to discourage the choice of the edges that have been selected more often. This method should generate trees as diverse as possible. However, these alternative tree generator methods do not produce better results than the ones obtained with the basic one.

The data concerning the Spanning Trees are then used in the model in the form of the binary parameter a_{ij}^{sq} previously defined.

The values for parameter a_{ij}^{sq} are calculated, through a flow formulation, before running the model for each possible association between Spanning Trees and commodities. Note that given a Spanning Tree $s \in S$ and

Test	Output	Function of	Input parameters
1	Energy savings	|Q|	|N|,|E|
2	Time	|Q|	|N|,|E|
3	Energy savings	K	|N|,|E|
4	Energy savings	|S|	|N|,|E|

Table I
Roadmap of the optimizations performed.

a VLAN $q \in Q$, then the path between the origin node o_q and the destination node b_q is unique in s and the values of parameter a_{ij}^{sq} are uniquely determined. a_{ij}^{sq} must satisfy the following constraints:

$$\sum_{(i,j)\in A:T[i,j,s]=1} a_{ij}^{sq} + \\ - \sum_{(k,i)\in A:T[k,i,s]=1} a_{ij}^{sq} = \begin{cases} 1 & \text{if } i = o_q \\ -1 & \text{if } i = b_q \\ 0 & \text{otherwise} \end{cases} \\ \forall i \in N, \forall q \in Q, \forall s \in S \qquad (7)$$

Once the best subset of Spanning Trees and the best mapping of the traffic demands (VLANs) to them is known, then optimization model (1)–(6) can be run using CPLEX and AMPL. Thus, the traffic flows are routed over the network over paths defined by the STs the VLANs have been assigned to in such a way that a portion of the network is forced by the objective function of the model to remain unused. This makes possible to turn off the elements of that portion of the network, which are put into sleep mode to conserve energy. This leads to use the fewest possible nodes and arcs, while continuing to ensure full connectivity and the management of all requests.

V. Numerical results

In the following paragraphs the results obtained from the optimization of different networks are shown. These optimizations can be summarized as follows in Table I, according to the data analysed.

Results were initially obtained by solving the optimization model as a function of the number of VLANs $|Q|$ to be routed.

Figure 3 shows how much energy is saved by putting to sleep a portion of the network according to the model, compared to the fully switched on consumption.

The data relates to networks of different sizes (different $|N|$ and $|E|$). It is easy to see that the big savings are obtained with the largest networks, since small networks (from 10 to 20 nodes) are forced to keep active most of their network elements even to route small amounts of traffic. Note that the energy savings achieved decreases proportionally with the increase on the routed VLANs, until the network is saturated and no more savings can be obtained. As expected, the biggest savings are also achieved in case of low traffic levels, when it is easier to find unused network elements. All values are averaged over 15 different instances.

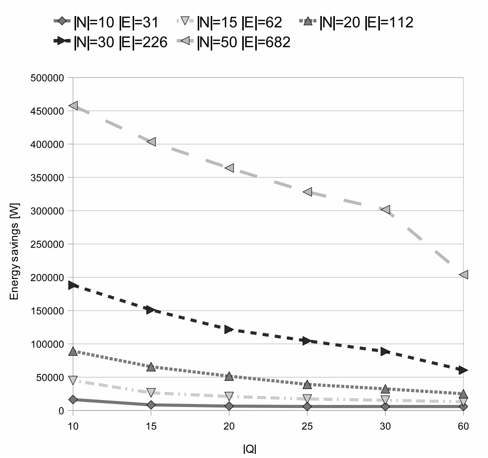


Figure 3. Energy saving for networks of different sizes and different number of considered VLANs $|Q|$ obtained by applying the model with the energy objective.

$\|N\|$	Differential savings [W] as a function of K				
	2	**3**	**5**	**10**	**15**
10	0	0	5940	0	0
15	0	0	26152	0	0
20	0	0	61902	0	0
30	-8658	-540	153828	0	0
50	-35982	-18126	406530	+7848	+7848

Table II
Differential energy savings for networks of different sizes and a varying number of MSTI allowed. Differentials have been taken with respect to the case of $K = 5$.

Concerning the mean computational time required to solve the model, it varies from less than 2 seconds to 1000 seconds for the biggest networks, when up to 60 VLANs are considered. This is the time limit that has been imposed.

In all cases it was considered that a maximum number of $K = 5$ Spanning Trees were chosen out of $|S| = 30$ possible Spanning Trees. Actually most of the switches support up to 15 Multiple Spanning Tree Instances, but increasing the allowed number of instances to more than 5 does not produce significant benefits, as shown by Table II.

Moreover, as regards to the number of trees used, it tends to be less than 5 (up to even a single tree) when the energy objective function is considered, according to Table III

The tests also show that increasing the size of set S does not improve the performance of the model in terms of energy savings, but merely expands the computational time.

Figure 4 shows the results obtained solving problem $P1$ for a network with $|N| = 10$ Nodes, $|E| = 31$ Edges

*

	Average # of MSTI				
$\|N\|$	$\|Q\| = 10$	$\|Q\| = 15$	$\|Q\| = 20$	$\|Q\| = 30$	$\|Q\| = 60$
10	3.6	4	4.6	4.93	4.93
15	3.4	4	4.47	4.87	5
20	2.67	3	4.07	4.67	5
25	3.47	2.13	2.87	3.53	5
30	2.67	1.8	2.27	3.4	4.93
60	2.73	1.93	1.5	1.93	3.13

Table III
Average number of MSTI activated.

and $|Q| = 60$ VLANs. The links on which traffic is routed are in bold.

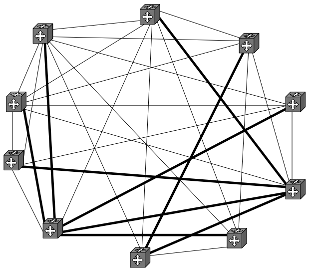
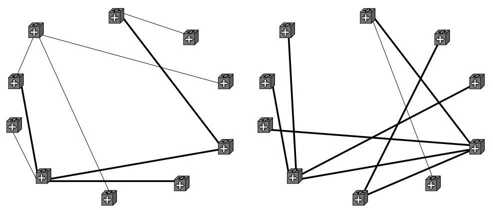
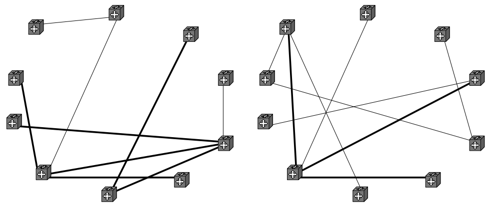


Figure 4. The network obtained minimizing the energy consumption.

Figure 5 illustrates in more detail the results concerning the network of Figure 4. It shows the Spanning Tree Instances chosen for the network, highlighting the links being used to route the commodities in each Spanning Tree. Note that in this case, only four ST have been included in the solution.



(a) ST1 (b) ST2



(c) ST3 (d) ST4

Figure 5. The Spanning Tree Instances chosen for the network of Figure 4.

VI. Conclusions

In this article we have presented an optimization model and a resolution approach for the optimal energy management of Carrier Ethernet networks using the MSTP. We have been able to evaluate the amount of savings that are involved in the scheme and how different network features affect energy consumption. We found that energy savings are more important in larger networks and that from the energy standpoint, there is an ideal number of trees that should be included in the solution.

Future work includes the improvement of the proposed model by adopting exact methods to generate the Spanning Tree via column generation, the introduction of load balancing indexes and an extensive experimental evaluation in real Metropolitan area networks.

References

[1] M. Gupta and S. Singh. Greening of the internet. In *ACM SIGCOMM 2003*, pages 19–26, 2003.

[2] C. Gunaratne, K. Christensen, S. Suen, and B. Nordman. Reducing the energy consumption of ethernet with an adaptive link rate (ALR). *IEEE Transactions on Computers*, 57(4):448–461, Apr. 2008.

[3] P. Reviriego, J. A. Hernandez, D. Larrabeiti, and J. A. Maestro. Performance evaluation of energy efficient ethernet. *IEEE Communications Letters*, 13(9):697–699, September 2009.

[4] K. Ho and C. Cheung. Green distributed routing protocol for sleep coordination in wired core networks. In *(INC) 2010*, May 2010.

[5] H. Mellah and B. Sansò. Review of facts, data and proposals for a greener Internet. In *Broadnets '09*, Sep. 2009.

[6] W. Chen, D. Jin, and L. Zeng. Design of Multiple Spanning Trees for traffic engineering in Metro Ethernet. In *ICCT 2006*, Nov. 2006.

[7] A. Kolarov, B. Sengupta, and A. Iwata. Design of multiple reverse spanning trees in next generation of ethernet-VPNs. In *(IEEE) Globlecom 2004*, Nov. 2004.

[8] M. Padmaraj, S. Nair, M. Marchetti, G. Chiruvolu, M. Ali, and A. Ge. Metro ethernet traffic engineering based on optimal multiple spanning trees. In *WOCN 2005*, Mar. 2005.

[9] A.F. De Sousa. Improving load balance and resilience of Ethernet Carrier networks with IEEE 802.1S Multiple Spanning Tree Protocol. In ICN/ICONS/MCL 2006.

[10] D. Santos, A. de Sousa, F. Alvelos, M. Dzida, M. Pioro, and M. Zagozdzon. Traffic engineering of multiple spanning tree routing networks: the load balancing case. In *NGI '09*, Jul. 2009.

[11] A. Meddeb. Smart spanning tree bridging for carrier ethernets. In *(IEEE) Globlecom 2008*, Dec. 2008.

[12] G. Mirjalily, F.A. Sigari, and R. Saadat. Best multiple spanning tree in metro ethernet networks. In *ICCEE '09*, Dec. 2009.

[13] https://www.cisco.com/en/US/prod/collateral/routers/ps5763/CRS-3_16-Slot_DS.html.