UNIVERSITÀ DI PISA

Scuola di Dottorato in Ingegneria "Leonardo da Vinci"

Corso di Dottorato di Ricerca in
Ingegneria dell'Informazione

Tesi di Dottorato di Ricerca

The Internet of the Future: Quality of Service and Energy Efficiency

SSD: ING-INF/03

Autore:

Gianfranco Nencioni

Relatori:

Ing. Rosario G. Garroppo

Prof. Stefano Giordano

Prof. Franco Russo

Anno 2012

To my mother.

Manuscript received February 28, 2012

Accepted March 28, 2012

SOMMARIO

Una delle sfide più importanti del ventunesimo secolo nel campo delle reti di telecomunicazioni è la minimizzazione del consumo di energia fornendo congiuntamente una Qualità del Servizio (QoS). Pertanto, di recente la comunità di ricerca ha iniziato a studiare una serie di iniziative miranti a migliorare la QoS e l'efficienza energetica nell'Internet del futuro. In questo scenario, le Service Overlay Network (SON) sono emerse come un modo proficuo per affrontare questi problemi senza modificare l'infrastruttura sottostante. Invece, il Network Power Management (NPM) cerca i metodi che sono in grado di ottenere risparmi energetici sfruttando opportunamentele caratteristiche energetiche dei dispositivi di rete. In questa tesi, viene analizzato il problema della progettazione topologica di una SON dal punto di vista delle prestazioni. Poichè la soluzione analitica del problema è computazionalmente troppo complessa, si confrontano le prestazioni di un insieme limitato di topologie note. Sulla base di euristiche, tre nuove topologie overlay vengono proposte. Attraverso numerose simulazioni, le prestazioni delle topologie overlay candidate vengono valutate in diversi scenari di rete, tenendo conto del carico e del traffico accettato tra i nodi overlay. Inoltre, questa tesi si concentra sul NPM descrivendo quattro problemi di progettazione di rete per ridurre il consumo energetico delle reti attuali e future. I problemi sono risolti per mezzo di risolutori MILP e MINLP, che ottengono delle soluzioni ottimali o approssimate. Dal momento che in scenari di reti di grandi dimensioni questi approcci sono computazionalmente troppo complessi, sono proposte varie euristiche per i diversi metodi di NPM. L'efficacia degli approcci proposti e dell'euristiche viene esplorata in diversi scenari di rete reali, valutando l'impatto di diversi parametri di rete. I risultati mostrano che le topologie SON sono un'ottima scelta perchè mantengono le stesse prestazioni riducendo l'overhead associato. Inoltre, questa tesi mette in luce l'importanza di una buona caratterizzazione del comportamento energetico dei dispositivi di rete. Notevoli risparmi energetici possono essere raggiunti sfruttando le caratteristiche di potenza dei dispositivi. Le euristiche proposte sono in grado di ridurre il tempo di calcolo e di ottenere risparmi energetici comparabili.

ABSTRACT

One of the most important challenges of the twenty-first century in the telecommunication network field is the minimization of the energy consumption jointly providing Quality of Service (QoS). Therefore, recently the research community started to study a set of approaches aimed at improving QoS and energy efficiency in the Internet of the Future. In this scenario, Service Overlay Networks (SONs) have emerged as a profitable way to address these issues without changing the underlying infrastructure. Instead, the Network Power Management (NPM) framework looks for methods which are able to achieve energy savings by suitably exploiting the power consumption features of energy aware network devices.

In this thesis, the topology design problem of a SON from a performance point of view is analyzed. Since the analytical solution of the problem is computationally too complex, we compare the performance of a limited set of well-known topologies. Based on heuristics, three new traffic demand-aware overlay topologies are also proposed. Through extensive simulations, the performance of the candidate overlay topologies is evaluated in different network scenarios, taking into account overhead and accepted traffic between the overlay nodes. Moreover, this thesis focuses on NPM by describing four power aware network design problems for reducing the power consumption of current and future networks. The problems are solved by means of MILP and MINLP solvers, which obtain optimal or approximated solutions. Since these approaches are computationally too complex in large network scenarios, several heuristics are proposed for the different NPM methods. The effectiveness of the proposed approaches and heuristics is explored under different real network scenarios by evaluating the impact of several network parameters. The results show that the proposed SON topologies are a valuables option in the different analyzed scenarios because they maintain the same performance reducing the associated overhead. Furthermore, this thesis highlights the importance of well characterizing the power behavior of the network devices. Notable power savings can be only achieved by exploiting each power aware features. The proposed heuristics can reduce the computational time and obtain comparable power savings.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

LIST OF PUBLICATIONS

International Journals

1. Davide Adami, Christian Callegari, Stefano Giordano, Gianfranco Nencioni, and Michele Pagano, "Design and Performance Evaluation of Service Overlay Networks Topologies", Academy Publisher Journal of Networks (JNW), Vol 6, No 4, April 2011, Pages 556-566, DOI: 10.4304/jnw.6.4.556-566.

International Conferences

2. Rosario G. Garroppo, Stefano Giordano, Gianfranco Nencioni, and Maria Grazia Scutellà, "Network Power Management: models and heuristic approaches", IEEE Global Telecommunications Conference (GLOBECOM 2011), Dec 5-9, 2011, Houston,Texas, USA.

3. Christian Callegari, Rosario G. Garroppo, Stefano Giordano, Gianfranco Nencioni, and Michele Pagano, "Power measurement campaign for evaluating the energy efficiency of current NICs", First International Workshop on Sustainable Internet and Internet for Sustainability (SustaInet), Jun 20, 2011, Lucca, Italy.

4. Rosario G. Garroppo, Stefano Giordano, Gianfranco Nencioni, and Michele Pagano, "Energy Aware Routing based on Energy Characterization of Devices: Solutions and Analysis", 4th International Workshop on Green Communications (GreenComm4), Jun 9, 2011, Kyoto, Japan.

5. Christian Callegari, Rosario G. Garroppo, Stefano Giordano, and Gianfranco Nencioni, "A New Markov Model for Evaluating the ALR Dual-Threshold Policy", 2nd International Workshop on Green Communications (GreenComm2), Dec 4, 2009, Honolulu, Hawaii, USA.

6. Davide Adami, Christian Callegari, Stefano Giordano, Gianfranco Nencioni, and Michele Pagano, "Design and Performance Evaluation of Service Overlay Networks Topologies", International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS 2009), Jul 13-16, 2009, Istanbul, Turkey.

7. Davide Adami, Christian Callegari, Stefano Giordano, Gianfranco Nencioni, and Michele Pagano, "Topology Design of a Service Overlay Network for e-Science Applications", INGRID 2009, Apr 1-3, 2009, Alghero, Italy.

Other works

8. Gianfranco Nencioni, Rosario G. Garroppo, Stefano Giordano, and Maria Grazia Scutellà, "Power Aware Routing and Network Design with Bundled Links", Conference of Gruppo nazionale Telecomunicazioni e Teoria dell'Informazione (GTTI 2011), Jun 20-22, 2011, Taormina, Italy.

CHAPTER 1

INTRODUCTION

1.1 Motivations

In the last decade, with the tremendous growth of multimedia technologies and the increasing popularity of real-time applications, the support of Quality of Service (QoS) in the Internet has been in a great demand and several architectures to provide QoS (e.g., Integrated Services (IntServ [18] and Differentiated Services (DiffServ) [16]) have been proposed by the IETF (Internet Engineering Task Force). Nevertheless, the deployment of these approaches is unlikely to be feasible in the long run, because IntServ has scalability problems, and DiffServ can only provide QoS with very coarse granularity. Moreover, appropriate business models are difficult to be introduced, because the Internet Service Providers (ISPs) are only concerned with providing QoS in their own administrative domains. As a result, the current Internet basically provides only a best-effort packet delivery service.

Though, for provisioning a QoS-like features, current current networks are widely overprovisioned. Considering that the current network equipments are not energy aware, i.e. they consume always the same energy when they are underutilized or not. The energy saving has become one of the most important challenges of the twenty-first century for environmental and economical reasons. From an environmental perspective, due to the lack of diffusion and efficiency of renewable energy, the reduction of power consumption is important because the production of energy is directly related to the emission of carbon dioxide (CO_2), the main reason of global warming. From an economical point of view, the incremental growth of the energy price and of the power demand of emerging countries makes energy saving a key issue to contain the increase of energy cost in various sectors (industrial, commercial, residential). Furthermore, we are witnessing an explosive growth in the use of Information and Communication Technologies (ICTs) equipment that is rapidly becoming a major consumer of energy.

Indeed, according to the SMART 2020 study, CO_2 emissions from ICT are increasing at a rate of 6% per year and with such a growth rate they could represent 12% of worldwide emissions by 2020 [30]. These trends are confirmed by other studies. As an example, the European Commission DG INFSO report [25] estimated European telcos and operators to have an overall network energy requirement equal to 14.2 TWh in 2005, which will rise to 21.4 TWh in 2010, and to 35.8 TWh in 2020 if no green network technologies will be adopted. Furthermore, the power consumption of Verizon during 2006 was 8.9 TWh (about 0.26% of USA energy requirements) [47], while the NTT group reports that the amount of electric power in fiscal year 2004 needed for telecommunications in Japan was 4.2 TWh [34].

In this scenario, the research community is studying a set of approaches for improving the energy efficiency in the Future Internet. Detailed and up to date surveys on the different approaches investigated for an energy efficient networking are presented by the authors of [17] and [55]. Recent works on energy efficient networks have defined energy aware problems and have proposed solutions on two relevant aspects: the *Network Device Design* and the *Network Power Management*.

The Network Device Design consists in devising energy efficient mechanisms implemented in network equipments, which leads to the design of energy aware network devices [32] [36]. In this field, we can mention two main approaches:

- the *resource adaptation* that attempts to reduce energy consumption by adapting the power consumption to the provided resources (processing, bandwidth, etc.)
- the *low power mode* that provides the maximum resources during the active period to maximize the idle period when the device is switched to a state characterized by a very low power consumption.

Examples of these approaches are Adaptive Link Rate (ALR) [32] and Low-Power Idle (LPI) [36] respectively, two mechanisms for reducing power consumption of Ethernet Network Card.

The Network Power Management (NPM) is expressed by methods aimed to achieve further energy savings by means of appropriate strategies, in terms of protocols and network control/management, that exploit the power consumption features of energy aware network devices.

If energetic improvements can be achieved by new devices and management techniques realized from scratch, the QoS improvements have to be obtained by means of solutions with a minimum impact on current and future Internet. Concerning this, Service Overlay

Networks (SONs) have emerged as a profitable way to provide QoS. A SON consists in a virtual network built on top of an IP network. A SON is composed of overlay nodes, which can be customized so as to build in complex features and to cooperate with each other in order to provide new services without any change to the routers of the underlying IP network. The overlay nodes can be connected each other by means of overlay link, which are logical links and really are IP-layer paths, usually composed of one or more physical links. Moreover, an overlay path consists of one or more overlay links.

1.2 Main contributions

This thesis is specifically focused on the design of SON topologies and on the study of NPM methods.

In particular, the topology design problem of a SON from a performance point of view is addressed. Since the analytical solution of the problem is computationally too complex, the performance of a limited set of well-known topologies is compared. Based on heuristics, three new traffic demand-aware overlay topologies are proposed. Through extensive simulations, the performance of the candidate overlay topologies is investigated in different network scenarios, taking into account overhead and accepted traffic between the overlay nodes.

Moreover, this thesis focuses on NPM by describing four power aware network design problems, with related mathematical models, for reducing the power consumption of current and future networks. Each problem is based on different characterizations and power awareness of the network devices, leading either to Mixed Integer Linear Programming (MILP) models or to Mixed Integer Nonlinear Programming (MINLP) models. Given that these problems are NP-hard, several heuristics of the NPM methods are also. The effectiveness of the proposed approaches and heuristics is explored by performing extensive simulations under different real network scenarios and evaluating the impact of several network parameters.

1.3 Outline

The remainder of this thesis is structured as follows.

Chapter 2 addresses the topology design of SON. In particular, the basic concept of SON are outlined and the proposed topology design problem is formalized. Moreover, some existing SON topologies as well as the new overlay topologies are presented. Finally, a simulative

analysis to investigate the impact of the overlay topologies on performance and overhead is presented.

Chapter 3 introduces the problem of NPM. In particular, a set of general NPM models is presented and several heuristic approaches are proposed.

Chapter 4 analyzes the performance of the NPM models and the heuristics presented in the previous chapter. In particular, the effectiveness of the proposed methods and approaches is explored under different real network scenarios, by evaluating the impact of several network parameters.

Chapter 5 finally draws some conclusions for this thesis and discusses open issues and further perspectives for this research field.

CHAPTER 2

SERVICE OVERLAY NETWORKS

2.1 Background

In recent years, Service Overlay Networks (SONs) have emerged as a profitable way to facilitate the deployment of QoS-sensitive applications, such as VoIP, videoconference, online gaming, etc. A SON is a virtual network built on top of an IP network, whose nodes, the overlay nodes, can be customized so as to build in complex features and to cooperate with each other in order to provide new services without any change to the routers of the underlying IP network. Pairs of overlay nodes are connected by means of logical links, also known as overlay links, which really are IP-layer paths, usually composed of one or more physical links. Moreover, an overlay path consists of one or more overlay links. A distinguishing characteristic of SONs is that the overlay links could be overlapped at the physical layer even though they are completely separated at the overlay layer. Non-overlay traffic or other overlay links may pass through a part or a whole group of IP-layer links. This means that the overlay links capacities are not fixed and cannot be controlled by the overlay nodes. To obtain satisfactory performance, the overlay nodes need continuously probing the network, so as to obtain updated information on the status and performance of the overlay links.

SONs can be classified as End-user Overlay Networks and Backbone Overlay Networks. The first type of SON [12] [11] is constructed among the end hosts without any support from the intermediate network nodes. Though highly flexible, End-user Overlay Networks cannot guarantee end-to-end QoS, since the overlay links typically cross many intermediate Autonomous Systems (ASs), and the uncontrolled peering structure is unlikely to provide direct QoS support to the end-users. Due to these difficulties, the second type of overlay networks was introduced in several research works [24] [31] [48] [41] [40]. Usually managed by third parties, Backbone QoS Overlays may be classified in two different categories, based

on the overlay nodes location: either only at the edge (see [24] [31] [40]), or also in the core of different domains (see [48] [41]).

To optimize the performance and profitability of a SON, the first step is the selection of the best topology connecting the overlay nodes. When considering different topologies, it is necessary to understand how they affect the overlay routing performance and how to efficiently build overlay topologies connecting all the overlay nodes. Several works have focused on the selection of the best overlay links (e.g. [52]), but other issues, such as binding end systems to overlay access nodes [52], positioning the overlay nodes [35] [19] or choosing the right number of overlay nodes [19], have also been faced. In these studies, the overlay topology is usually represented as a graph and the topology design problem is expressed as an optimization problem. The general approach relies on the use of heuristic algorithms that allow finding a near-optimal solution.

Most works [52] [35] [26] [49] [57] analyze the topology design problem from an economic point of view with the aim to minimize the cost for the deployment of the SON. Only a few works [39] [42] deal with the SON topology design problem from a network performance perspective. In [39], the authors aim at finding the overlay topology minimizing a cost function which takes into account the overlay link creation cost and the routing cost. They also highlight how the traffic demand affects the creation of new overlay links. In [42], instead, the authors compare several existing and some new overlay topologies in terms of resilience.

In this thesis, the topology design problem is formalized by taking into account the traffic demand among the overlay nodes as performance metric to be optimized. Since this problem is computationally too complex, we investigate the performance and overhead of several candidate SON topologies by varying the number of overlay nodes and the IP network size. More specifically, three different models for the topology at IP-layer (flat, hierarchical, real) are taken into account and the performance of seven overlay topologies built on top of those IP networks is compared: four topologies are well-known since they have been already introduced in the literature, whereas the other three (K-Shortest-Path-Tree, Pruned Adjacent-Connection and Demand-aware Adjacent- Connection) are completely new.

In the performance evaluation, the best overlay topology is considered the one that, on equal terms of accepted traffic and performance, has the lowest overhead, due to the overlay network maintenance traffic.

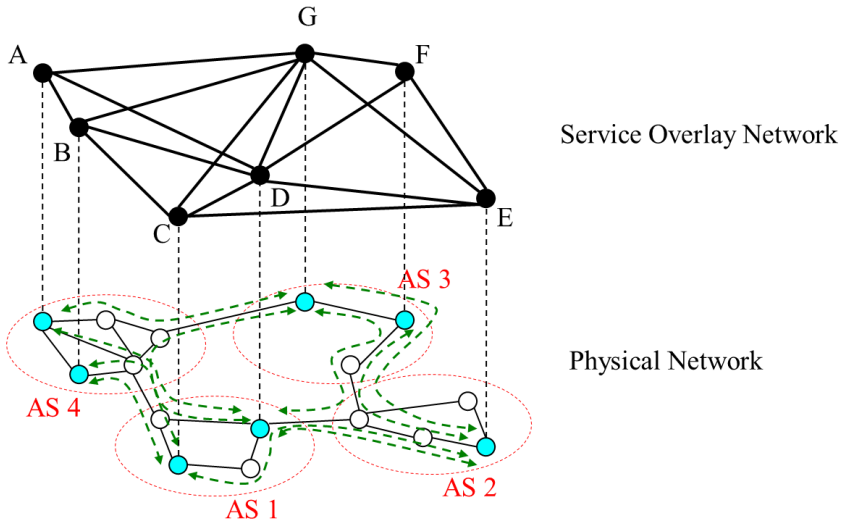


Figure 2.1: *Backbone QoS Overlay with Intra-domain nodes*

2.2 Problem Statement

This thesis is focused on Backbone QoS Overlays with intra-domain nodes (Figure 2.1), where the overlay nodes can be placed either at the edge or in the core of each domain.

In the following, a more detailed description of this architecture is not provided, because it is out of scope. The chosen architecture, indeed, is only an instance used to evaluate the overlay routing performance when varying the overlay topology.

The experimental results obtained in this thesis are also valid in the case of Backbone QoS Overlays (see Figure 2.2) with Inter-domain nodes.

In the following, we suppose that:

- The location of the overlay nodes is pre-determined.
- The metric associated to each overlay link is the delay and the overlay path between a pair of overlay nodes is selected by using the Dijkstra algorithm.
- Each overlay path is composed of IP-layer links. At IP layer, the cost of each link is assigned in inverse proportion to the link bandwidth and the shortest path between a pair of IP nodes is computed by using the Dijkstra algorithm.

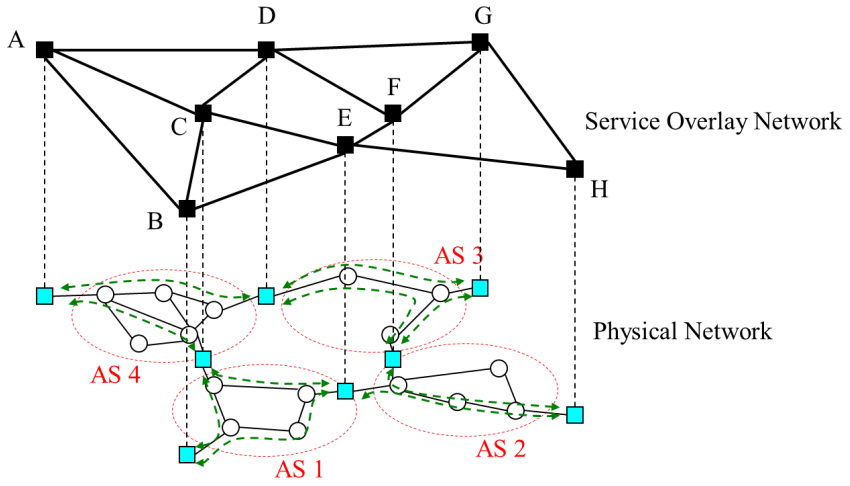


Figure 2.2: *Backbone QoS Overlay with Inter-domain nodes*

Therefore, the overlay topology construction problem can be formulated as follows. Let us consider:

- The IP-layer topology $G_P(V_P, E_P)$ where
  - V_P is the set of nodes.
  - E_P is the set of IP layer links.
  - N is the total number of V_P nodes.
- A set of overlay nodes, V_O, which is a subset of V_P.
  - M is the total number of V_O nodes.
- The IP-layer path-link indicator function P_{ij}^{mn}, where $P_{ij}^{mn} = 1$ if the IP-layer path between m and n includes the IP-layer link ij.
- The overlay path-link indicator function Q_{mn}^{xy}, where $Q_{mn}^{xy} = 1$ if there is an overlay path from x to y that includes the overlay link mn.
- The delay of the IP-layer link ij, D_{ij}.
- The traffic demand between the overlay nodes x and y, T_{xy} .

The goal is to find the topology $G_O(V_O, E_O)$ (i.e., a sub-set of overlay links, E_O) that minimizes the cost function:

$$\sum_{ij} T_{xy} \sum_{mn} Q_{mn}^{xy} \sum_{ij} D_{ij} P_{ij}^{mn} \tag{2.1}$$

in which the end-to-end delay is weighted by the traffic demand, with the constraints:

- $\sum_{mn} e_{o_{mn}} \leq d$, where d is the maximal node degree and $e_{o_{mn}} = 1$ if an overlay link between m and n exists, $e_{o_{mn}} = 0$ otherwise.
- $\sum_{mn} Q_{mn}^{xy} \sum_{ij} D_{ij} P_{ij}^{mn} \leq \delta$, where δ is the overlay paths delay constraint.
- $\sum_{xy} T_{xy} Q_{mn}^{xy} \leq \tau_{mn}$, where τ_{mn} is the available bandwidth of the overlay link mn.

It is worth highlighting that if popular or greedy nodes exist, the topology layout does not change, but such nodes are located at the centre, so as to minimize their latency with respect to the other nodes (see [43] for further details).

The degree constraint attends to limit the overhead associated to the overlay link monitoring (e.g., delay, available bandwidth, overlay traffic demand). Thus, if the node degree is higher, the overhead is greater.

2.3 Overlay Network Topologies

The problem formulated in the previous section is NP-hard and therefore it is too complex from a computational point of view. The basic idea behind our approach is to interconnect the overlay nodes by means of the most common network topologies and to choose the best topology with respect to a set of performance metrics.

This section lists the main features of several candidate topologies that appeared in literature. Moreover, three new overlay topologies are proposed.

2.3.1 Full-Mesh (FM)

In the FM overlay network topology, each node is adjacent to all the other nodes at overlay layer. Therefore, every node has the same number of neighbours, i.e. the same node degree (d). As said previously, to retrieve information on the status and performance of the overlay links, every node has to periodically send probing packets to all the neighbours at overlay

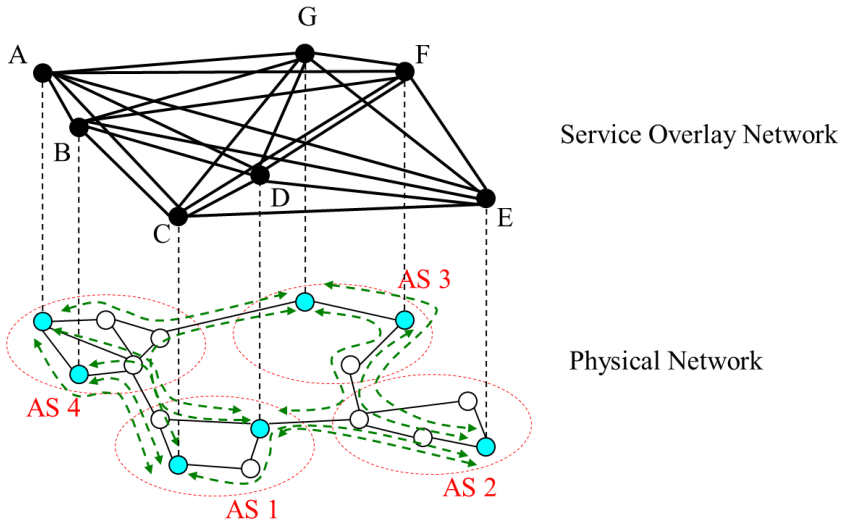


Figure 2.3: *Full-mesh overlay topology*

layer. The FM topology is therefore characterized by the highest overhead, due to the overlay monitoring traffic. An example of FM topology is shown in Figure 2.3.

2.3.2 K-Minimum Spanning Tree (KMST)

A minimum spanning tree (MST) is the lowest cost tree among all the candidate trees connecting a given set of nodes. A KMST overlay topology is composed of K MSTs, where the k-th MST of the composite graph is the MST of the initial graph excluding the edges of the previously computed MSTs. Therefore, the overlay links of the K trees are not overlapped. The K value can be chosen as a trade-off among cost, performance and node degree constraint. This approach has been proposed in [54] so as to minimize the overhead due to the amount of information exchanged for link monitoring. Figure 2.4 shows a 2MST, where the two trees are depicted with dashed and solid lines, respectively.

2.3.3 Mesh-Tree (MT)

This topology (called Mesh-Tree in [42]) is obtained by joining the MST connecting all the overlay nodes with the set of overlay links connecting the overlay nodes that have a

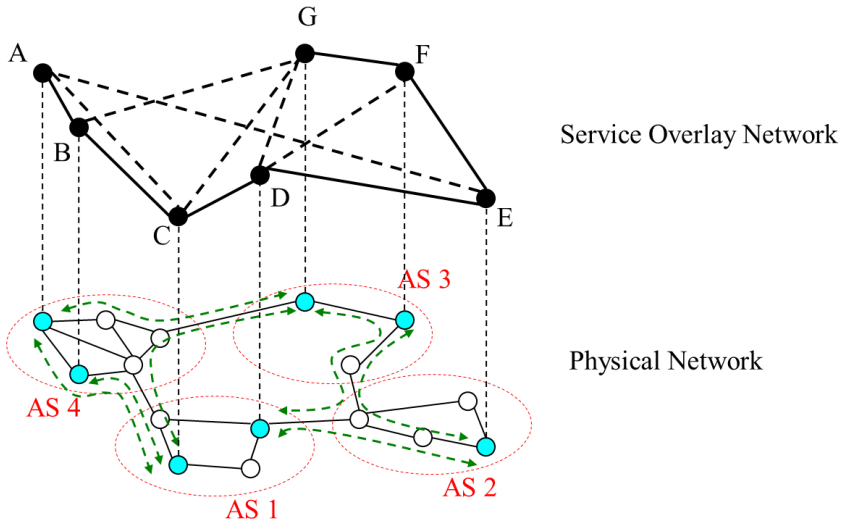


Figure 2.4: *Two-minimum-spanning-tree overlay topology*

grandchild-grandparent or uncle-nephew relationship in the MST. Figure 2.5 shows an MT overlay topology: the MST is represented with solid lines, whereas links joining nodes with a grandchild-grandparent or uncle-nephew relationship are dashed.

2.3.4 Adjacent-Connection (AC)

The Adjacent Connection (AC) topology relies on the knowledge of the IP-layer topology. The construction of the AC topology, proposed in [41] and [44], is based on the following condition: if no overlay node is directly connected to the nodes belonging to the IP-layer path between any pair of overlay nodes, an overlay link connecting this pair of overlay nodes is created. Figure 2.6 shows an example of AC overlay topology.

2.3.5 K-Shortest Path Tree (KSPT)

The K-Shortest Path Tree (KSPT) is first novel topology introduce in this thesis. It is constructed taking into account the traffic demand among the overlay nodes. A SPT is made up of minimum cost paths from a node (root) towards all the other nodes. In this case, the metric is the overlay link delay. The choice of the K value, like in KMST, results from a trade-off

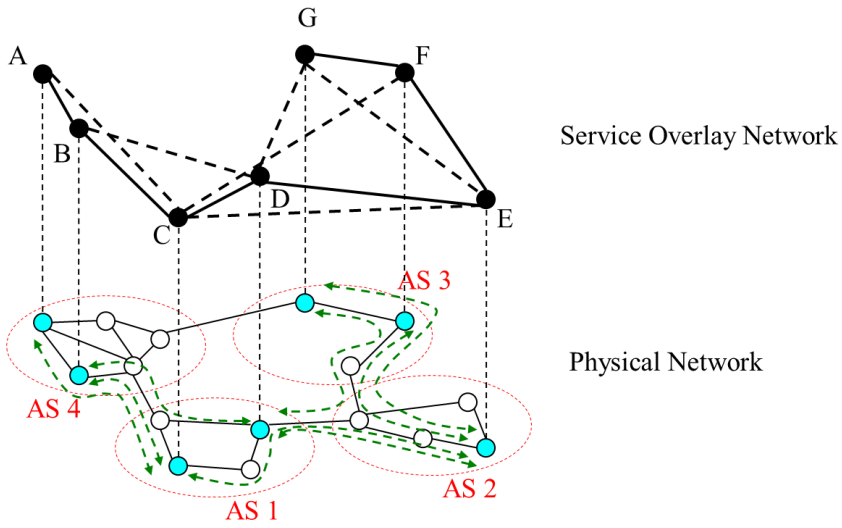


Figure 2.5: *Mesh-tree overlay topology*

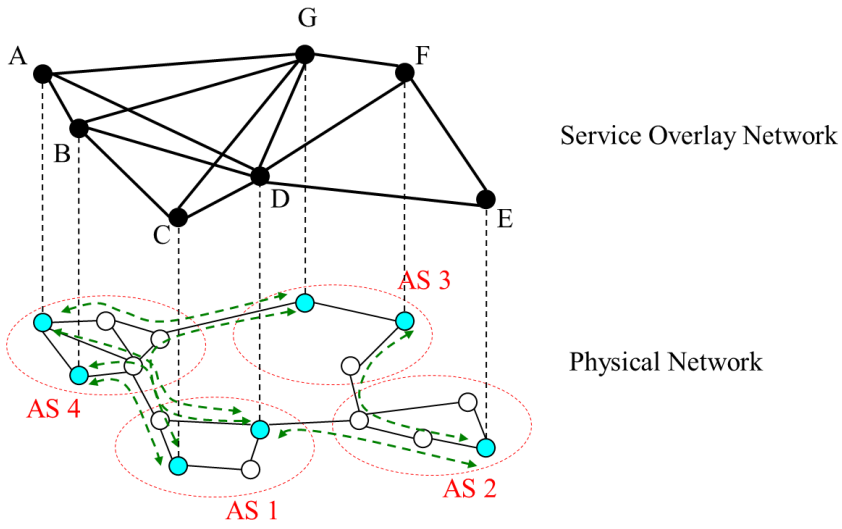


Figure 2.6: *Adjacent-connection overlay topology*

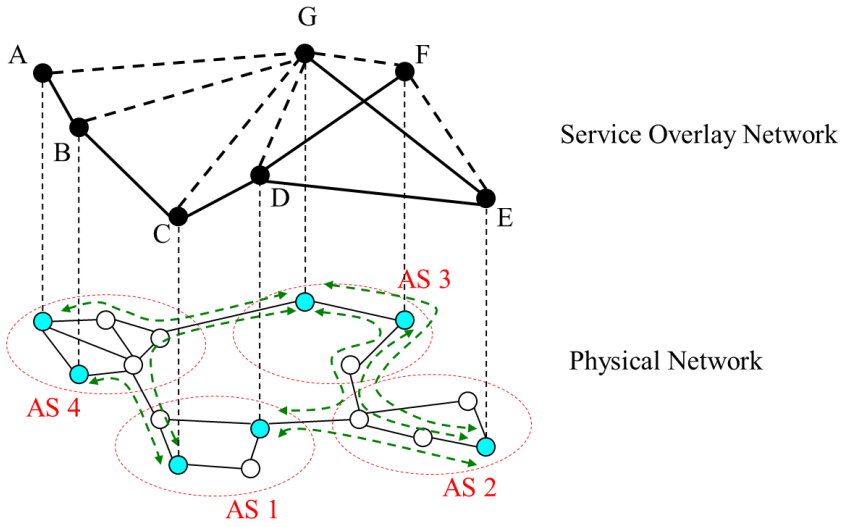


Figure 2.7: *Two-shortest path tree overlay topology*

among cost, performance and node degree constraints.

A KSPT topology can be constructed as follows:

1. Initialize $K_{Th} = ceiling[M/3]$, $K = 0$.
2. Create a SPT for each overlay node.
3. Select the SPT whose root corresponds to the overlay node with the greatest traffic demand.
4. Add the SPT to the topology layout. Increment the value of K ($K = K + 1$).
5. Calculate the average node degree. If it is greater than the node degree constraint or K_{Th} is reached, end. Otherwise, continue.
6. Select the SPT with minimum overlap with the created topology. Go to step 4.

Figure 2.7 shows a 2SPT, where the two trees, with roots G and C, are depicted by using dashed and solid lines, respectively.

2.3.6 Pruned Adjacent-Connection (PAC)

The Pruned Adjacent-Connection (PAC), is a novel topology, which is proposed and constructed for a real network scenario by exploiting the previous results obtained in flat and hierarchical network scenarios. PAC is a modified version of the AC topology, where some overlay links are deleted so as to reduce the node degree. In more detail. an overlay link is deleted if it connects two nodes with the highest degree. The number of overlay links to be deleted is determined by a fixed degree threshold, T_h, which is proportional to M.

A PAC topology can be constructed as follows:

1. Create an AC topology and determine the degree of each overlay node.
2. Select the overlay node, i, with the highest degree.
3. Select the neighbour of i, j, with the highest degree.
4. Delete the overlay link between i and j, and update the node degree.
5. Calculate the average node degree. If it is greater than the threshold T_h, go to step 2. Otherwise, end.

Figure 2.8 shows an example of PAC overlay topology.

2.3.7 Demand-aware Adjacent-Connection (DAC)

As the previous one, the Demand-aware Adjacent-Connection (DAC), is a new topology aimed at a real network scenario as PAC. However, DAC allows to take also into account the traffic demand among the overlay nodes. DAC, like PAC, is a modified version of the AC topology, where some overlay links are deleted to reduce the node degree. In more detail, an overlay link is deleted if it interconnects two nodes with the highest traffic demand. Also in this case, the number of overlay links to be deleted is determined by a fixed degree threshold, T_h, proportional to M.

A DAC topology can be constructed as follows:

1. Create an AC topology, determine the degree of each overlay node, and calculate the aggregate traffic demand of each overlay node, as the sum of the incoming and outgoing traffic.

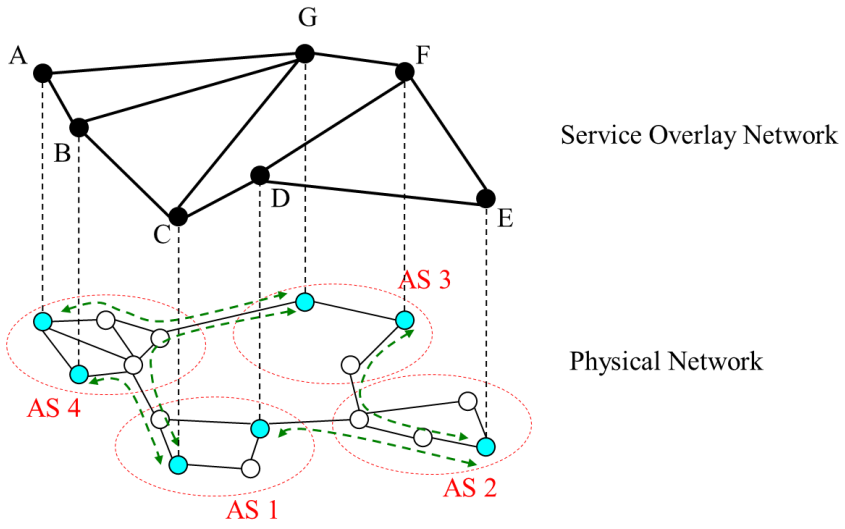


Figure 2.8: *Pruned adjacent-connection overlay topology*

2. Select the overlay node, i, with the lowest aggregate traffic demand and whose degree is greater than the threshold T_h.

3. Select the neighbour of i, j, with the lowest aggregate traffic demand and with a node degree greater than the threshold T_h.

4. Delete the overlay link between i and j, and update the node degree.

5. Calculate the average node degree. If it is greater than the threshold T_h and i and j exist, go to step 2. Otherwise, end.

Figure 2.9 shows an example of DAC overlay topology.

2.4 Performance Analysis

2.4.1 Simulations Settings

To compare the performance of the topologies described in the previous section, flat and hierarchical IP- layer topologies generated by means of BRITE [2] are considered.

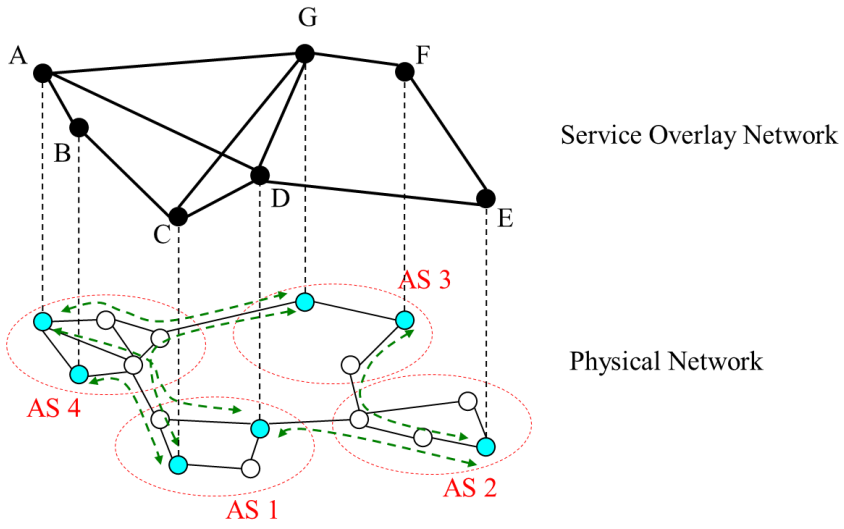


Figure 2.9: *Demand-aware adjacent-connection overlay topology*

In the flat topology, the generation model to interconnect the nodes is based on the Waxman's probability [53]:

$$P(u,v) = \alpha \cdot e^{\frac{-\delta}{\beta \cdot L}} \tag{2.2}$$

where $P(u,v)$ is the probability that a link between the nodes u and v is created, α ($0 < \alpha \leq 1$) and β ($0 < \beta \leq 1$) are Waxman specific parameters (in our simulations $\alpha = 0.03$, $\beta = 0.03$), δ is the Euclidean distance between the nodes, and L is the maximum distance between any two nodes. The link bandwidth has been assigned according to a uniform distribution in the range $[10, 1000] Mb/s$. It is worth highlighting that this value does not represent the IP-layer link capacity, but the available bandwidth, since non-overlay traffic may pass through the same links. The link delay has been assigned in proportion to the link length.

To create a hierarchical random topology, a top-down approach that consists of the following three steps has been adopted:

1. Generate an AS-level topology according to the Waxmans model (for redundancy reasons, we set $\alpha = 1$).

2. For each AS, generate a router-level topology by using the Waxmans model with the

same parameters as for the flat topology.

3. Finally, use the random method to interconnect ASs asc dictated by step 1: if (i, j) is a link in the AS-level topology, then two nodes (u, v), randomly chosen from ASs (i, j), are interconnected at $512Mb/s$.

Simulations have been carried out randomly associating overlay nodes with the generated IP-layer nodes and interconnecting them through one of the topologies described in Section 2.3.

For a fair comparison of the candidate topologies and to also meet the node degree constraint d, other research works modify the construction of the topology as follows: if an overlay node degree exceeds d in the resulting overlay topology, only the d closest neighbours overlay nodes are maintained, while the others are pruned. Since this approach dramatically affects the intrinsic nature of each topology, the overlay node degree is considered as a primary performance metric without changing the resulting overlay topology.

Moreover, the traffic demand is generated according to a uniform distribution normalized with respect to M, so that the overall traffic demand is statistically the same when the number of overlay nodes changes. A customized simulation environment has been developed by using C programming language. Three simulation scenarios have been considered:

1. The IP-layer topology is flat [$N = (100, 150, 200, 250, 300)$] and the number of overlay nodes changes [$M = (10, 20, 40, 60, 80\%) \cdot N$].

2. The IP-layer topology is hierarchical [$N = (100, 200, 300, 400, 600)$, Number of ASs=4, Number of nodes per-AS=(25, 50, 75, 100, 150)] and the number of overlay nodes changes [$M = (10, 20, 40, 60, 80\%) \cdot N$].

3. A case study, corresponding to a "real network scenario" with a large hierarchical IP-layer topology [N=600, Number of ASs=4, Number of nodes per AS=150] and a small number of overlay node [$M = (1, 2, 4, 6, 8\%) \cdot N$].

The average value and the 99% confidence interval of each performance metric have been evaluated in the three scenarios by performing a set of 50 independent simulations for each pair (N, M).

2.4.2 Performance Metrics

To compare the performance of different overlay topologies, four parameters are introduced, each of them highlighting a specific topology feature.

- *Bandwidth Rejection Ratio (BRR)*

This performance parameter represents the probability that an overlay path with bandwidth and delay requirements can not be created due to bandwidth unavailability:

$$BRR = \frac{\text{Number of bandwidth rejected end-to-end overlay paths}}{\text{Total number of end-to-end overlay paths}} \tag{2.3}$$

- *Delay Rejection Ratio (DRR)*

This performance parameter represents the probability that an end-to-end overlay path with bandwidth and delay requirements can not be created due to delay limitations although the bandwidth constraint could be satisfied:

$$DRR = \frac{\text{Number of delay rejected end-to-end overlay paths}}{\text{Number of bandwidth accepted overlay paths}} \tag{2.4}$$

- *Average Node degree (AND)*

This performance parameter provides information on the topology overhead. If d_i is the neighbours number of the i-th overlay node, AND can be defined as follows:

$$AND = \frac{\sum_{i=1}^{M} d_i}{M} \tag{2.5}$$

- *Accepted Traffic Weighted Delay (ATWD)*

This performance parameter describes the capability of the topology to associate low delays to the highest traffic demands:

$$ATWD = \frac{\sum_{ij}(\text{Accepted traffic demand between the nodes } i,j \times \text{Delay}_{ij})}{\text{Total accepted traffic}} \tag{2.6}$$

It is worth emphasizing that the ATWD is strictly related to the cost function (2.1) defined in Section 2.2. Indeed, the only difference is that the ATWD is normalized with respect to the accepted traffic. This normalization is necessary for a fair comparison, because the accepted traffic varies with the overlay topology.

2.4.3 Simulations Results

A different graph has been worked out for each performance metric in case of flat and hierarchical topologies. In the following, the simulation results concerning AND, BRR and DRR are reported only for the largest size of the IP-layer network, whereas as regards ATWD the results obtained for the smallest size of the IP-layer network are also reported, since the ATWD performance significantly varies with the number of IP-layer nodes. Since the experimental results obtained in the flat topology cases (see the following sub-section), show that KSPT outperforms the other topologies, the simulations with PAC and DAC topologies are only performed when the IP-layer network is hierarchical and in the real network scenario. The behaviour of both topologies is analyzed with two values of the degree threshold: $T_h = (M/3, M/2)$ in the hierarchical IP-layer network casel, $T_h = (M/2, 2 \cdot M/3)$ in the real network scenario case, respectively.

Flat IP-layer Network Model

Figure 2.10 shows the AND trend in a flat IP-layer network ($N = 300$) when M ranges from $0.10 \cdot N$ to $0.80 \cdot N$. It is worth highlighting that the behaviour of this metric is not affected by the size of the IP-layer network, so these results may also be extended to IP-layer networks with a different number of nodes.

The graph outlines that MT always has the minimum node degree that remains constant independently of M.

Moreover, the AND of KMST and KSPT increases almost linearly with M, but KSPT outperforms KMST due to the different way K is selected and the mechanism used to limit the value of K when the AND is too high.

Concerning the AC overlay topology, at the beginning, the AND is the highest one and increases up to reaching a maximum value when $M = 0.20 \cdot N$. Afterwards, it starts decreasing and, when $M \geq 0.60 \cdot N$, AC outperforms not only KMST, but also KSPT. Indeed, when M increases, it is more likely that another overlay node exists along the path between whichever pair of overlay nodes and therefore the number of overlay links decreases. The AND of FM (not reported in the graph) is always equal to $M - 1$.

Figure 2.11 reports the BRR average value and confidence interval for $N = 300$. As outlined in the graph, FM, KMST, AC and KSPT show a similar behaviour. MT, instead, always has the highest BRR. This result is in accordance with the node degree trend: since MT has the lowest node degree, also the overall available bandwidth is the lowest.

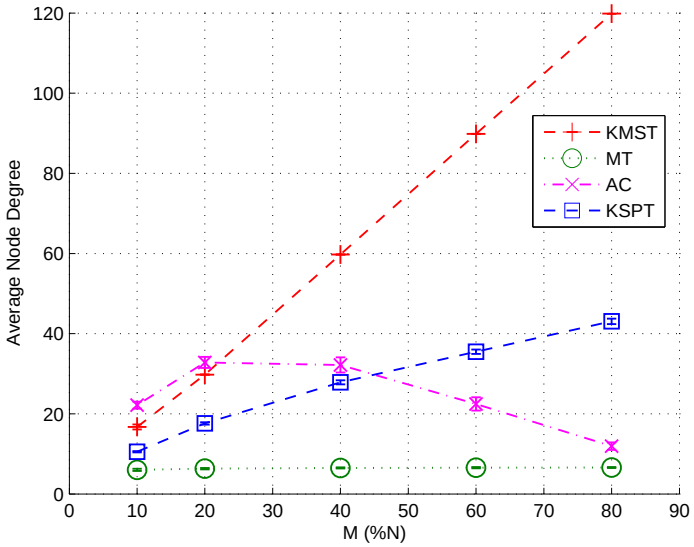


Figure 2.10: *AND for a flat IP-layer network with N=300*

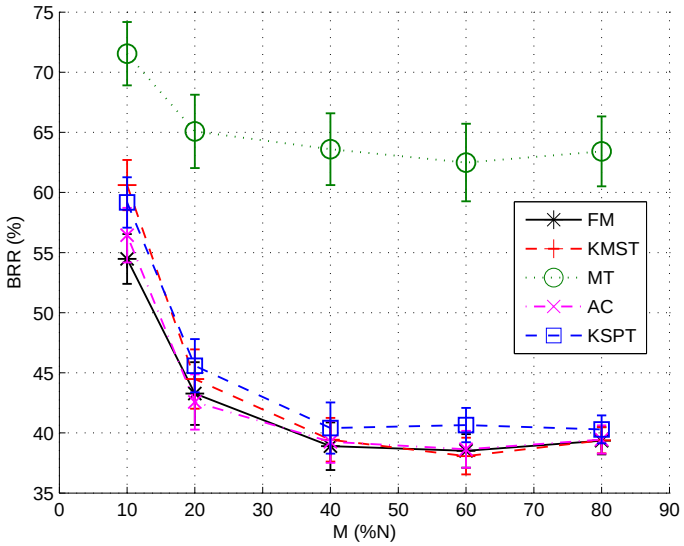


Figure 2.11: *BRR for a flat IP-layer network with N=300*

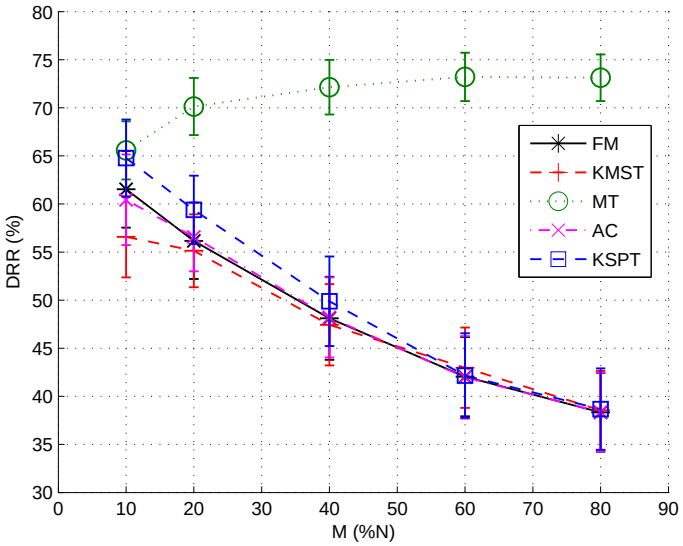


Figure 2.12: *DRR for a flat IP-layer network with N=300*

Figure 2.12 shows the DRR trend. Also in this case, FM, KMST, KSPT and AC show a similar trend that decreases with M. Instead, MT has the worst performance, since its DRR values are always significantly higher than the ones obtained for the other topologies when $M \geq 0.20 \cdot N$. This is due to the lower node degree of the MT topology.

Finally, let us consider the ATWD parameter. By definition, this metric has low values when the overlay topology favours the creation of overlay paths with the lowest delay between the overlay nodes that exchange the largest amount of traffic. In a small size flat IP-layer network (see Figure 2.13), KSPT and MT perform better than the other topologies. On the contrary, as shown in Figure 2.14, in a large size flat IP-layer network MT sharply outperforms FM, KMST, KSPT and AC.

From the simulation results, we can infer that MT performs worse than the other overlay topologies because, although it is characterized by the lowest AND and ATWD metrics, the BRR has the highest values (see Figure 2.11) and therefore most of the traffic demand is rejected. In flat IP-layer networks, the overlay topology should be chosen based on the number of overlay nodes. If the number of both overlay and IP nodes is small, KMST is the best overlay topology. Instead, if the number of overlay nodes is small, but the IP-layer

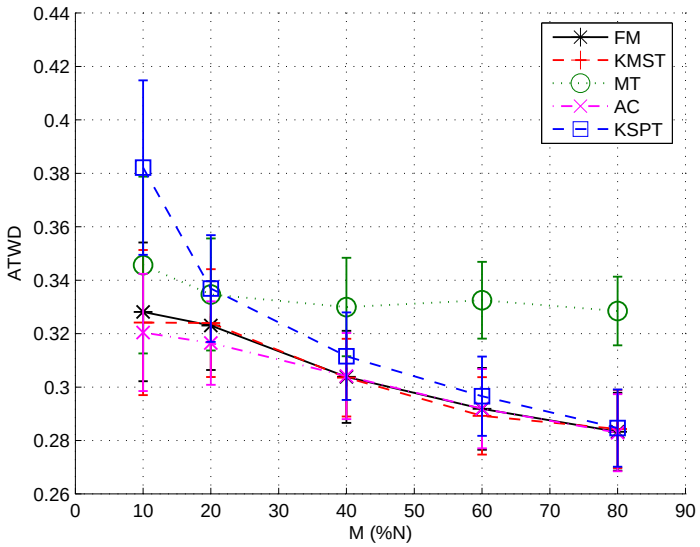


Figure 2.13: *ATWD for a flat IP-layer network with N=100*

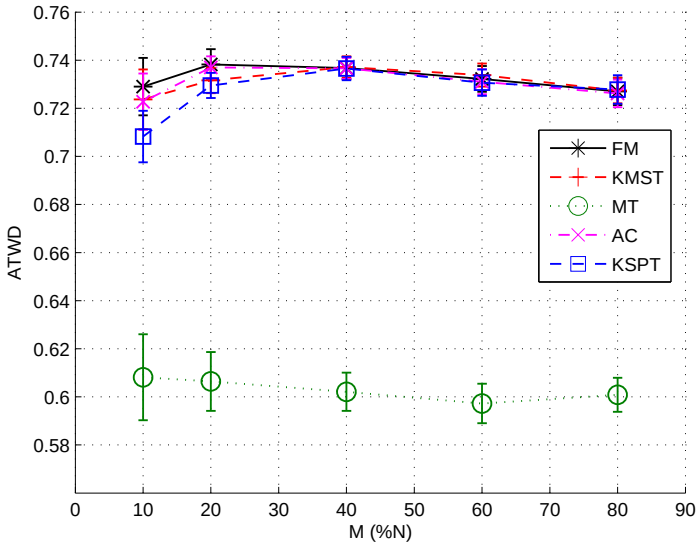


Figure 2.14: *ATWD for a flat IP-layer network with N=300*

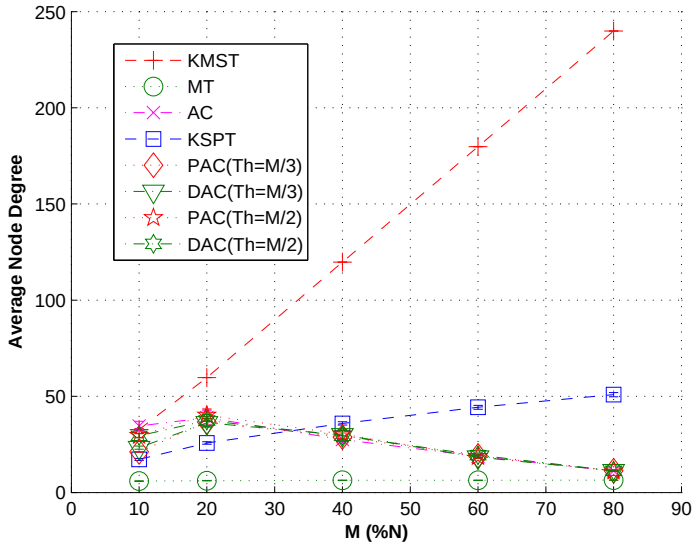


Figure 2.15: *AND for a hierarchical IP-layer network with N=600*

network size is large, the best overlay topology is KSPT. When the number of overlay nodes is large, AC is always the best overlay topology, because it performs in the same way as the other overlay topologies, but its AND is the lowest.

Hierarchical IP-layer Network Model

In a hierarchical IP-layer network, the AND behaviour (see Figure 2.15) is similar to that obtained for a flat IP-layer network. Nevertheless, the graph outlines that AC always has a lower AND than KMST. PAC and DAC allows reducing the AND only when $M = 0.10 \cdot N$ and the reduction is in inverse proportion to T_h.

Figure 2.16 and Figure 2.17 show the BRR and DRR behaviours for $N = 600$, respectively. Also in this case, the trends are similar to those obtained for flat IP-layer topologies and MT performs worse than all the other candidate topologies. However, it is relevant to highlight that, apart from MT, KSPT has the highest BRR and DRR and the performance difference among KSPT and FM, KMST, AC is greater for small and middle size overlay networks.

The BRR of PAC and DAC is the same as FM, KMST, and AC, instead the DRR is higher. When $T_h = M/3$ PAC and DAC perform worse than KSPT. Basically, DAC performs better

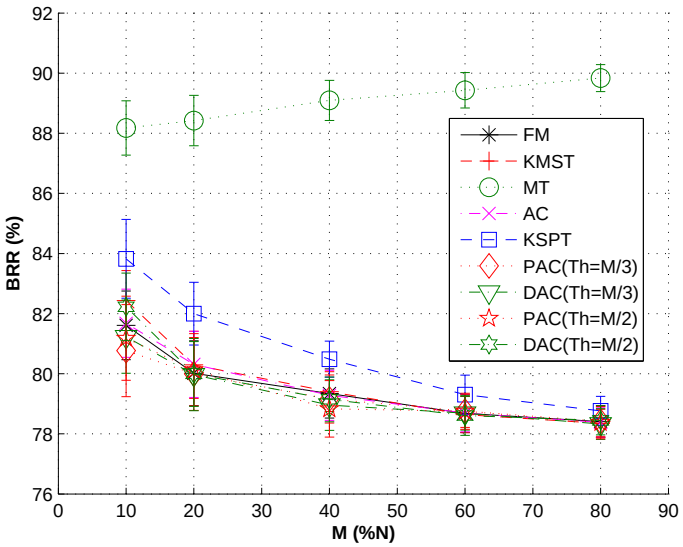


Figure 2.16: *BRR for a hierarchical IP-layer network with N=600*

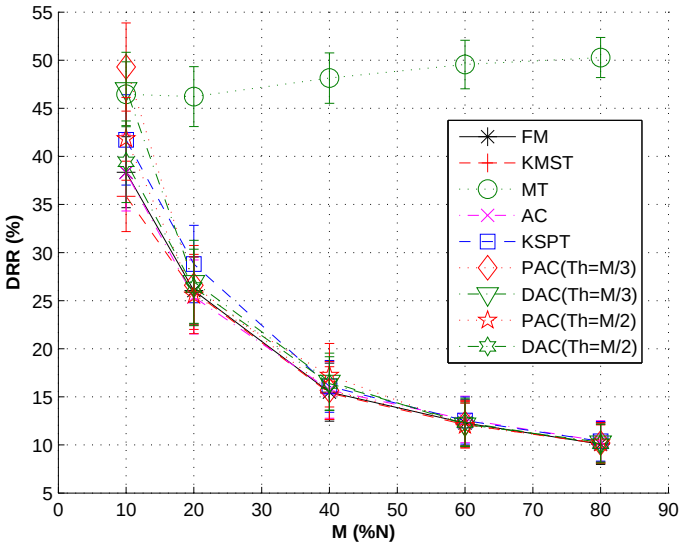


Figure 2.17: *DRR for a hierarchical IP-layer network with N=600*

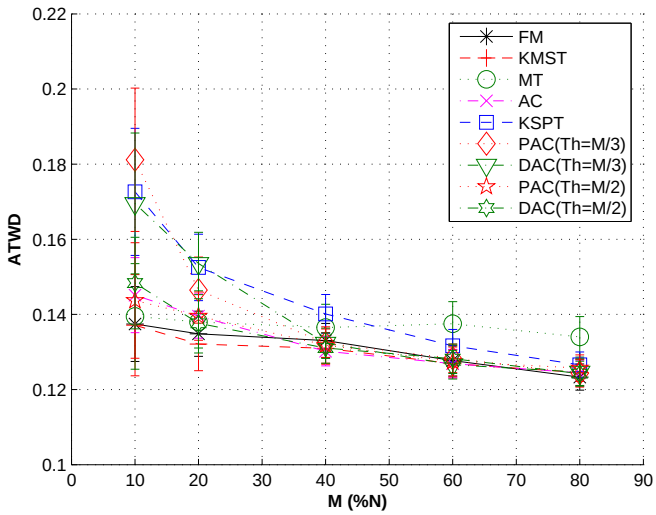


Figure 2.18: *ATWD for a hierarchical IP-layer network with N=100*

than PAC, while DAC ($T_h = M/2$) performs like FM, KMST, and AC.

Figure 2.18 and Figure 2.19 highlight that the ATWD parameter behaviour in small and large size hierarchical IP-layer networks, respectively, shows some differences with respect to the flat topology. Indeed, although the number of IP-layer nodes is increased, MT and KSPT do not exhibit any performance improvement as it occurred in case of a flat topology.

PAC and DAC perform better in small size hierarchical IP- layer networks and with a higher degree threshold.

In summary, also in hierarchical IP-layer networks, MT underperforms the other overlay topologies for the same reasons as in case of a flat network. As far as the remaining topologies are concerned, AC outperforms all the other topologies, due to the knowledge of the underlay topology which allows AC to reduce the node degree and, as a result, the overhead. PAC and DAC are two valuable alternatives to AC to further reduce the node degree, but a performance worsening occurs. More specifically, DAC performs better than PAC, highlighting the advantages of building traffic demand-aware topologies. Finally, the results of the simulations outline that KSPT is not a suitable topology for hierarchical IP networks, since it always performs worse than the other overlay topologies (except MT).

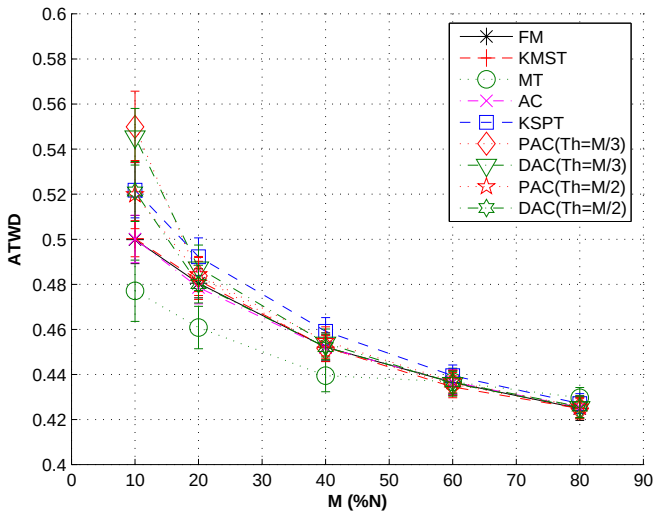


Figure 2.19: *ATWD for a hierarchical IP-layer network with N=600*

Case study: real network scenario

Figure 2.20 shows the AND trend in a real network scenario when M ranges from $0.01 \cdot N$ to $0.08 \cdot N$.

The graph outlines that the AND increases almost linearly with M for all the overlay topologies except for MT, which has the minimum AND that remains constant independently of M.

The AND of AC always have the maximum node degree. To be noted that PAC and DAC reduce the AND of AC. When the degree threshold is equal to $2 + M/3$, PAC and DAC have almost the same AND as KMST.

Regarding KSPT, it outperforms KMST due to the reason explained in the previous sections.

Figure 2.21 reports the BRR average value and confidence interval. As outlined in the graph, all the topologies show a similar decreasing behaviour. When $M > 0.01 \cdot N$, BRR is very low, because the overall traffic demand is low in this scenario. When $M = 0.01 \cdot N$ the bandwidth request for each node is high due to the way used for generating the traffic matrix (see Section 2.4.1) and since it overloads the links, BRR assumes a significant value.

Figure 2.22 shows the DRR behaviour. In this case, KMST and AC show a similar trend

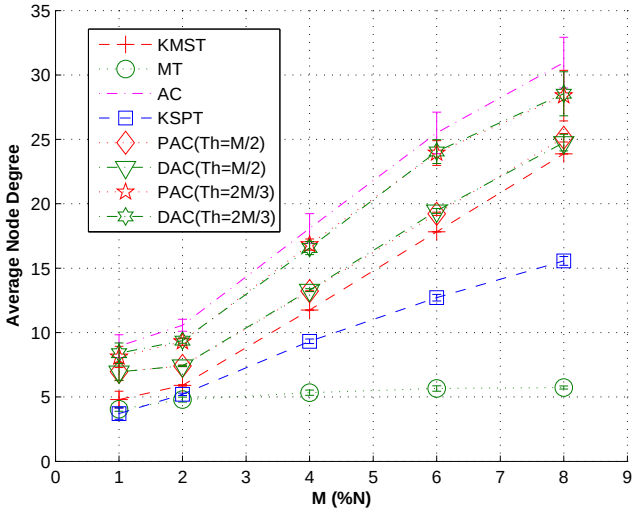


Figure 2.20: *AND for real network scenario with N=600*

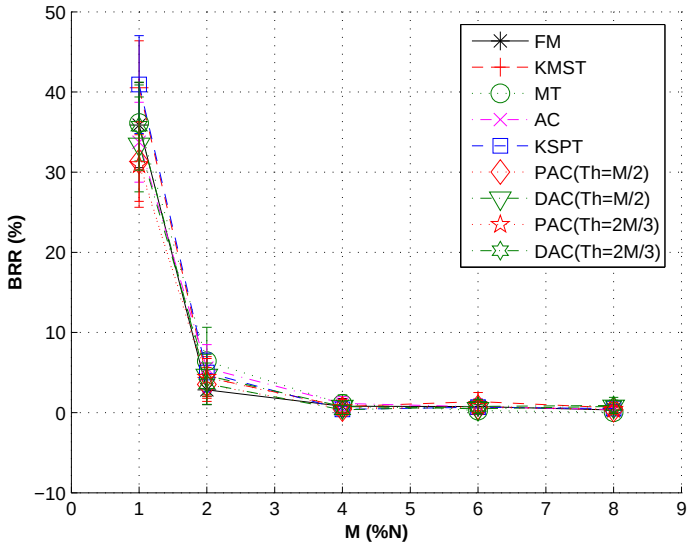


Figure 2.21: *BRR for real network scenario with N=600*

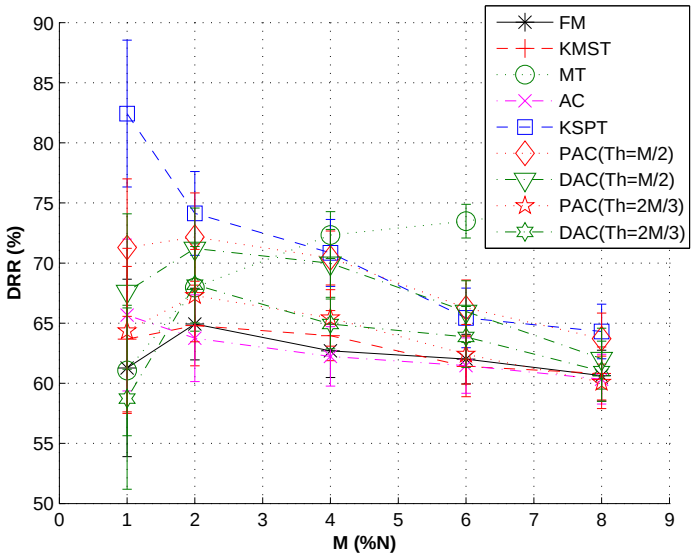


Figure 2.22: *DRR for real network scenario with N=600*

that decreases with M. Instead, MT has an increasing trend. To be noted that KSPT has the worst performance, while PAC and DAC have a worse DRR than AC, due to the introduction of the node degree threshold.

Figure 2.23 highlights that the ATWD behaviour, when the number of overlay nodes is very small ($M = 0.01 \cdot N$), depends on the overlay nodes position. All the overlay topologies have similar behaviour, only MT always has the minimum ATWD that remains constant independently of M.

To sum up, in a real network scenario MT does not underperform the other overlay topologies as in the previous cases, even though it has the lowest degree. KSPT topology has an high DRR. As far as the remaining topologies are concerned, KMST outperforms all the other topologies, due to the large amount of available bandwidth (for $M \geq 0.02 \cdot N$ $BRR \approx 0$) that allows it to exploit its ability of minimizing the global delay. Regarding the two novel topologies, PAC and DAC, they have a lower degree than AC, so they have similar performance to KMST.

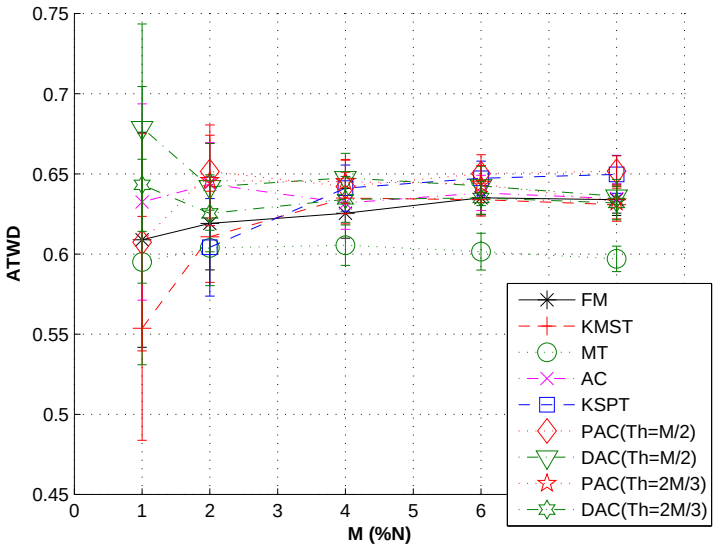


Figure 2.23: *ATWD for real network scenario with N=600*

2.5 Conclusions

In this chapter, the topology design problem of SONs with performance requirements has been addressed taking into account both the traffic demand and the overhead. This problem has been theoretically formalized, but since it is computationally too complex, the performance of some reference overlay topologies has been compared and three new topologies, called KSPT, PAC, and DAC have been introduced. Through extensive simulations, the performance and overhead of each overlay topology have been investigated either when the IP-layer network model is flat or hierarchical and in a real network scenario.

The results presented in this chapter show that MT always performs worse than the other overlay topologies. Moreover, when the size of the flat IP-layer network is large and the number of overlay nodes is small, KSPT is a valuable option.

Instead, when the IP-layer network is hierarchical, the AC topology outperforms all the other network topologies, because it takes advantage of the underlay topology knowledge. Moreover, PAC and particularly DAC are two possible alternatives to the AC topology to reduce the node degree. Finally, in a real network scenario with a large amount of available

bandwidth, KMST outperforms all the other overlay topologies and MT does not underperform the other overlay topologies as in the previous cases.

CHAPTER 3

NETWORK POWER MANAGEMENT

3.1 Background

Generally speaking, the Network Power Management consists in determining the design and the routing strategies that permit to minimize the overall power consumption of a network by taking into account the power consumption of the network elements and the traffic demand among the nodes. Therefore, two critical issues can be devised in NPM: the time horizon characterizing the traffic demands and the power behavior of the network devices.

Regarding the first issue, the time scale of the NPM approaches should be peak/off-peak traffic periods. In fact, usually there are no significant changes in the traffic matrix within each peak and each off-peak period. Therefore, the off-peak period (or peak period) can be exploited for estimating the traffic matrix in the next peak period (or off-peak period) and then computing the solution to optimize the network power consumption. The duration of the peak/off-peak period is in the order of day/night, the maximum acceptable time for computing a NPM solution can be reasonably assumed in order of 6/12 hours.

Regarding the second issue, the energy characterization of the network devices is important because it determines what can be exploited to minimize the overall power consumption of the network. In this thesis, a general power consumption model of a router has been taken into account, which is composed of three main components [37]:

- chassis;
- Physical Interface Cards (PICs)

- route processor.

The chassis can be powered off (i.e. it works at a low power mode); the corresponding power consumption can be assumed constant if the chassis is powered on, and zero otherwise.

The energy consumed for transferring a bit from a node u to a node v is due to diverse components, such as the power consumed by the PIC in node u used to transmit the bit and that used by the PIC of node v to receive it. In the rest of the thesis, when considering traffic sent from a certain node u to a certain node v, the corresponding power consumption and the related capacity value will be associated with the PIC at node u, without distinguishing the power contributions given by the transmitter and the receiver PICs. Obviously, in the reverse direction, they will be associated with the PIC at node v. Furthermore, since in most actual scenarios the network operators try to use similar devices in their core network, we shall assume that the power consumption and the capacity of the PIC used to transmit along (u,v) are equal to those of the PIC used for transmitting along (v,u). However, in the case of a link connecting two nodes with diverse hardware features, the power consumption and the capacity of the PICs used to transfer the traffic in the two directions could be different. Each PIC can be powered off. In particular, there is a constant, non zero, power consumption when the PIC is powered on, and a zero power consumption when the PIC is powered off.

As far as the power consumption of the route processor is concerned, it generally depends on the traffic load of the router in a nonlinear way (see [56]). In [45], the authors present several possible behaviors:

- *linear* - switch architectures like Batcher, Crossbar and Fully-Connected;
- *logarithmic* - equipments implementing hibernation techniques;
- *cubic* - equipments that use energy saving methods such as dynamic voltage and dynamic frequency scaling (DVS-DFS);
- *on-off* - almost the totality of current network equipments that consume always the maximum power in every load condition.

Finally, in modern core networks pairs of routers are typically connected, for each traffic direction, by multiple PICs that form one logical bundled link [23]; this technique is called link aggregation and is standardized by IEEE 802.1AX [38]. Link aggregation technique is widely diffused because it allows one to easily upgrade the link capacity by adding new PICs, and to reach link capacities bigger than that available by using the current fastest technology.

For example, a 40 Gb/s bundled link may comprise four OC-192 PICs with capacity $10Gb/s$ each. By taking into account the link aggregation technique, in this thesis the ability of each PIC of the bundled links to be powered off has been explored.

Based on the energetic and technological characterization of the network devices described above, four basic NPM problems, which will be deeply investigated hereafter, are listed as follows:

- *PAR* - Given traffic demands associated with the nodes, PAR consists in determining the traffic routing strategy that permits to minimize the overall power consumption of the network, by taking into account only the power consumption of the nodes concerning the route processor;
- *PAND* - Given traffic demands associated with the nodes, PAND consists in determining the traffic routing strategy that permits to minimize the overall power consumption of the network, by considering the possibility to power off entire links and/or nodes of the network;
- *PARND* - Given traffic demands associated with the nodes, PARND consists in determining the traffic routing strategy that permits to minimize the overall power consumption of the network, by jointly considering the possibility of powering off entire links and nodes of the network, and the power consumption of route processor;
- *PARND-BL* - Given traffic demands associated with the nodes, PARND-BL consists in determining the traffic routing strategy that permits to minimize the overall power consumption of the network, by even considering the possibility to power off single PICs of bundled links.

3.2 Literature review

Starting from the seminal work on the study of power efficient network devices presented by the authors of [33], recently many efforts have been devoted to define strategies aimed at reducing the power consumption of the whole network infrastructure, and not only the one of single or few components.

In this framework, a first set of works has been devoted to define models for the problem PAND, introduced in Section 3.1. In particular, the authors of [20] present first measurements

of power consumption of networking devices, and then evaluate the total network consumption of a certain network topology given the power footprint of each element. The authors propose a standard Capacitated Multicommodity Minimum Cost Flow (CMCF) formulation [29] for finding the minimum set of network elements to be powered on in order to guarantee the required service. However, the complexity of the proposed model grows very fast as the number of the devices increases. The model in [29] is thus very expensive to solve even for small networks. The authors of [21] study the strategies to concentrate network traffic on a minimal subset of network resources in order to power off network nodes and links while still guaranteeing full connectivity and maximum link utilization constraints. Furthermore, by taking into account strict constraints on the computational time, they provide efficient heuristics that permit to find an approximated solution in an acceptable computational time also in large network scenarios. A variation of this class of problems is presented by the authors of [27], which propose a simple heuristic to power off links when bundles of multiple physical cables are present.

A second set of works has been focused on the definition and the solution of PAR problem. The authors of [45] presented a study on the reduction of the overall power consumption that can be achieved when the energy profiles of the network devices are taken into account during routing and traffic-engineering operations. They consider coarse linear approximations of the power consumption of the network devices and, then use a linear programming solver (e.g CPLEX [5]). However, PAR is a nonlinear multicommodity flow problem, since the power consumption of energy aware network devices is generally a nonlinear function of their workload. Given this feature, the PAR problem should be modelled solved by means of a nonlinear programming solver such as Ipopt (Interior Point OPTimizer) [6], a software package for large-scale nonlinear optimization that implements an interior-point line-search filter method. This has been performed in this study, as reported in the next chapter.

A third set of works has been proposed aimed at jointly considering the power aware routing and the network design (PARND). In particular, the authors of [15] discuss the problem by considering different technology assumptions. However, differently from our work, they assume that nodes and even links have a linear energy behavior. Furthermore, this computational study takes into account only the GEANT topology composed of 23 nodes and 74 links. Hence, differently from this work, they do not consider the impact of the network topology on the performance of the discussed solution. The authors of [13] propose a heuristic to solve the PARND problem. Differently from this study, they consider only the link energy consumption, which is assumed to be linearly proportional to the traffic that flows through.

In this thesis, there are the preliminary ideas for modelling the nonlinear behavior of the routers based on the indications in [37]. In particular, the considered energy model is given by the sum of two constant addends associated with the power consumption of the chassis and the PICs, and one variable (nonlinear) element associated with the route processor.

On the other hand, none has proposed and solved a general NPM problem like PARND-BL, that permits to minimize the overall power consumption of a network by choosing a routing strategy that jointly exploits the power behavior of the route processors as well as the ability of powering off the chassis of the routers and all (or even single) PICs of bundled links.

The presented models have been evaluated by finding their optimal and/or approximated solutions and proposing new heuristic solutions.

An additional contribution of this thesis is to report the results of several wide computational experimentations, on several real network topologies, aimed at investigating the efficiency of the proposed power aware approaches from different perspectives.

3.3 Problem formulations

Let us introduce the parameters and the notation to define and formulate the power aware problems studied in this paper. The starting point of the analysis is a network modeled as a directed graph $G = (V,E)$, where V denotes the set of the nodes and E is the set of the arcs, modelling the network links.

The following parameters are assumed to be given in order to characterize the power consumption of the network elements:

- P_{uv}^{PIC} is the power consumption of a PIC that transmits traffic from node u to node v (i.e., a PIC of link (u,v));
- P_{v}^{C} is the power consumption concerning the chassis of the node v;
- $P_{v,T(v)}^{RP}$ is the power consumption concerning the route processing of the node v at the traffic throughput $T(v)$; as justified before, hereafter $P_{v,T(v)}^{RP}$ is assumed to be a nonlinear function.

Since each logical link is generally composed of a set of PICs, the overall power consumption for the traffic transmission on a link (u,v) is equal to the number of powered on PICs, in node u connected to v, multiplied by P_{uv}^{PIC}. In other words, the maximum power consumption

associated with a directional link (u,v) is given by $N_{uv} \cdot P_{uv}^{PIC}$, where N_{uv} is the maximum number of PICs forming the bundled link that connects node u to node v.

Concerning the traffic demand and the capacity of nodes and links, we define:

- D is the set of the origin-destination pairs of the traffic matrix;
- d_{sd} is the traffic demand between the source node, s, and the destination, d;
- C_v^N is the capacity of node v;
- C_{uv}^{PIC} is the capacity of each PIC which composes the link (u,v);
- C_{uv}^L is the capacity of the link (u,v) (in the case of bundled link $C_{uv}^L = C_{uv}^{PIC} \cdot N_{uv}$).

Three sets of variables are defined to state the NPM problems:

- f_{uv}^{sd} is the amount of d_{sd} flowing through the link (u,v);
- $x_v \in \{0,1\}$ is set to 1 when the node v is powered on, and to 0 otherwise;
- n_{uv} is the number of powered on PICs which compose the link (u,v).

The traffic throughput of node v can then be defined as the total traffic entering v plus the flow originated from v, according to the following formula:

$$T(v) = \sum_{(u,v)\in E} \sum_{(s,d)\in D} f_{uv}^{sd} + \sum_{(v,d)\in D} d_{vd}. \tag{3.1}$$

Now the NPM problems addressed in this study can be presented, together with related mathematical models. The first problem is a power aware routing problem that, by assuming that all the nodes and all the PICs of the network are powered on, aims to route the traffic demands in such a way as to minimize the overall power consumption of the network. The *PAR* problem leads to a nonlinear multicommodity flow model. The second problem, *PAND*, considers the possibility to power off nodes and links of the network in order to reduce the overall power consumption. This is done by approximating the power behavior of the nodes to the worst case, so leading to a Mixed Integer Linear Programming (MILP) design and routing model. Problem *PARND* generalizes both *PAR* and *PAND*. In fact it addresses both the possibility of powering off nodes and links as well as the routing strategy, by however considering the true, nonlinear power behavior at the nodes. The corresponding model is therefore a mixed integer nonlinear programming model. Finally the last problem, *PARND-BL*, represents a further level of generalization, since it allows that even single PICs of

bundled links can be powered off. In this way a hierarchy of NPM models is defined and studied, having *PARND-BL* on the top, models *PAR* and *PAND* on the bottom, and *PARND* at the intermediate level. To the best of our knowledge, this is the first time that a nonlinear Power Aware Routing and Design model is investigated from a computational perspective. Furthermore, model *PARND-BL* is innovative in our opinion, since the possibility of powering off single PICs of logical links was never investigated in the context of power aware routing and network design. Therefore, as outlined before, model *PARND-BL* has to be considered as an original contribution of this paper.

3.3.1 Power Aware Routing (PAR)

The PAR problem consists in determining the traffic routing strategy that permits to reduce the overall power consumption of the network by taking into account only the power consumption of the nodes concerning the route processor, i.e. $P^{RP}_{v,T(v)}$. In fact, this problem does not consider the possibility of powering off any network element. In particular, the assumption is made that each node is active, and each link (u,v) has the maximum number, N_{uv}, of active PICs. The PAR problem can be formulated according to the following nonlinear multicommodity flow model. Note that the variables x_v and n_{uv} are not present, since the corresponding decisions are not taken into account by the optimization problem under consideration.

$$\text{Minimize} \sum_{v \in N} P^{RP}_{v,T(v)} \tag{3.2}$$

subject to

$$\sum_{(v,u)\in E} f^{sd}_{vu} - \sum_{(u,v)\in E} f^{sd}_{uv} = \begin{cases} d_{sd} & \text{if } v = s \\ -d_{sd} & \text{if } v = d \\ 0 & \text{otherwise} \end{cases} \quad \forall (s,d) \in D \ \ \forall v \in V \tag{3.3}$$

$$T(v) \leq C^N_v \quad \forall v \in V \tag{3.4}$$

$$\sum_{(s,d)\in D} f^{sd}_{uv} \leq C^L_{uv} \quad \forall (u,v) \in E \tag{3.5}$$

$$f^{sd}_{uv} \in \mathbb{R} \geq 0 \quad \forall (u,v) \in E \ \ \forall (s,d) \in D \tag{3.6}$$

Equations (3.3) are the classical flow conservation constraints, instead Equations (3.4) and (3.5) are the node and the link capacity constraints, respectively. Finally, Equations (3.6) provide the flow variable definition.

3.3.2 Power Aware Network Design (PAND)

Differently than PAR, the PAND problem considers the possibility to power off links and nodes in order to reduce the overall power consumption of the network. This is achieved by taking into account the possibility to power off nodes as well as entire logical links, i.e., the assumption is made that either the PICs composing each link are all active, or they are all inactive. Concerning each powered on node v, the proposed model approximates its power behavior by considering the maximum power consumption at v, i.e. $P_v^{MAX} = P_v^C + P_{v,C_v^N}^{RP}$. The assumptions above lead to the following MILP model, where each design variable n_{uv} either holds 0 or is set to the maximum number of PICs composing (u,v), i.e. N_{uv}:

$$\text{Minimize} \sum_{v \in N} P_v^{MAX} \cdot x_v + \sum_{(u,v) \in E} P_{uv}^{PIC} \cdot n_{uv} \tag{3.7}$$

subject to the flow conservation constraints (3.3) and to constraints (3.6), plus

$$T(v) \leq C_v^N \cdot x_v \quad \forall v \in V \tag{3.8}$$

$$\sum_{(s,d) \in D} f_{uv}^{sd} \leq C_{uv}^{PIC} \cdot n_{uv} \quad \forall (u,v) \in E \tag{3.9}$$

$$\sum_{(u,v) \in E} n_{uv} \leq \sum_{(u,v) \in E} N_{uv} \cdot x_v \quad \forall v \in V. \tag{3.10}$$

$$x_v \in \{0,1\} \quad \forall v \in V \tag{3.11}$$

$$n_{uv} \in \{0, N_{uv}\} \quad \forall (u,v) \in E \tag{3.12}$$

Equations (3.8) and (3.9) are the node and the link capacity constraints extended to include the possibility to power off nodes and links, respectively. Equations (3.10) state that a node can be powered off only when all the incident links are powered off. Finally, Equations (3.11) define $x_v,\ \forall v \in V$, as binary variables, while Equations (3.12) impose that only entire links can be powered off.

3.3.3 Power Aware Routing and Network Design (PARND)

Problem PARND refines PAND by jointly considering the possibility of powering off links and nodes of the network, and determining a power aware routing strategy, under the more accurate power behavior at the nodes given by the nonlinear function $P_{v,T(v)}^{RP}$. The PARND

problem can therefore be formulated according to the following mixed integer nonlinear programming model:

$$\text{Minimize} \sum_{v \in N} P^{RP}_{v,T(v)} + \sum_{v \in N} P^{C}_{v} \cdot x_v + \sum_{(u,v) \in E} P^{PIC}_{uv} \cdot n_{uv} \tag{3.13}$$

subject to the same constraints of PAND (given by (3.3), (3.6) and (3.8) - (3.12)).

3.3.4 Power Aware Routing and Network Design with Bundled Links (PARND-BL)

The objective of the last problem, PARND-BL, is to minimize the overall power consumption of the network by even considering the possibility to power off single PICs of bundled links. The problem can then be formulated as follows:

$$\text{minimize} \sum_{v \in V} P^{RP}_{v,T(v)} + \sum_{v \in V} P^{C}_{v} \cdot x_v + \sum_{(u,v) \in E} P^{PIC}_{uv} \cdot n_{uv} \tag{3.14}$$

subject to most constraints of PAND (i.e. (3.3), (3.6) and (3.8) - (3.11)) and to

$$n_{uv} \in \mathbb{N}_0 \leq N_{uv} \quad \forall (u,v) \in E \tag{3.15}$$

In particular, constraints (3.15) guarantee that even single PICs of bundled links can be powered off.

3.4 Heuristic of the PAR problem

The Dijkstra-based Power-aware Routing Algorithm (DPRA) consists in the partitioning in small quantities, δ, of the traffic demand *sd*, and in the calculation of the minimum power consumption path for δ taking into account the resources already allocated in the network. This procedure is recursively executed for all couples of nodes, and until all the traffic demands reported in the traffic matrix D are allocated. At each iteration, the proposed heuristic associates at each oriented link a cost equal to the increase of the power consumption of the destination node. This parameter is calculated taking into account the δ, the traffic allocated on the considered link, and the power consumption of the route processor in the destination node. Then, the Dijkstra's algorithm is used to compute the minimum cost path. The pseudo-code of the proposed DPRA is shown in the Algorithm 1.

Algorithm 1 Dijkstra-based Power-aware Routing Algorithm

Given: $G(V,E)$ and D

1: Set δ_0, $T^{CI}(v) = \sum_{u \in V} d_{vu}\ \forall v \in V$, and $d_{sd}^{RES} = d_{sd}\ \forall (s,d) \in D$
2: **repeat**
3: Select randomly a couple sd such that $d_{sd}^{RES} > 0$
4: Set δ
5: Calculate cost $w_{uv}\ \forall (u,v) \in E$
6: Calculate $C_v^{MAX}\ \forall v \in V$ and $C_{uv}^{MAX}\ \forall (u,v) \in E$
7: Delete links and nodes that do not satisfy maximum utilizations
8: Run *Dijkstra's algorithm* between s and d with cost w_{uv}
9: Update $T^{CI}(v) += \delta\ \forall v \in P_{sd}^{CI}$ and $d_{sd}^{RES} -= \delta$
10: **until** $d_{sd}^{RES} == 0 \quad \forall (s,d) \in D$

After the initialization (step 1), DPRA begins the iterations and selects sd, then δ is set as shown in Algorithm 2. Note that the choice of parameter δ_0 is a trade off between accuracy and simulation time. At step 5, the cost of the link $uv\ \forall (u,v) \in E$ is calculated as follows:

$$w_{uv} = P^{RP}_{v,T^{CI}(v)+\delta} - P^{RP}_{v,T^{CI}(v)} \tag{3.16}$$

where $T^{CI}(v)$ is the traffic throughput of node v at the current iteration.

Algorithm 2 Set δ

if $d_{sd}^{RES} \geq \delta_0$ **then**
  $\delta = \delta_0$
else
  $\delta = d_{sd}^{RES}$
end if

Afterwards, the maximum resources available at each node, C_v^{MAX}, and at each link, C_{uv}^{MAX}, are calculated as explained in Algorithms 3 and 4. In particular, C_v^{MAX} is calculated taking into account the amount of traffic directed to the considered node v. In order to avoid that the algorithm could be blocked for lacking of resources at the receiving node, resources of node v must be reserved. Similarly, resources should be reserved in the link (u,v). To avoid the blocking of the algorithm for lacking of resources, three different cases should be considered (obviously when $(u,v) = (s,d)$, we do not need to reserve resources):

- when u is the source of the traffic demand, we should permit the reception of the traffic having v as destination; to this aim we allocate to each link attached to the node v an equal share of the traffic transmitted to it;

- when v is the destination of the traffic demand, we should permit the transmission of the traffic generated by u; also in this case, we allocate to each link attached to the node u an equal share of the total traffic generated in u;

- when (u,v) is an intermediate link of the traffic demand d_{sd}, we should allocate the resources needed for the reception of the other traffic having v as destination and those necessary for the transmission of the other traffic having u as source; in both cases, we assume that the whole traffic is uniformly distributed among the links attached to the node.

The next step consists in deleting (or equivalently in setting the cost to ∞) the nodes and the links that have not enough resources to participate to the next allocation process carried out by means of the Dijkstra's algorithm. In particular, defining $A^{CI}(uv)$ as the amount of traffic allocated on the link (u,v) until the current iteration, the link (u,v) is deleted from the graph iff $A^{CI}(uv)+\delta > C_{uv}^{MAX}$. Similarly, the node u is deleted iff $T^{CI}(u)+\delta > C_u^{MAX}$.

Algorithm 3 Calculate C_v^{MAX} $\forall v \in V$

if $v = d$ **then**
  $C_v^{MAX} = C_v^N$
else
  $C_v^{MAX} = C_v^N - \sum_{i \in V} d_{iv}^{RES}$
end if

At the step 8, Dijkstra's algorithm runs using the costs w_{uv}, then the $T^{CI}(v)$ and d_{sd}^{RES} values are consequently updated (P_{sd}^{CI} is the set of nodes belonging to the path from s to d). Then, after the update of the variables $T^{CI}(v)$ and d_{sd}^{RES}, the algorithm returns to step 3 until $d_{sd}^{RES} = 0 \quad \forall (s,d) \in D$.

The computational complexity of DPRA is about $\frac{|V|\cdot(|V|-1)}{2}\lceil\frac{\sum_{sd\in D} d_{sd}}{\delta_0}\rceil \times O(|E|+|V|\log|V|)$, where $|V|$ is the number of nodes, $|E|$ is the number of links, and $O(|E|+|V|\log|V|)$ is the computational complexity of the efficient implementation of the Dijkstra's algorithm [22].

Algorithm 4 Calculate C_{uv}^{MAX} $\forall uv \in E$

if $u = s$ and $v = d$ **then**
  $C_{uv}^{MAX} = C_{uv}^{L}$
else if $u = s$ and $v \neq d$ **then**
  $C_{uv}^{MAX} = C_{uv}^{L} - \frac{\sum_{i \in V} d_{iv}^{RES}}{deg(v)}$
else if $u \neq s$ and $v = d$ **then**
  $C_{uv}^{MAX} = C_{uv}^{L} - \frac{\sum_{i \in V} d_{ui}^{RES}}{deg(u)}$
else
  $C_{uv}^{MAX} = C_{uv}^{L} - \frac{\sum_{i \in V} d_{iv}^{RES}}{deg(v)} - \frac{\sum_{\substack{i \in V \\ i \neq v}} d_{ui}^{RES}}{deg(u)}$
end if

3.5 Heuristic of the PARND problem

For solving PARND a new heuristic (see Algorithm 5) is proposed, this heuristic is based on a PAND solution proposed in [21].

The heuristic begins by sorting the nodes by means of a least-flow (LF) policy, namely the nodes are sorted basing on their traffic throughput after solving the PAR problem on the network with all elements powered on (i.e., $x_v = 1$ $\forall v \in V$ and $y_{uv} = 1$ $\forall (u,v) \in E$). Afterwards, each node, and consequently all its adjacent links, is disabled according to the considered ordering and the PAR problem is solved on the restricted network. Note that only nodes without traffic demand towards/from any other node can be disabled. If the resolution successfully ends and the overall power consumption is reduced, then the network power consumption is updated and the iterations continue, otherwise the node is enabled again. When all nodes are considered, the PAR problem is solved for the new network topology where the previous disabled nodes and their adjacent links are powered off. Based on this solution, the links are sorted by taking into account the link flows (LF policy). Subsequently, a procedure similar to that used to power off the nodes is applied to the sorted list of the links. The solution of PAR applied to the resulting network topology finally provides the power-aware routing strategy and the overall power consumption.

It is relevant to observe that in the case the nodes cannot be powered off, the heuristic can be limited to the iterative procedure associated with the links.

Algorithm 5 Heuristic for the PARND problem

```
1: SN=sort nodes (V);
2: min_pow_cons = ∞;
3: for all v ∈ SN do
4:    disable node v;
5:    solve PAR problem ⇒ (status, power_consumption);
6:    if (status == Solve_Succeeded AND power_consumption < min_pow_cons) then
7:       min_pow_cons = power_consumption;
8:    else
9:       enable node v;
10:   end if
11: end for
12: SL=sort links (E);
13: min_pow_cons = ∞;
14: for all (u,v) ∈ SL do
15:   disable link (u,v);
16:   solve PAR problem ⇒ (status, power_consumption);
17:   if (status == Solve_Succeeded AND power_consumption < min_pow_cons) then
18:      min_pow_cons = power_consumption;
19:   else
20:      enable link (u,v);
21:   end if
22: end for
23: solve PAR problem.
```

3.6 Heuristics of the PARND-BNBL problem

In backbone networks, the powering off of a node should be a critical issue, this is mainly due to reliability and robustness reasons. Hence, a new version of PARND-BL, called Power Aware Routing and Network Design in Backbone Network with Bundled Links (PARND-BNBL), has been considered. The PARNB-BNBL problem consists in modifying the PARND-BL to avoid of powering off the chassis of the nodes as follows:

$$\text{minimize} \sum_{v \in V} P^{RP}_{v,T(v)} + \sum_{(u,v) \in E} P^{PIC}_{uv} \cdot n_{uv} \tag{3.17}$$

subject to the Equations (3.3), (3.4), (3.6), (3.9), and (3.15).

For solving PARND-BNBL a new heuristic (see Algorithm 6) is proposed, this heuristic is based on the following Fast Greedy Heuristic (FGH) approach.

Algorithm 6 PARND-BNBL Heuristic

```
1: Solve PAR ⇒ (status, Power_Consumption);
2: Remove cables to match flows and set FE = ∅;
3: Initialize Prev_cons = Power_Consumption;
4: repeat
5:    Sort edges (greatest spare capacity);
6:    repeat
7:       Select the first edge ∉ FE → (u, v);
8:       Disable one cable from selected edge (u, v);
9:       Solve PAR ⇒ (status, Power_Consumption);
10:      if (status == Solve_Succeeded AND Power_Consumption < Prev_cons) then
11:         Remove cable from uv;
12:         Update Prev_cons = Power_Consumption;
13:      else
14:         Enable cable and FE = FE ∪ {uv};
15:      end if
16:   until (status == Solve_Succeeded AND FE == E)
17: until (FE == E)
```

The FGH heuristic is based on the Maximum Spare Capacity (MSC) problem, which is used as a building block.

Taking into account the notations introduced in the previous section, the MSC problem can be formulated as follows:

$$\text{minimize} \sum_{(u,v)\in E} f_{uv} \tag{3.18}$$

subject to the Equations (3.3), (3.6), (3.4), and

$$f_{uv} = \sum_{(s,d)\in D} f_{uv}^{sd} \leq C_{uv}^{L} \quad \forall (u,v) \in E. \tag{3.19}$$

Obviously, Equation (3.19) ensures that no link carries more traffic flow than its capacity.

The FGH heuristic consists in solving the MSC problem to obtain the flow fuv assigned to each edge. Then, taking into account that all the flows are still satisfied, the maximal number of cables are removed. After these cable removals, the edge with the greatest spare capacity is identified, i.e., we find the (u,v) for which:

$$\arg\max_{(u,v)}(C_{uv}^{PIC} \cdot n_{uv} - f_{uv}) \tag{3.20}$$

where n_{uv} denotes the number of remaining cables after the cable removals. In order to reduce the excess traffic to be rerouted only one cable is considered for the removal. Assuming the considered removal, the MSC is solved with the new link capacities. If the problem has a feasible solution, the cable considered for removal is permanently removed, otherwise it is not removed and the corresponding edge is marked as final, i.e. no additional cables are removed from final edges (FEs). Then in the next iterations, the identification of the edge with the greatest spare capacity is performed ignoring all final edges. The iterative procedure concludes when all edges become final.

The choice of a heuristic based on FGH is due to the attractiveness of FGH, given by its simplicity. The main differences with respect to FGH are that at each iteration a cable is removed if the overall power consumption is actually reduced and the PAR problem is used as a building block in place of MSC. These differences are particularly relevant when the power consumption due to route processing is dependent on the node throughput. In this case, the power consumption due to route processing has to be explicitly addressed in the problem formulation, and just to power off links/PICs does not necessarily determine a reduction of the overall network consumption.

The general structure of the heuristic framework is depicted by Algorithm 6. In the following, two different versions of this framework are presented, each one based on a different solution of the PAR problem.

3.6.1 Time Limited PAR Heuristic (TLPH)

In the first version of the heuristic, the PAR has been solved by means of IpOpt [6], a software package for large-scale nonlinear optimization that implements an interior-point line-search filter method, considering the actual power consumption of nodes and finding the optimal solution.

It has been observed that this approach is time expensive because, when there is not a solution to the PAR problem, the IpOpt requires a lot of time to establish it. Therefore, to

speed up the proposed heuristic, the running time of IpOpt for solving PAR problem has been bounded as following:

- At step 1 of the Algorithm 6, calculate the running time for solving PAR $\rightarrow T$;
- At step 9, bound the running time for solving PAR at $(T+1s)*\gamma$, where γ is a multiplicative factor.

The drawback of this method is that the time can expire even if PAR can be solved ("false negative" event). Hence, a suitable value of the multiplicative factor γ should be chosen to limit both time and false negative events.

3.6.2 PAR Meta Heuristic (PMH)

Algorithm 7 HPAR

1: Initialize $T^I(v)\ \forall v \in N$, $f^I_{uv}\ \forall (u,v) \in E$, and d^I_{sd};
2: Calculate maximum demand $\Rightarrow (s^M, d^M)$;
3: **repeat**
4: Set cost $w(v) = P^{RP}_{v,T^I(v)+d^I_{s^M d^M}} - P^{RP}_{v,T^I(v)}\ \forall v \in V$;
5: Solve $MCCR(w, T^I(v), f^I_{uv})$ from s^M to $d^M \Rightarrow (P, status)$;
6: **if** $(status != Solve\_Succeeded)$ **then**
7: break;
8: **end if**
9: Update $T^I(v)\ \forall v \in P$, $f^I_{uv}\ \forall (u,v) \in P$ and $d^I_{s^M d^M}$;
10: Calculate maximum demand $\Rightarrow (s^M, d^M)$;
11: **until** $(d^I_{s^M d^M} = 0)$

In order to make the heuristic approach faster and faster, a new simple strategy to solve the PAR problem, summarized in Algorithm 7, is studied. The algorithm is very simple and consists in considering one origin-destination $(s,d) \in D$ at the time as follows. After the initialization of the node loads $T^I(v) = 0\ \forall v \in N$, the flows $f^I_{uv} = 0\ \forall (u,v) \in E$, and the traffic demands $d^I_{sd} = d_{sd}\ \forall (s,d) \in D$, the origin-destination pair having the maximum demand, i.e. $(s^M, d^M) | d^I_{s^M d^M} \leq d^I_{sd}\ \forall (s,d) \in D$, is calculated. Then, a suitable cost $w(v)\ \forall v \in N$ is set and a path from s^M to d^M, P, is computed by using a Minimum-cost Capacity-constrained Routing (MCCR), which finds (if it exists) the minimal cost $(w(v))$-path between two nodes

that satisfies the link and node constraints. If the MCCR finds a feasible solution, the values of the node loads $T^I(v) += d^I_{s^M d^M}\ \forall v \in P$, the flows $f^I_{uv} += d_{s^M d^M}\ \forall (u,v) \in P$, the traffic demand $d^I_{s^M d^M} = 0$ are updated, otherwise the algorithm ends. Successively, the new maximum traffic demand is computed, if it is equals to 0 the algorithm begins a new iteration, otherwise the algorithm ends.

3.7 Heuristic of the PARND-BL

For solving PARND-BL, a new heuristic, called *HPARND-BL*, is proposed, which is an evolution of the heuristic presented in the previous section.

The only difference with respect to the previous heuristic (Algorithms 6) is the addition of a part to power off nodes (see Algorithm 8 from step 4 to step 16).

The added part is similar to the following part for powering off the cables. At each iteration, the *transit node* (i.e. a node $v \in V$ is a transit node if $\sum_{(s,v)\in D} d_{sv} + \sum_{(v,d)\in D} d_{vd} = 0$) with the greatest spare capacity is removed, it is marked as final vertex (FV) and the PAR is solved by considering the new topology. If the problem has a feasible solution and the overall power consumption is actually reduced, the selected node is permanently removed, otherwise it is not removed. In the next iterations, the selected node is the one not included in the final vertexes with the greatest spare capacity. The iterative procedure ends when all nodes become final vertexes.

Algorithm 8 HPARND-BL

1: Solve PAR $\Rightarrow$ $(status, Power\_Consumption)$;
2: Remove cables to match flows and set $FE = \varnothing$ and $FV = \varnothing$;
3: Initialize $Prev\_cons = Power\_Consumption$;
4: **repeat**
5: Sort vertexes (greatest spare capacity);
6: **repeat**
7: Disable the first vertex $\notin FV \rightarrow v$;
8: Set $FV = FV \cup v$;
9: Solve PAR $\Rightarrow$ $(status, Power\_Consumption)$;
10: **if** $(status == Solve\_Succeeded$ AND $Power\_Consumption < Prev\_cons)$ **then**
11: Remove vertex v;
12: Update $Prev\_cons = Power\_Consumption$;
13: **else**
14: Enable vertex v;
15: **end if**
16: **until** $(status == Solve\_Succeeded$ AND $FV == V)$
17: **until** $(FV == V)$
18: **repeat**
19: Sort edges (greatest spare capacity);
20: **repeat**
21: Select the first edge $\notin FE \rightarrow uv$;
22: Disable one cable from selected edge uv;
23: Solve PAR $\Rightarrow$ $(status, Power\_Consumption)$;
24: **if** $(status == Solve\_Succeeded$ AND $Power\_Consumption < Prev\_cons)$ **then**
25: Remove cable from uv;
26: Update $Prev\_cons = Power\_Consumption$;
27: **else**
28: Enable cable and $FE = FE \cup uv$;
29: **end if**
30: **until** $(status == Solve\_Succeeded$ AND $FE == E)$
31: **until** $(FE == E)$

CHAPTER 4

PERFORMANCE ANALYSIS OF NPM

4.1 Evaluation of PAR problem and its heuristic

In this section, an extensive and accurate analysis of the PAR problem and the DPRA heuristic introduced in previous chapter is presented. The PAR problem has been solved by using a MILP solver and three different linear approximation of the function that represents the power behaviors of route processors. The analysis is focused on the evaluation of the potential power savings offered by PAR and its heuristic in different load and topology conditions. Note that in this section no QoS features have been taken into account, i.e. $\rho = 1$.

4.1.1 Simulations settings

The considered scenario is a European core network topology obtained from the Simple Network Description Library [8]; in particular, the file *nobel-eu* is considered. The network, shown in Figure 4.1, is composed by 28 nodes and 41 links.

Each node represents a core router; the use of the Juniper T1600 core router is assumed, which has a total throughput capacity of 1600Gb/s and a maximum power consumption of 8352W [3]; thus all nodes of the networks have the same power consumption of the route processor (i.e., P_v^{RP}). Consequently, a node capacity $C_v^N = 1600Gb/s \ \forall v \in V$ and a link capacity $C_{uv}^L = 600Gb/s \ \forall (u,v) \in E$ are assumed. Concerning the origin-destination demands, the traffic matrix has been obtained from the data file "Nobel-2 directed graph" downloaded from [8]; the file contains the measured traffic for each couple of nodes *sd* of the considered network scenario (we assume that the reported values are Gb/s). The amount

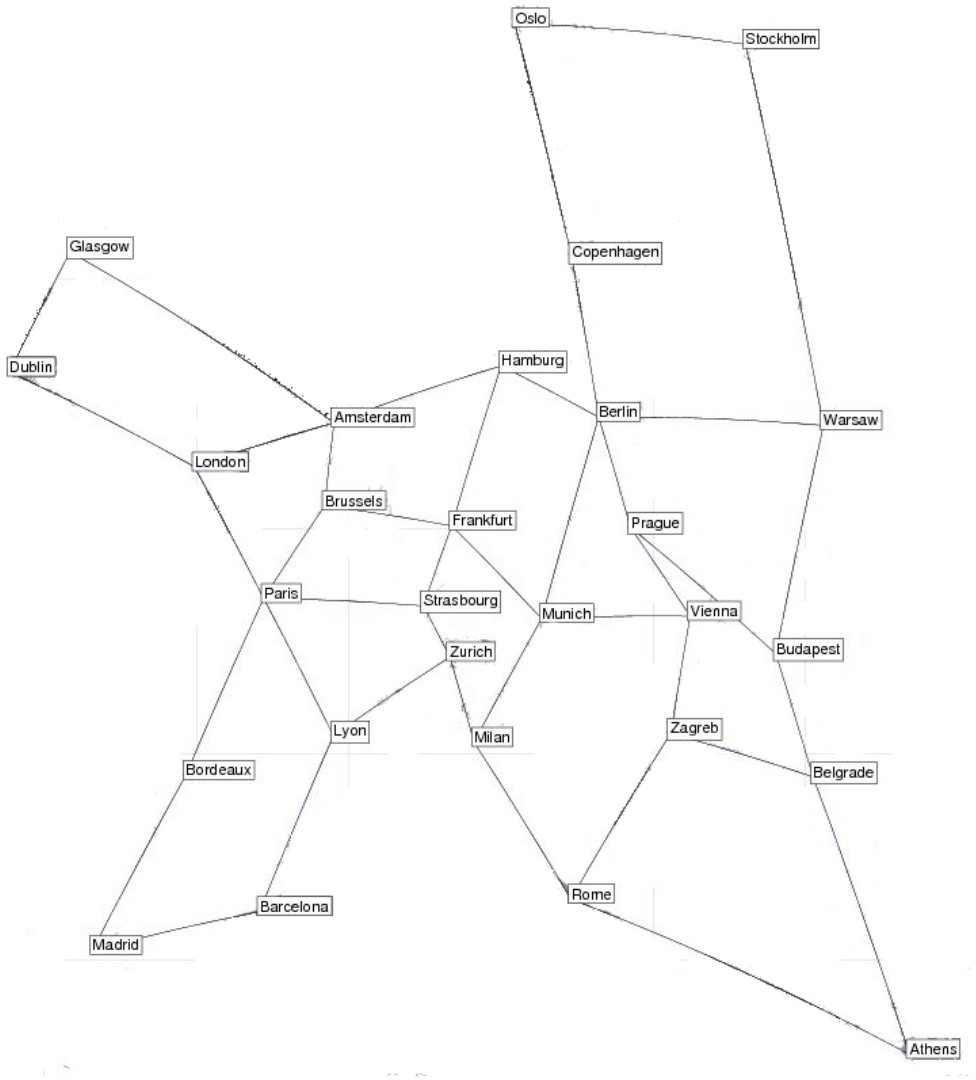


Figure 4.1: *European core topology considered in the simulation study* - nobel-eu

of traffic demand is of 1898 Gb/s, distributed among 378 active pairs (i.e. couples of nodes, sd with $d_{sd} > 0$). The mean traffic demand of an active pair is of about 5 Gb/s; then the parameter δ of the DPRA is set to the 2% of the mean traffic demand, i.e. to 0.1 Gb/s. This value of δ permits to achieve a good trade-off between performance and computation time.

Concerning the power consumption component of the route processing for the energy characterization of the devices, the attention is foccused on a *cubic curve*, since it represents the state-of-the-art of circuit-level energy-efficient mechanisms [45]. In particular, the cubic curve gives the energy behavior of network equipments that use energy savings techniques such as Dynamic Voltage and Dynamic Frequency Scaling (DVS-DFS), which permit energy consumption to scale with resource requirements. Current routers do not implement such techniques and are very energy inefficient, but they could be implemented in the next generation routers.

In order to solve the PAR problem, three different linear approximations of the cubic curve are considered. The first one approximates the cubic curve with 20 segments; this will be denoted as *20seg*. The second approximation considers four segments and will be denoted as *4seg*. The edges of the segments has been chosen taking into account the values of the ECR Initiative™ [4], which requires to measure the power consumption at 0%, 10%, 30%, 50%, and 100% of the total throughput capacity. Finally, the approximation based on only two segments (denoted as *2seg*) is also considered, which has been used in the simulation analysis of PAR discussed in [45]. The used approximations of the cubic curve are depicted in Figure 4.2.

The performance parameter considered in the comparison of the PAR and DPRA is the power savings of these algorithms with respect to the Shortest Path Routing (SPR). The power savings are defined as follows:

$$Power\ Savings_A = \frac{Power_{SPR} - Power_A}{Power_{SPR}} \times 100, \tag{4.1}$$

where the subscript A indicates the algorithm considered in the analysis (i.e PAR or DPRA). In this analysis, the SPR is calculated by means of CPLEX, which is set to find the minimum hop paths between two nodes taking into account the constraints on the link and node capacity.

The study is carried out considering two different aspects of the network: the load and the topology. In particular, in order to evaluate the impact of the load on the algorithm performance, we define the following parameter, denoted as Traffic Load (TL),

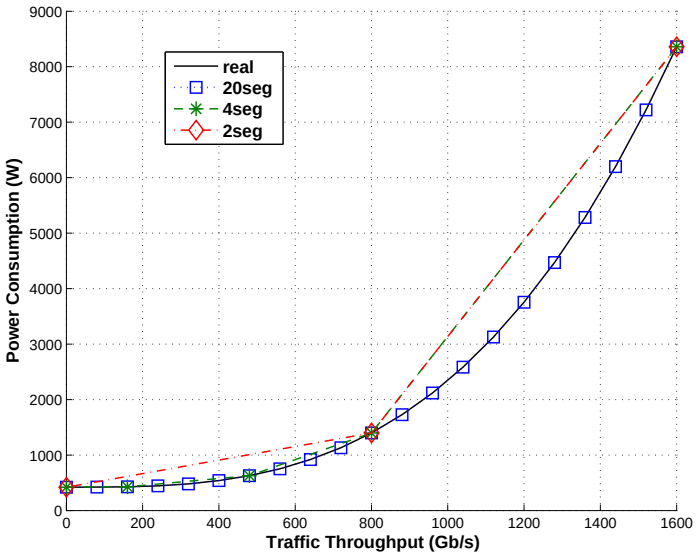


Figure 4.2: *Cubic curve and its linear approximations*

$$TL = \frac{1}{N} \cdot \sum_{u \in V} \frac{\sum_{v \in V} d_{uv}}{C_u^N}. \tag{4.2}$$

The *TL* represents the average fraction of the maximum capacity that a node should reserve for transmitting the locally generated traffic. Obviously, when the traffic load of a node is equal to 1, the node has not resources for forwarding or receiving traffic produced by others. The traffic load of the reference traffic matrix is 0.084. In order to vary the traffic load, the reference traffic matrix downloaded from [8] has been multiplied by diverse values.

As concerns the impact of the network topology on the algorithms performance, two indexes of the graph theory are considered: the average node degree and the node degree distribution. Starting from the considered topology, which has an average degree of 2.90, three new topologies are generated by applying one of the modifications reported in the following:

- *rand-add*: to randomly add links to the original topology to increase the average degree of one;
- *rand-add-2*: as *rand-add*, but with an increase of the average degree of two;

- *const*: to add and remove links in order to obtain a topology, where each node has a degree equal to 3.

The procedure used to randomly add links consists in a first step of choosing randomly with uniform distribution a node of the network. Then, considering all neighbors not connected with the chosen node, a link is added towards a neighbor chosen randomly with a probability inversely proportional to the geographic distance.

4.1.2 Simulation results

The discussion of the simulation results is organized in two subsections, depending on the considered aspect of the network features, i.e. traffic load and topology.

Traffic Load

The power savings (in %) obtained considering the different approximations of the power consumption of the route processors and for diverse traffic load values are summarized in Figure 4.3. A first observation regards the impact on the algorithm performance of the number of segments used to approximate the cubic curve. In particular, when the number of segments is reduced, the power savings are lost for low values of traffic load. This conclusion is supported by the comparison of the Figure 4.3(a), where about the 10% of the power consumption is saved with the PAR and a traffic load of 0.1, with the Figure 4.3(c), where no power saving is observed until a traffic load of 0.1. Until TL values lead all the nodes to work in the first segment of the approximate cubic curve, the PAR solutions are the same as the SPR. This statement is supported by the results shown in [45], in the case of a network having the same linear power behavior; the simulation results demonstrated that in this condition, the minimization of the overall power consumption of the network leads to the same results of the shortest path routing. When the TL value leads to have network nodes working in diverse parts of the linear approximation of the cubic curve (hence, in some cases characterized by different slopes), the minimization process of the PAR induces a diverse distribution of the traffic with respect to the SPR, resulting into power savings. This observation explains the increase of the power savings with the TL and the absence of power savings for low values of TL. In particular, the length of the first segment of the approximating curve is directly correlated to the value of the TL where the power savings begin. This remark is supported by the Figure 4.3(b), where the power savings of the PAR curve are higher than zero already after

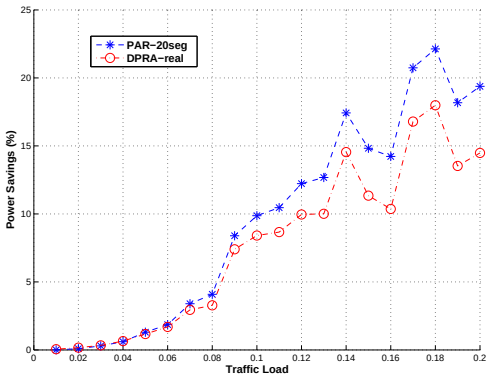
Table 4.1: *Comparison of the computation times (in s) and the* 99% *CI*

	TL=0.1	TL=0.2
PAR *(2seg)*	2.03 ± 0.01	3.41 ± 0.02
PAR *(20seg)*	4.78 ± 0.03	6.73 ± 0.04
DPRA *(real cubic curve)*	1.34 ± 0.04	2.71 ± 0.06

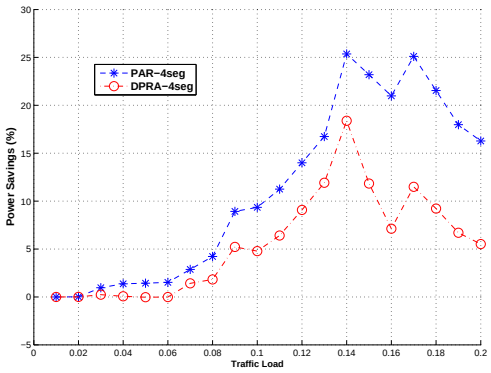
the third point. Indeed, in this case the length of the first segment of the approximating curve is less than $1/4$ of the *2seg* case, as shown in Figure 4.2. As concerns the DPRA performance, the coarse approximation of the cubic curve leads to the worsening of the performance with respect to the PAR. In particular, in the *2seg* case, Figure 4.3(c) shows the lack of power savings when the DPRA is used. On the contrary, the performance of the DPRA and the PAR are very close when we improve the approximation of the cubic curve. It is relevant to note that, in the DPRA, the cubic curve is considered in the calculation of the link costs w_{ij}; hence the utilization of the actual cubic curve, i.e. without linear approximation, is not a problem and does not increase the complexity of the algorithm. On the contrary, the linear approximation of the cubic curve is mandatory in the case of the PAR; furthermore, the complexity of the PAR algorithm increases with the number of segments used in the linear approximation. Table 4.1 reports the mean values and the 99% Confidence Interval (CI) of the time (in s) necessary for the two compared algorithms to produce the results, for two diverse TL values (the most significatives have been chosen, i.e. near the conditions of the actual traffic matrix and one of the points where the power savings are high). The CIs are calculated taking into account the results of 100 different runs. The values reported in the table highlight that the DPRA running with the actual cubic curve is about 3 times faster than the PAR solved with the *20seg* approximation. Furthermore, the DPRA is faster than the PAR also when the *2seg* approximation is considered.

Impact of Network Topology

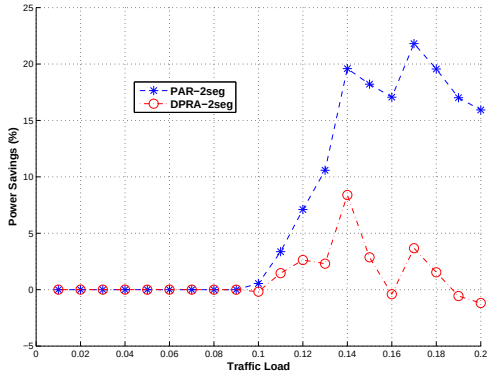
For this analysis, the energy savings are defined as $Energy\ Savings_A = Energy_{SPR} - Energy_A$, where $Energy_{SPR}$ is the energy consumed by the overall network when the SPR is used, and $Energy_A$ the energy consumed when the analyzed algorithm A, i.e. PAR or DPRA, is considered. The results of the energy savings obtained with the diverse network topologies and for different traffic loads are reported in Figures 4.4 (PAR method with the *20seg* approximation



(a) Real (DPRA) and *20seg* approx. (PAR) of cubic curve



(b) *4seg* approximation of cubic curve



(c) *2seg* approximation of cubic curve

Figure 4.3: *Power savings as a function of the TL and the approximation of cubic curve*

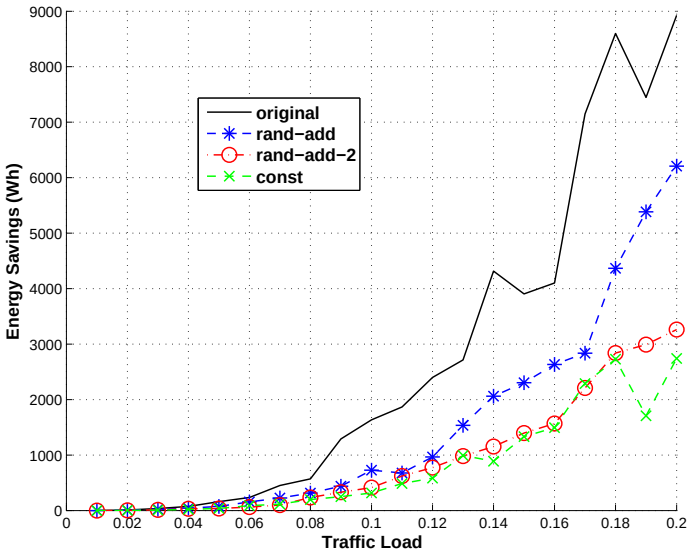


Figure 4.4: *Energy savings and network topology - PAR Case*

of the cubic curve) and 4.5 (DPRA algorithm with the real cubic curve). In particular, the comparison of the *rand-add* with the *rand-add-2* in the Figure 4.4 shows the reduction of the energy savings when the mean network degree increases. This result is due to the fact that when the mean network degree increases, the probability of finding alternative routes, more energetic efficient than the SPR path, decreases. As an example, in the boundary case of a complete meshed network, the link directly connecting the source and the destination is the shortest path, but also the only path with the lowest energy consumption; this path consumes only the energy at the transmitter/source node and at the receiver/destination node.

Furthermore, the energy savings decrease when all the nodes of the network have the same degree. The energy savings obtained with the *const* network topology are comparable with those achieved with the *rand-add-2*, although this last scenario has an higher network degree than the *const* (which has a mean network degree similar to the original network). In order to explain this behavior, the load of each network node has been analyzed in the four topologies after the application of the simple SPR, in the case $TL = 0.2$. In particular, although the average load is almost equivalent (i.e. about 600 Gb/s), a difference in terms of standard

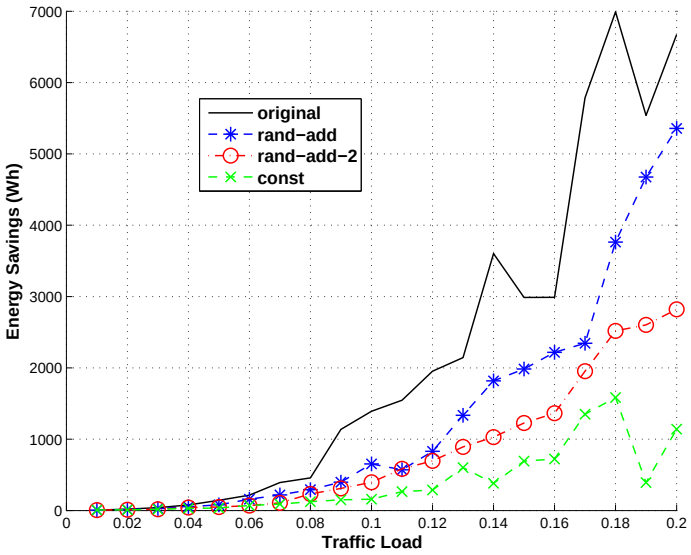


Figure 4.5: *Energy savings and network topology - DPRA Case*

Table 4.2: *Standard deviation of load of each network node after the SPR (Gb/s)*

original	rand-add	rand-add-2	const
433.15	298.36	270.28	241.93

deviation is observed (see Table 4.2). The comparison of the values of the table and the energy savings curves of Figure 4.4 highlights that the higher is the standard deviation of the load of a node after the application of the SPR the higher are the energy savings.

The results obtained with the DPRA algorithm, shown in Figure 4.5, lead to similar conclusions, although in this case the differences of the energy savings obtained with the diverse network topologies are less apparent.

4.2 Evaluation of PARND problem and its heuristic

The simulation study presented in this section is devoted to compare the performance in terms of network power consumption provided by different strategies presented in the previous chapter (excluding PARND-BL) . In particular, the results indicated as SPR, PAR, PAND and PARND refer to the optimal solution of the shortest path routing, the PAR, the PAND and the PARND problems. On the contrary, HPAND and HPARND refer to the results obtained by using the C++ code implementing the LF heuristic of [21] for the PAND problem, and the proposed heuristic for the PARND problem. In particular, the code implementing HPARND exploits the Ipopt for solving the PAR problems during the iterations of Algorithm 5. For analyzing different network load conditions, the traffic matrix has been multiplied by a scalar β. Also in this section no QoS features have been taken into account in all the approaches, i.e. $\rho = 1$.

It is relevant to note that the overall power consumed by the network is calculated by summing up the $P_{uv}^{PIC} \cdot N_{uv}$ and the P_v^C of the active links and nodes, as well as the $P_{v,T(v)}^{RP}$ calculated for each active node by taking into account its throughput $T(v)$.

4.2.1 Simulations settings

In the following, two core network scenarios are presented to analyze the performance of the different approaches. In both network scenarios, each node represents a core router. As in evaluation presented in the previous section, for each node the use of the Juniper T1600 core router is assumed, thus all nodes of the networks have the same energy behavior. For each link the use of multiple SONET/SDH OC768c/STM256 PICs is assumed, these PICs have a payload bandwidth of $38.486 Gb/s$ and a power consumption of 65.7W [9]. Thus, all links have the same capacity. To determine the number of PICs for each link, we have computed the maximum load of a link using the SPR, aimed at minimizing the number of hops between the source and the destination of each traffic demand. Thus, the link capacity is the minimum multiple, N_{uv}, of the PIC capacity higher than or equal to the maximum load.

The results reported in [20] show that the power consumption of a chassis is equal to about 200W for all classes of routers; thus, for each node v we have set $P_v^C = 200W$.

In summary, the following parameters are considered:

- $P_v^{MAX} = 8352W$ and $C_v^N = 1600 Gb/s \; \forall v \in V$;
- $P_{uv}^{PIC} = 65.7W$ and $C_{uv}^{PIC} = 38.486 Gb/s \; \forall uv \in E$;

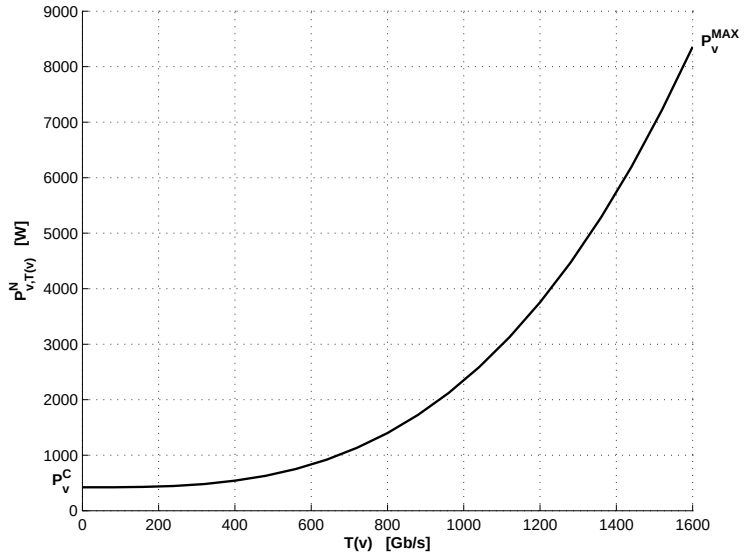


Figure 4.6: *Power consumption of node v,* $P^N_{v,T(v)} = P^C_v + P^{RP}_{v,T(v)}$

- $P^C_v = 200W \ \forall v \in V$;

where P^{MAX}_v is the maximum power consumption of node v.

Concerning the component of the route processor for the energy characterization of devices, as in the previous evaluation, a *cubic curve* (see Figure 4.6) is considered.

Based on the previous cited parameters of actual routers, the power consumption concerning the route processor can be computed as follows.

$$P^{RP}_{v,T(v)} = \frac{P^{MAX}_v - P^C_v}{(C^N_v)^3} \cdot T(v)^3 \ \forall v \in V. \tag{4.3}$$

Network Scenario "nobel-eu"

The first considered topology is the nobel-eu network, which has been already presented in the previous section.

As in the previous evaluation, the traffic matrix is obtained by the data file "Nobel-2 directed graph" downloaded from [8]. But, since the file contains the undirected traffic demand

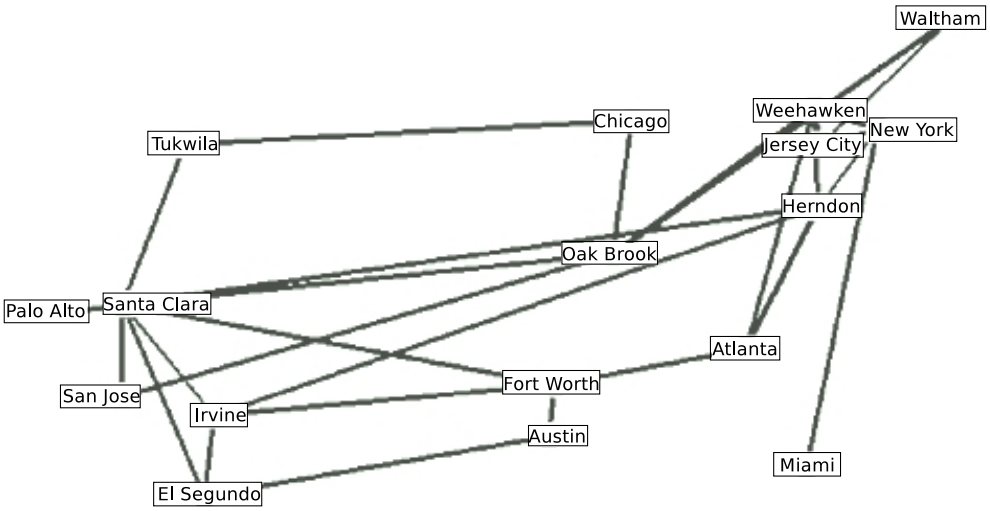


Figure 4.7: *Network topology of Exodus (US)*

between each couple of nodes, the demand has been randomly split between each direction to obtain the directed traffic demands.

The total amount of traffic demand is 1898 Gb/s, distributed among the 756 pairs (i.e. all couples of nodes $(s,d) \in D$). The mean traffic demand of an active pair is of about 2.5 Gb/s.

Considering this traffic demand, the link capacity has been computed as previously described and a link capacity equals to 307.888Gb/s (i.e. $N_{uv} = 8 \ (u,v) \in E$) and a power consumption of $525.6W \ \forall (u,v) \in E$ have been obtained.

Network Scenario "Exodus"

The other considered network scenario is the Exodus (US) network topology (see Figure 4.7), which is obtained from the dataset AS 3967 in the Rocketfuel study [7]. The network topology is obtained from the original data set by merging in one router all routers located at the same city and in one link all parallel links.

To obtain the traffic matrix, the mean value of traffic demand from s to d is set as in the nobel-eu scenario, i.e. $\overline{d_{sd}} = 2.5Gb/s$. Then, each element of the matrix d_{sd} is extracted from a uniform distribution: $d_{sd} = U\left[0.5 \cdot \overline{d_{sd}},\ 1.5 \cdot \overline{d_{sd}}\right] \ \forall s,d \in V$.

Taking into account this traffic demand, after applying the SPR and calculating the max-

imum load link, the link capacity is 115.458Gb/s (i.e. $N_{uv} = 3\ \forall(u,v) \in E$) and the power consumption is of $197.1W\ \forall(u,v) \in E$.

4.2.2 Simulation results

Results for the nobel-eu scenario

The first parameter considered in the comparison is the overall power consumed by the network for transporting the traffic demands. The results are summarized in Figure 4.8 for three values of β and for the SPR, the PAR, the HPAND and the HPARND strategies. The results concerning the PAND and the PARND have not been added to the Figure, due to the difficulties in finding the optimal solution for these problems. In particular, CPLEX (Optimization Studio Accademic Research Edition 12.2) did not provide a solution of the PAND problem after 72 hours, when running on a PC with 12 Intel Xeon L5440 @2.27GHz CPUs, 24GB RAM, and an Intel S5520HC Motherboard. Furthermore, it is relevant to note that, for $\beta = 1.1$, SPR and HPAND (using SPR in its iterations) did not find a solution satisfying the constraints on the link capacity. Indeed, in both cases the link transporting the traffic from Hamburg to Berlin has a load higher than the maximum capacity, fixed to $307.888 Gb/s$. On the contrary, HPARND permits to satisfy the constraints on the links capacity also for $\beta = 1.1$. The performance of HPAND and HPARND are quite similar for $\beta = 0.9$ and $\beta = 1.0$; in both cases the two heuristics lead to the same number of switched on links, equal to 56 and 58, respectively. Hence the small differences between the values of the power consumed using these heuristics is due to the routing strategy. Exploiting a power-aware routing strategy, HPARND permits to save further energy with respect to HPAND. Furthermore, for $\beta = 1.1$ HPARND permits to satisfy all traffic demands and simultaneously to switch off 24 unidirectional links. In this condition, the maximum link load is equal to about 253 Gb/s, which represents about the 82% of the link capacity.

Figure 4.9 shows the load of the links obtained with SPR, HPAND and HPARND. The comparison of the curves highlights that HPARND leads to transfer traffic demands from links that would be underutilized with the SPR strategy towards other selected links. This optimization of the routes leads to the increase of the average load of the links, but permits also to switching off some links that in the case of the SPR strategy would be lightly loaded.

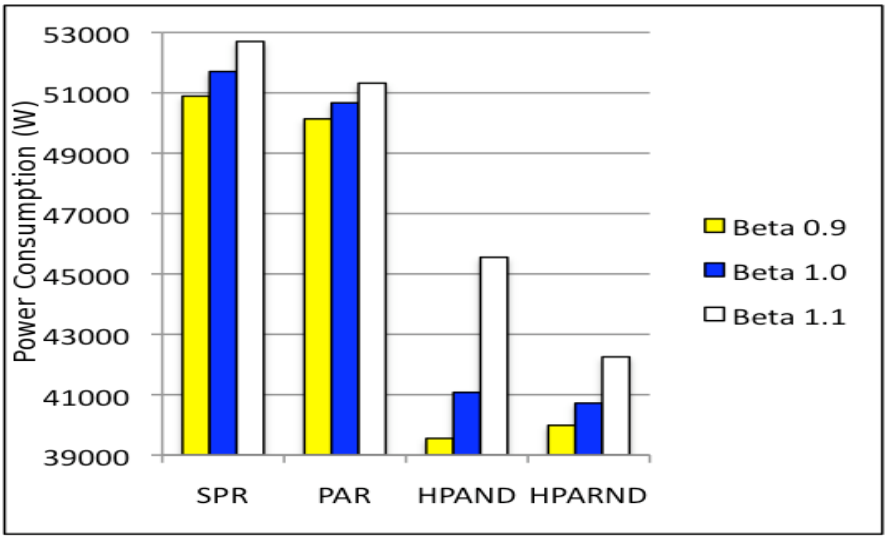


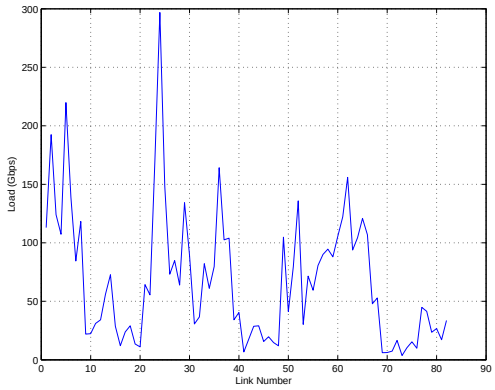
Figure 4.8: *Power consumption vs. the compared strategies - nobel-eu Scenario*

Results for the Exodus scenario

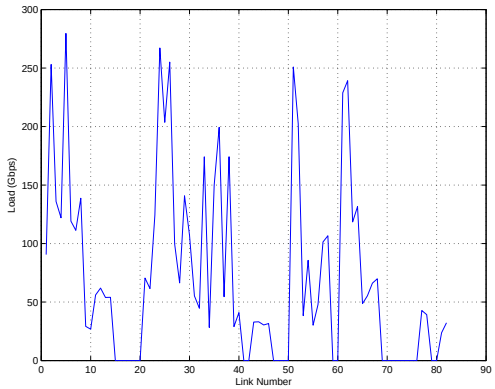
In this small network scenario, PAND and PARND can be also solved via CPLEX. The results in terms of the power consumption are summarized in Figure 4.10 for three values of β. Before the analysis of the results, it has be noted that the simple SPR algorithm does not allow to satisfy the constraints on the maximum link load for $\beta = 1.1$. In this scenario, the load of the links transporting the traffic from Herndon to New York and in the opposite direction, is higher than the link capacity.

Analyzing the Figure 4.10, it shows that the proposed HPARND permits to achieve results near the optimum, calculated by solving PARND with CPLEX. Furthermore, the distance between the overall power consumption obtained with HPARND and the optimum one increases as the traffic demands increase; indeed, whereas for $\beta = 0.9$ HPARND and PARND produce similar values of power consumption, for $\beta = 1.1$ HPARND gives a power consumption that is about 2% higher than the optimum one. Moreover, for the same value of β HPAND leads to an overall power consumption that is about 13% higher than the optimum one, calculated by solving the PAND problem with CPLEX.

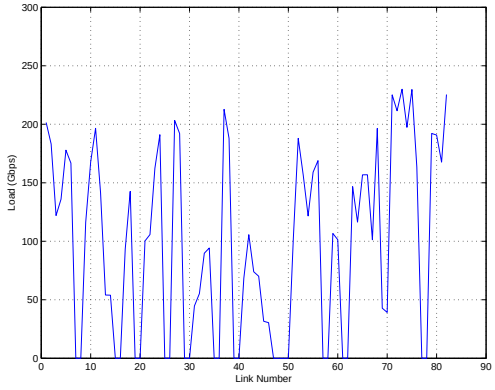
This different behavior of the two heuristics with respect to their corresponding optimum solutions is further emphasized by the analysis of the number of active links needed to satisfy



(a) SPR solution



(b) HPAND solution



(c) HPARND solution

Figure 4.9: *Load of the links for* $\beta = 1.0$ *- nobel-eu Scenario*

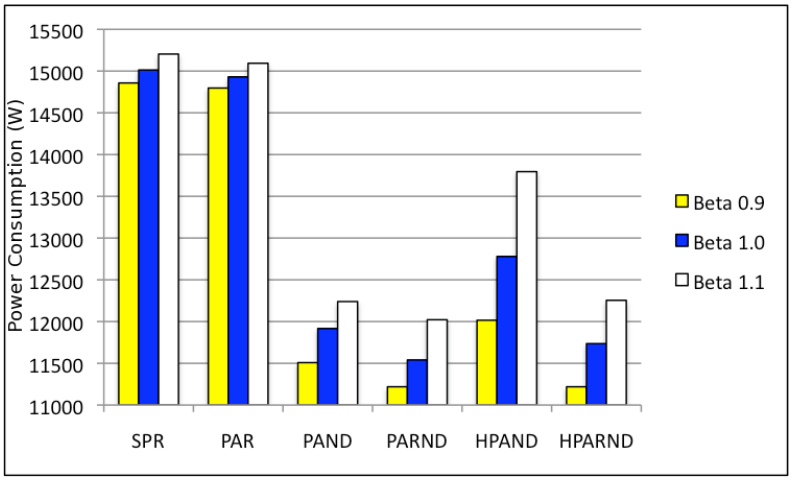


Figure 4.10: *Power consumption vs. the compared strategies - Exodus Scenario*

the traffic demands. In particular, whereas HPAND requires 40 active links to satisfy the traffic demands for $\beta = 0.9$, only 36 links are sufficient for PAND, HPARND and PARND. In the case $\beta = 1.1$, 48 links are needed for HPAND, whereas only 38 are required by the PAND and the PARND strategies. Furthermore, in this case HPARND requires 40 links; this result points out that when the traffic demand increases the worsening of the HPARND performance with respect to the PARND solution is less apparent than that observed by comparing the HPAND and the PAND solutions.

Figures 4.11 and 4.12 show the load of the links obtained respectively with the HPAND and the PAND strategy for $\beta = 1.1$. The comparison of the curves highlights that HPAND does not converge to a solution that provides an adequate concentration of routes permitting to switch off some links. This drawback is due to that, if the algorithm make the off target choice by powering off a suboptimal node/link, it will never backtrack to correct the mistake. The results of different simulation runs have highlighted that the local minimum obtained with this procedure is always slightly far from the optimum. Furthermore, as already observed, the performance worsens when the traffic demands increase. On the contrary, this drawback is limited with the proposed HPARND, due to the load balancing action of the PAR strategy used during the iterations.

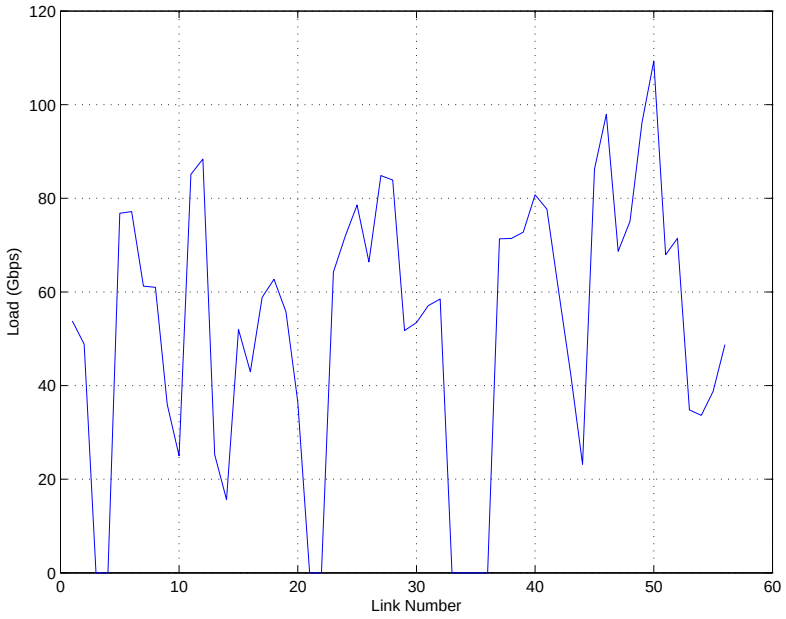


Figure 4.11: *Load of the links for* $\beta = 1.1$ *- Exodus Scenario, case HPAND*

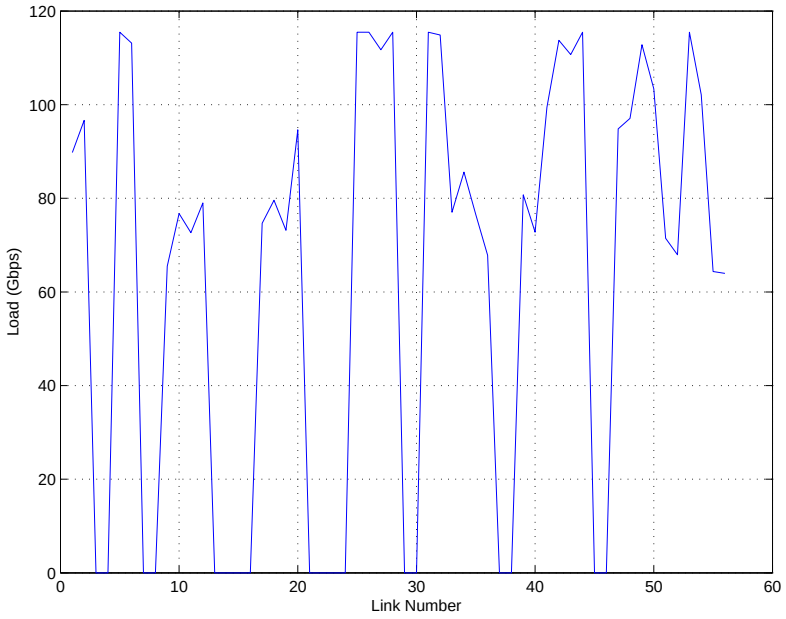


Figure 4.12: *Load of the links for* $\beta = 1.1$ *- Exodus Scenario, case PAND*

4.3 Performance analysis of PARND-BL problem

In this section, the following problems are addressed:

- *PARND* - Power Aware Routing and Network Design problem, where multiple paths for single demand are allowed ($f_{uv}^{sd} \in \mathbb{R} > 0$) and only entire links can be powered off ($n_{uv} \in \{0, N_{uv}\}$) as formulated in the previous chapter;
- *SP-PARND* - Single Path Power Aware Routing and Network Design problem, where a single path for traffic demand is permitted ($f_{uv}^{sd} \in \{0, d_{sd}\}$), differently of constraints (3.6) and only entire links can be powered off ($n_{uv} \in \{0, N_{uv}\}$);
- *PARND-BL* - Power Aware Routing and Network Design with Bundled Links problem, where multiple paths for single demand are allowed ($f_{uv}^{sd} \in \mathbb{R} > 0$) and single cables of bundled links can be powered off ($n_{uv} \in \mathbb{N}_0$) as presented in the previous chapter.

These problems have been solved by means of a MILP solver by using a piecewise linear approximation (20 segments) of $P_{v,T(v)}^{RP}$. It is relevant to note that the solution of the PARND-BL problem, with the constraint of a single path for each traffic demand, has not been obtained since it requires a lot of memory and computational resources.

In the analysis, the proposed approaches have been compared with MSC and a single path version of MSC (SP-MSC), where $f_{uv}^{sd} \in \{0, d_{sd}\}\ \forall (u,v) \in E\ \forall (s,d) \in D$

Note that, also in this section, no QoS features have been taken into account in all the approaches, i.e. $\rho = 1$.

4.3.1 Simulation settings

Considering the Exodus network scenario presented in previous section, the overall power consumptions of the network by using the above-mentioned approaches are compared.

Afterwards, the power behavior of MSC, PARND, and PARND-BL approaches are studied by varying several parameters of the network as follows:

- *Bundle Size* - Vary the number of cables that compose the bundled links, $N_{uv} \in \{2,3,4,5,6\}$ $\forall (u,v) \in E$;
- *Demand Factor* - Multiply the traffic demand, d_{sd} $\forall (s,d) \in D$, by a multiplicative factor, $\alpha \in \{0.7, 0.8, 0.9, 1.0, 1.1, 1.2\}$;

Table 4.3: *Power Consumption (W)*

MSC	SP-MSC	PARND	SP-PARND	PARND-BL
15086.4	15080.4	11540.2	11548.8	8958.3

- *Number of Core Nodes* - Vary the number of core nodes, which are nodes that do not send or receive traffic, *Core Nodes* $\in \{0,1,2,3,4,5\}$.

In the reference scenario *Core Nodes* is equal to 0, since all nodes are senders and receivers, therefore the ability of powering off nodes is not exploited. Hence, the traffic from/to several nodes has been deleted and they are considered as core nodes, in this way the ability of powering off nodes has been explored. The nodes with the highest degree have been selected as core nodes.

4.3.2 Simulation results

Table 4.3 shows the power consumptions by using the different approaches. The results show that MSC and PARND consume about the same power of their single path version. This result is due to the fact that in their optimal solution of the problems is often unsplittable, i.e. a single path per demand is used; furthermore, the observed maximum number of used paths per demand is at most two. The results show that PARND can save about the 25% of the power consumed by MSC, instead PARND-BL can save over the 40%. Since in this network scenario the ability of PARND and PARND-BL to completely power off nodes is not exploited, the power savings are due to a routing strategy that allows to power off links/cables and reduce the power consumption for route processing.

In Figure 4.13, the results obtained for $\alpha = 1.0$ show that, using PARND-BL, the power consumption does not change when the bundles size increases, because additional cables are powered off. Instead, the power consumption linearly increases with the bundle size when MSC and PARND are used, because the power consumption related to the links grows. Using PARND, the power consumption increases slowly, because the number of active links is lower than using MSC. Indeed, the PARND solution permits to power off some links by moving their traffic load towards alternative paths.

The results in term of power savings obtained for different demand factors are summarized in Figure 4.14. Observing the figure, the power savings of PARND and PARND-BL decrease

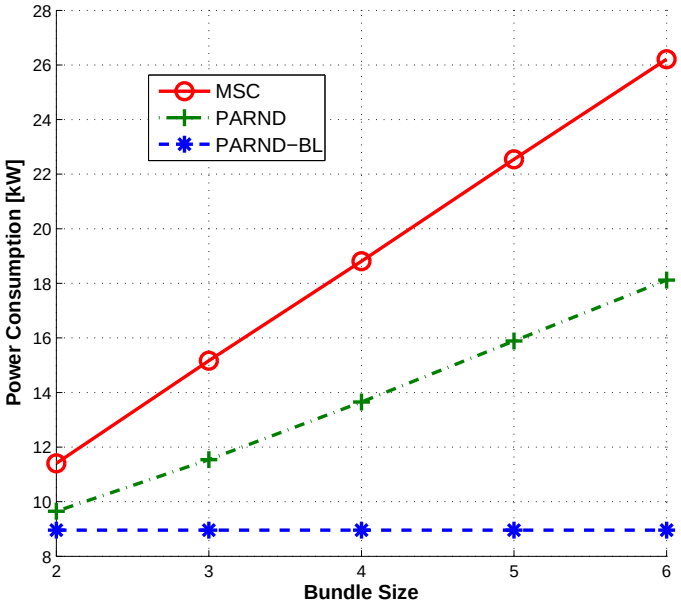


Figure 4.13: *Power Consumption vs. Bundle Size (N_{uv} $\forall uv \in E$),* $\alpha = 1.0$

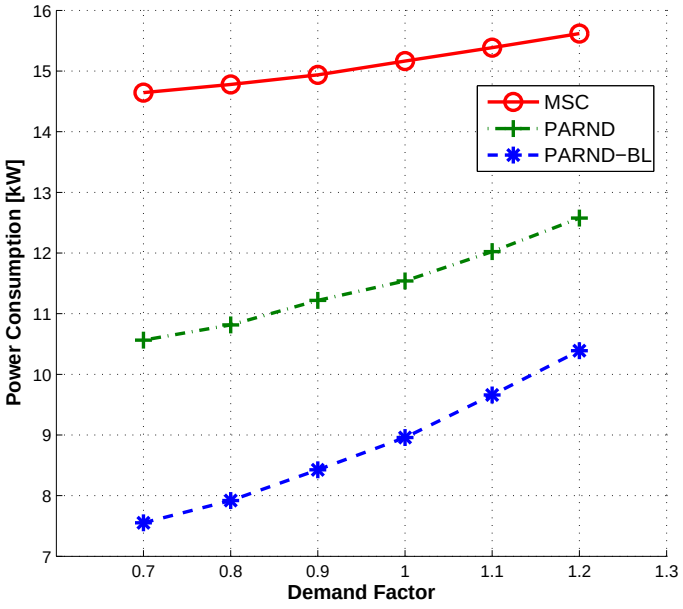


Figure 4.14: *Power Consumption vs. Demand Factor* (α), $N_{uv} = 3\ \forall uv \in E$

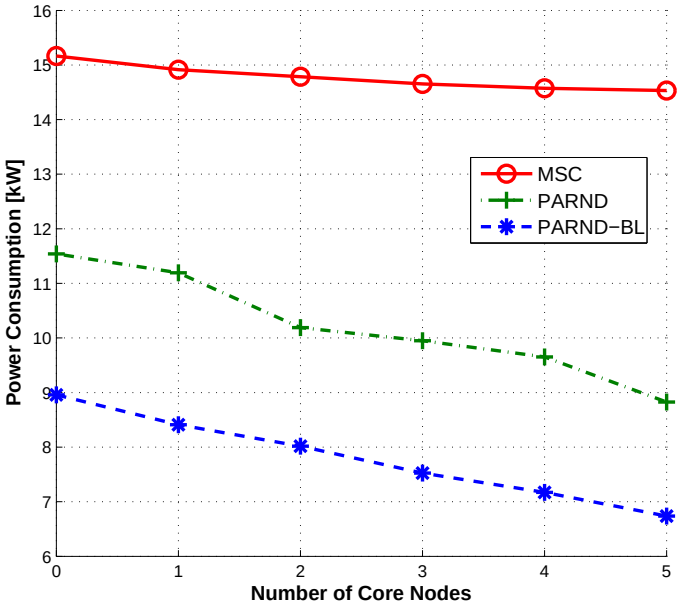


Figure 4.15: *Power Consumption vs. Number of Core Nodes*

when the traffic demand increases. Indeed, in this situation, i.e. high traffic demand, more and more links/cables have to be powered on. Especially using PARND-BL, power consumption increases faster than PARND because many cables have to be powered on to satisfy the traffic demand growth.

The last analysis has been focused on the evaluation of the power savings in the case some nodes of the networks are assumed as core nodes, i.e. they cannot assume the role of receiver or source of traffic demands. This study has been carried out with $N_{uv} = 3\ \forall uv \in E$, and $\alpha = 1.0$. The results, summarized in Figure 4.15, highlight the ability of PARND and PARND-BL to fully power off some nodes. In particular, PARND-BL saves up to about 55% of the power consumed by using MSC when there are five core nodes in the network. It is worth emphasizing that the power consumption reduction observed in the MSC curve is due to the fact that increasing the number of core nodes the total traffic demand decreases.

4.4 Evaluation of NPM models

In the following, the empirical behavior of the NPM approaches described in Section 3.3 is analyzed. The methods have been compared to SPR, the reference routing algorithm described before. SPR represents in fact the widely used approach in core networks, when no specific administrative or cost constraints are present.

The proposed models have been solved as follows:

- *PAR* - to optimality by Ipopt [6], a software package for large-scale nonlinear optimization problems, which implements an interior-point line-search filter method;
- *PAND* - to optimality by the mixed integer linear programming solver CPLEX;
- *PARND* - to optimality by the mixed integer nonlinear programming solver BONMIN 1.5.0 [1] (only for small instances) or by means of a linear approximation via CPLEX;
- *PARND-BL* - to optimality by the mixed integer nonlinear programming solver BONMIN 1.5.0 (only for small instances) or by means of a linear approximation via CPLEX.

In particular, the Integer Linear Programming models have been solved by using the IBM ILOG CPLEX Optimization Studio V12.2 [5], whereas the solution of the PAR problem has been calculated by using the Ipopt version 3.6stable [6]. The linear approximation (used for addressing the large PARND and PARND-BL instances) consists in approximating the cubic function representing the power consumption of the route processing, i.e. $P^{RP}_{v,T(v)}$ (see Figure 4.6), via a piecewise linear function composed of 20 segments, as in [28]. When possible, we also found the optimal solution of the nonlinear PARND and PARND-BL models by using BONMIN 1.5.0 [1]. The results have shown that the power consumption difference between the solutions obtained by the linear approximation of the models and those obtained by BONMIN are order of 0.01%. However, the computational time needed to BONMIN for determining the optimal solution is often two order of magnitude higher than the time spent by CPLEX to calculate the solution of the linear approximation. Furthermore, in some cases, such as PARND under the Sprintlink scenario, BONMIN produced no results after 160 hours of CPU time. As a consequence, in the remaining of the study we shall report only the results related to the linear approximation of PARND and PARND-BL.

In order to provide Quality of Service (QoS) solutions, in all the tested models we have limited the link utilization by multiplying C^{PIC}_{uv} by a factor $\rho \in (0,1)$ in the link capacity constraints (3.5) and (3.9). The value of ρ, which represents the link utilization, has to be

appropriately determined in order to guarantee the QoS (i.e. a limited delay). To this aim, we have modeled the transmission of the traffic on the link as a M/M/1 queue, where the service rate μ is the link capacity. By considering that the minimum link capacity of the network is C_{uv}^{PIC} (for those links which are composed of a single PIC), the mean delay is equal to $\frac{1}{C_{uv}^{PIC} \cdot (1-\rho)}$. Therefore, if we consider $\rho = 0.95$, the mean delay is equal to $51.97ns$, which is widely sufficient to guarantee a low end-to-end delay.

In conclusion, it relevant to be outlined that, since there are no commodity-dependent costs or capacities in the models introduced in the previous chapter and in order to reduce the computational effort required to solve them, their aggregated versions have been considered and implemented, where all the commodities having the same origin node are considered to be "the same kind of flow".

4.4.1 Simulation settings

The performance analysis of the presented problems has been carried out referring to a set of real core network scenarios. The first considered scenario is nobel-eu network topology presented in Section 4.1.1. The other considered network scenarios are backbone topologies obtained from the set of data collected during the Rocketfuel study [7]. In particular, the following topologies are considered: Exodus (US), Ebone (EU), Abovenet (Australia), and Sprintlink (US), which correspond respectively to the dataset AS 3967, AS 1755, AS 6461, and AS 1239 of the Rocketfuel study.

The nodes of all the tested networks are assumed to have the same energy profile, and the links are supposed to be symmetric, i.e. $C_{uv}^{PIC} = C_{vu}^{PIC}$, $P_{uv}^{PIC} = P_{vu}^{PIC}$, and $n_{uv} = n_{vu} \ \forall (u,v) \in E$. This last assumption implies that all the tested networks are indeed composed of undirected links, which are modelled in terms of two directed links, one for each direction.

Table 4.4 shows the statistics of the considered real core networks, i.e. the number of the nodes (# *Nodes*), the number of the undirected links (# *Links*), and the number of the nodes which are connected to the rest of the network with a single undirected link (# *S.L.N.*). In considering the influence of the network size in the model solution, it has to be emphasized that networks characterized by higher values of # *S.L.N.* are indeed more simple from the design viewpoint, since the single links connecting the mentioned nodes to the rest of the networks can not be powered off in case those nodes are sources or destinations.

The components of the considered routers (i.e. chassis, route processors) are the same of the previous evaluation, therefore their power characterization is also the same.

Table 4.4: *Statistics of the Network Scenarios*

	nobel-eu	**Exodus**	**Ebone**	**Abovenet**	**Sprintlink**
# Nodes	28	22	23	22	43
# Links	41	37	38	42	83
# S.L.N.	0	1	4	5	13

Concerning the origin-destination demands, for the *nobel-eu* topology the traffic matrix has been obtained as in the Section 4.2.1. Instead, to obtain the traffic matrix for the other topologies acquired from the Rocketfuel study, from a computational perspective we set the mean value of the traffic demand from s to d, i.e. $\overline{d_{sd}}$, to $\frac{\overline{d^s}}{|V|-1}$. Then, each element of the matrix d_{sd} has been extracted from a uniform distribution: $d_{sd} = U\left[0.5 \cdot \overline{d_{sd}},\ 1.5 \cdot \overline{d_{sd}}\right]\ \forall (s,d) \in D$. The choice of a traffic matrix where each node of the network is both source and destination of traffic implies that only links (or PICs) can be powered off. Hence, although the PAND, PARND and PARND-BL problems permit to power off both links and nodes, the computational analysis will be focused only on the possibility to power off the formers. Moreover, this kind of traffic demand generally implies a high computational effort for the model solution, because the number of the flow variables is proportional to the number of the sources multiplied by the number of links. In the considered topologies, the number of the flow variables varies in fact from 814 (Exodus) to 3569 (Sprintlink). This justifies some high computational times that will be reported, despite the apparently small size of some networks (which are however realistic, since they represent real core networks).

As far as the link capacity is concerned, the maximum number of PICs per link has been computed as follows:

$$N_{uv} = \lceil \frac{\max(f_{uv}, f_{vu})/\beta}{C^{PIC}_{uv}} \rceil \quad \forall (u,v) \in E, \tag{4.4}$$

where $f_{ij} = \sum_{(s,d)\in D} f^{sd}_{ij}$ denotes the total flow on link (i,j) when the Shortest Path Routing (SPR) is applied, while $\beta = 0.5$ denotes the overprovisioning factor (see [21]). In more details, to calculate the f^{sd}_{ij} we assumed that each traffic demand d_{sd} uses a single shortest path from s to d. The weight of each link is assumed equal to 1. Therefore, a minimal cardinality path has been computed for each origin-destination pair (s,d). For the tested, symmetric, networks we set $N_{uv} = N_{vu}$ for each undirected link (u,v).

Table 4.5 shows the statistics about the number of PICs per link for the different topologies,

Table 4.5: *Number of PICs per Link*

	nobel-eu	Exodus	Ebone	Abovenet	Sprintlink
Max	16	7	11	7	23
Avg	4.66	3.46	3.58	2.5	3.46
# S.P.L.	7	4	8	12	15

i.e. the maximum (*Max*) and the average (*Avg*) number of PICs per link, and the number of links composed of a single PIC (*# S.P.L.*).

4.4.2 Simulation Results

The performance and the behavior of the different approaches, on the various topologies, are analyzed by studying the following indicators:

- overall power consumption;
- quality of the solutions, expressed in terms of powered on links/PICs;
- CPU time;
- number of flow paths per origin-destination pair.

Overall power consumption

The network power consumptions (in W) obtained with the different approaches and for the considered topologies are summarized in Table 4.6. The results show that the power saving produced by PAR with respect to SPR is negligible. Indeed, the maximum power savings is about 3.5% (for the Ebone network), whereas the minimum is about 0.4% (for Abovenet and Exodus scenarios). The low power savings are mainly due to the low load of the networks, which implies that the nodes have on average a throughput value where the function $P^N_{v,T(v)}$ is quite linear and with a low slope. As an example, Figure 4.16 shows the histogram of the load of each network node when the SPR, the PAR and the PAND approaches are used in the Abovenet scenario. The figure shows that the majority of the nodes (above the 95%) have a throughput under 500 Gb/s, whereas the remaining nodes work under the 800 Gb/s. By taking into account the function $P^N_{v,T(v)}$, shown in Figure 4.6, we can observe that these throughput

Table 4.6: *Power Consumption (W) vs Topology*

	nobel-eu	Exodus	Ebone	Abovenet	Sprintlink
SPR	33704	22200	24565	19576	52966
PAR	32856	22120	23706	19504	51166
PAND	30731	17939	22174	23072	55518
PARND	28397	17547	20543	15876	41994
PARND-BL	19336	12841	14497	12307	30761

values are in a quite linear region of the curve, with a very low slope. This observation can explain the low power savings produced by the PAR solution, which takes into account the nonlinear behavior of the route processors, w.r.t. SPR. A similar conclusion can be drawn by taking into account the Exodus network topology. Indeed, Figure 4.17 shows that no node has a throughput higher than 600 Gb/s. Hence, also in this case the power savings obtained by means of the utilization of a power aware routing are negligible.

As far as the PAND approach is concerned, the results of Table 4.6 show that the power savings depend on the considered topology. For example, we can observe the high gains of PAND with respect to SPR for the Exodus topology, whereas PAND is less efficient by considering the Abovenet and the Sprintlink networks. This is essentially due to the different impact of PAND on the load of the nodes in the analysed network topologies, as revealed by Figures 4.16 and 4.17.

In order to explain this different behavior, the actions of the PAND solution on the considered topologies is analyzed. In particular, Figures 4.18 and 4.19 report the histogram of the number of PICs per link before and after the application of the PAND solution, in the Exodus and in the Abovenet scenario, respectively.

The comparison of these figures shows that, in the Exodus network, the PAND solution leads to power off the links composed of a high number of PICs. In this network scenario, the action of PAND is to power off links with high capacity. As a consequence, all nodes have a similar load. This feature of the node load jointly with powering off some links leads to power savings. And in fact, in Table 4.6 we can observe a power savings of PAND w.r.t. SPR of order of 19%.

On the contrary, Figure 4.19 highlights a diverse impact of the PAND solution on the Abovenet topology. Indeed, the majority of the powered off links are composed of a low

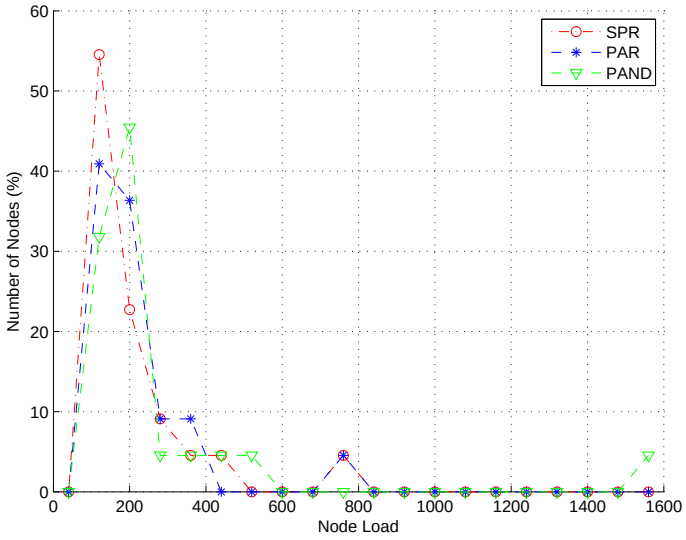


Figure 4.16: *Histogram of the load of nodes with the SPR, PAR and PAND approaches - Abovenet Topology*

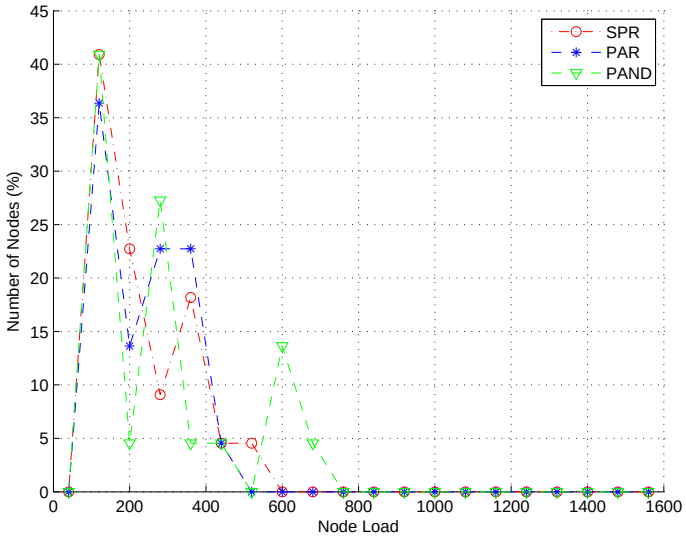


Figure 4.17: *Histogram of the load of nodes with the SPR, PAR and PAND approaches - Exodus Topology*

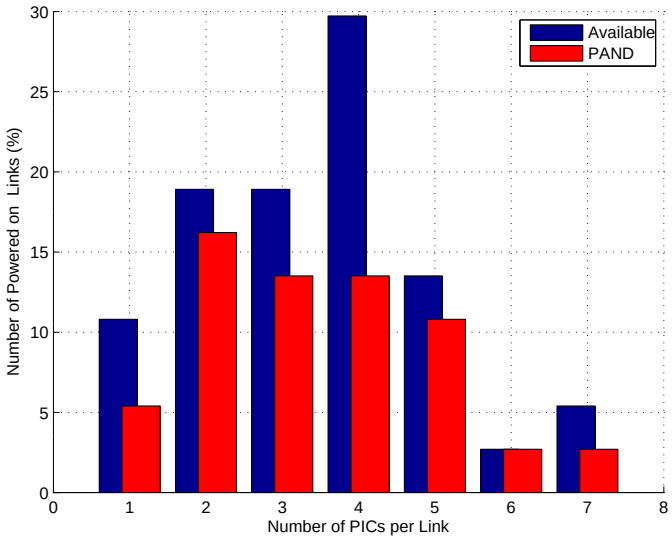


Figure 4.18: *Histogram of the PICs composing the powered on links - Exodus Topology*

number of PICs (i.e. 1 or 2), whereas the high capacitated links (i.e. composed of 5 or 7 PICs) remain powered on. The effect of this action is that, after the application of the PAND solution, the resulting topology is characterized by few links with high capacity. Hence, the nodes connected to these links tend to be overloaded, as confirmed by the presence of about the 5% of nodes with a throughput near the maximum one (see Figure 4.16). The increase of the power consumption due to the route processing component in these high throughput nodes thus nullifies the power savings due to the power off of links with low power consumption. As a consequence, the PAND solution leads to an increase of the overall network power consumption. A similar behavior has been observed in the Sprintlink topology.

The modelling of the true, nonlinear power behavior at the nodes, leading to model PARND, permits to improve considerably the performance of the network design strategy. Indeed, as shown by Table 4.6, the PARND solution permits to appreciably reduce the network power consumption in all the considered network scenarios: the minimum power savings is about 15.75% (nobel-eu scenario), whereas the maximum one is about 20.71% (Sprintlink scenario). The power savings obtained with the PARND-BL solutions are also considerable. In fact this approach permits to save, in some cases, about the 40% of the overall network

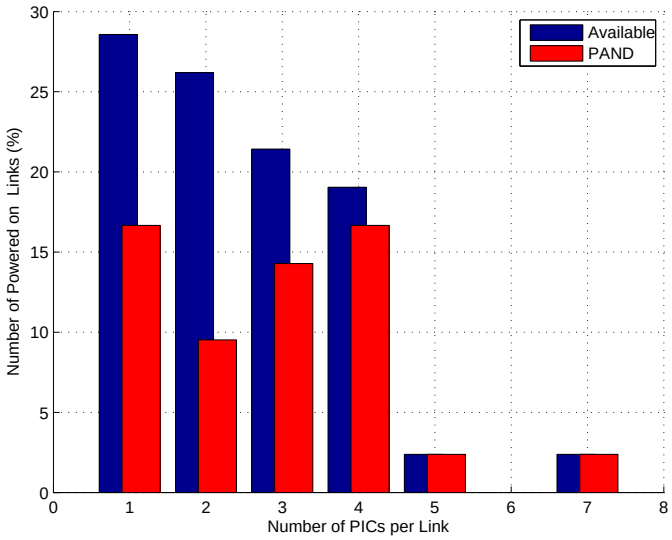


Figure 4.19: *Histogram of the PICs composing the powered on links - Abovenet Topology*

power consumption.

In order to better understand the mechanisms leading to these computational results, Figure 4.20 shows the histogram of the number of PICs per powered on links in the case of PAND, PARND, and PARND-BL; the histogram of the original number of PICs per links is also reported. In this network scenario, the figure highlights that PARND and PAND try, when possible, to power off the links composed of a large number of PICs (i.e. 7 and 10). Furthermore PAND strives for powering off a higher number of PICs (and links) than PARND (see Table 4.7). However, the joint work of the power aware routing permits to PARND to save about the 20.72% of power w.r.t. SPR. On the contrary, the obliviousness of the power consumption component due to the route processing incorporated by PAND leads this model to increase the overall network power consumption w.r.t. SPR, although a lower number of PICs are powered on w.r.t. PARND. This result is due to a worse distribution of the load among the nodes of the network, as shown in Figure 4.21. In fact, although for node throughput over 800 Gb/s PAND and PARND have a similar histogram, the former leads the 5% of nodes to work at the maximum throughput.

The best performance is however obtained by PARND-BL, which is able to power off

single PICs, by maintaining almost all links powered on. In fact, Sprintlink has 83 links (see Table 4.4), and only 8 links are powered off by the PARND-BL solution (see Table 4.7). Furthermore, Figure 4.20 shows that the majority of the powered on links are composed of one or two PICs which, in turn, permits a better distribution of the load among the nodes, as shown in Figure 4.21. In summary, the main conclusion we can drawn about the overall network power consumption is that an energy efficient network with routers able to scale the power consumption as a function of the traffic load should be composed of many "thin" links which are completely exploited, compatibly with the required QoS constraints. In particular, as shown by Figure 4.22, where the histogram of the link utilization obtained with the considered strategies is plotted, in the PARND-BL solution about the 60% of the links have an utilization near to the maximum allowed, which is set equal to $\rho = 0.95$. Except a low percentage of links (order of 10%) having a link utilization less than 0.1, the remaining links have in fact an utilization higher than 0.7. The results for the other network topologies are very similar. An analogous behavior can be observed for the PAND and the PARND curves, although some link utilizations fall in the range $[0.35, 0.7]$. On the contrary, the PAR curve highlights that the utilization of most links is in the range $[0.10, 0.7]$, and that only the 10% of the links achieve the maximum allowed utilization.

Quality of the solutions in terms of powered on links/PICS

Table 4.7 shows the number of powered on links/PICs after the application of the solutions provided by the different approaches. The results highlight that PAND permits to obtain the minimum number of links powered on in all the considered topologies, as previously observed. However, also considering the results in Table 4.6, even if PAND powers off more links and PICs than PARND, it does not permit to produce more power savings than PARND. To obtain a power reduction, as we explained in Section 4.4.2, the power consumption due to route processing, which depends on the node throughput, has to be explicitly addressed in the problem formulation, as modelled in PARND and in PARND-BL.

Moreover, PAND have two further drawbacks from the QoS and reliability/recovery points of view. For the QoS aspects, users can experience a worse quality of service because, by reducing the number of links (and eventually nodes) without taking into account the power consumption of the route processors, tthe nodes and the links may be overloaded. For the reliability/recovery aspects, PAND is weak because, by minimizing the number of powered on network elements, the network could be more often out of order and slower to

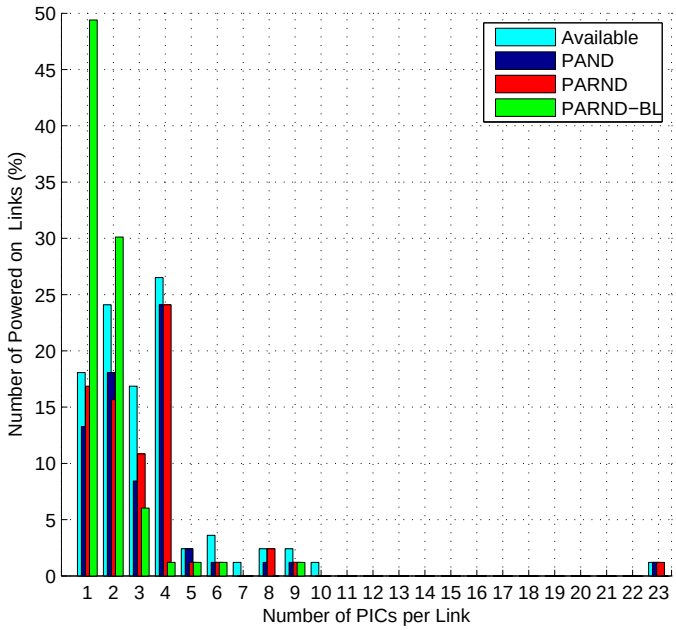


Figure 4.20: *Histogram of the PICs composing the powered on links - Sprintlink Topology*

Table 4.7: *Number of Powered On Links/PICs*

	nobel-eu	**Exodus**	**Ebone**	**Abovenet**	**Sprintlink**
PAND	32/140	24/82	27/104	26/73	59/198
PARND	33/142	27/86	28/105	30/74	62/206
PARND-BL	41/87	36/57	34/65	34/49	75/130

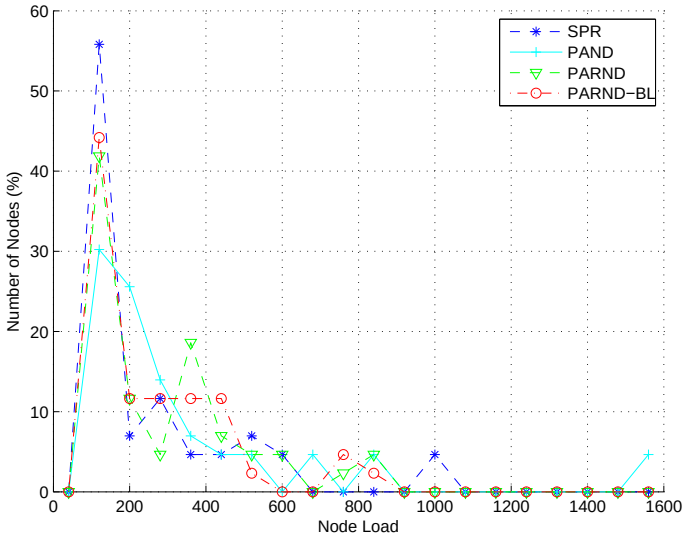


Figure 4.21: *Histogram of the load of nodes with the SPR, PAND, PARND and PARND-BL approaches - Sprintlink Topology*

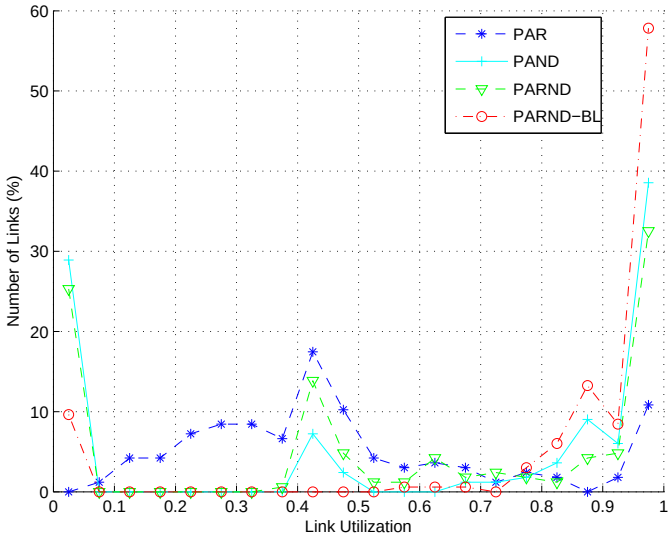


Figure 4.22: *Histogram of the link utilization with the PAR, PAND, PARND and PARND-BL approaches - Sprintlink Topology*

Table 4.8: *CPU times (s)*

	nobel-eu	Exodus	Ebone	Abovenet	Sprintlink
SPR	0.01	0.01	0.01	0.01	0.09
PAR	2.20	0.79	0.99	0.57	11.59
PAND	137.03	140.33	35.97	28.54	17806.17
PARND	139.94	211.02	7.07	15.39	12635.88
PARND-BL	305.94	648.25	111.55	561.27	45437.32

get recovered than in the other methods. In this respect, it is important to underline that PARND-BL is also preferable to PARND because to power on a PIC is faster than to power on a link. This is due to the fact that powering on a link is a topological change whereas powering on a single PICs is only an increase of the link capacity that can be done locally.

CPU time

Table 4.8 shows the CPU times required by the different approaches. The measurements have been carried out using a PC with 4 Intel Core i7 CPUs @ 3.07GHz (hyperthreading enabled), 8GB RAM, and an ASUS P6T DELUXE V2 Motherboard. The results highlight that, as natural, the running time grows with the complexity of the power aware problem to be solved. The only exception is given by PARND, which is faster than PAND. As mentioned in Section 3.1, the maximum acceptable CPU time for computing a NPM solution is order of 6/12 hours. Therefore, the CPU times of PAND and PARND are barely acceptable. Instead, the CPU time of PARND-BL may be unacceptable in some cases, as for the Sprintlink topology (see Table 4.8).

Number of paths per traffic demand

A relevant indicator to be considered in actual networks is the number of paths which are used to support the computed multicommodity flows. Indeed, in modern broadband communication networks, in order to apply the routing solutions provided by the diverse approaches we should refer to protocol architecture such as MPLS (MultiProtocol Label-Switched) [46]. In this architecture the paths, denoted as Label Switched Paths (LSP), are created and managed by means of signaling protocols (such as RSVP-TE, Resource reSerVation Protocol for

Table 4.9: *Number of Paths per Flow (avg/max)*

	nobel-eu	Exodus	Ebone	Abovenet	Sprintlink
PAR	1.14/4	1.07/2	1.11/3	1.06/2	1.16/4
PAND	1.01/3	1.01/2	1.02/2	1.02/3	1.02/2
PARND	1.01/2	1.03/2	1.03/2	1.04/2	1.02/3
PARND-BL	1.04/2	1.07/3	1.05/3	1.06/3	1.04/3

Traffic Engineering [14]), and support the traffic demand for a given pair of nodes. However, the splitting of a traffic demand in too much LSPs can deteriorate the performance of the networks, due to the overhead of the signaling protocols used to create and manage LSPs. Hence, the solutions provided by the diverse approaches should limit to few units the number of paths used to support the traffic of each origin-destination pair [10], [51], [50]. The results of the analysis from this perspective are summarized in Table 4.9. The table clearly shows that the average number of paths per traffic demand is very low for all the approaches on all the topologies. Furthermore, only PAR gives a solution with a maximum number of paths per traffic demand equal to 4. In most cases, the maximum number of paths per traffic demand is equal to 2 or 3. These values are acceptable in an actual MPLS-based network architecture. Thus, the proposed approaches proved to successfully address also this issue.

4.5 Evaluation of PARND-BNBL problem and its heuristics

In the following, a performance analysis of the PARND-BNBL and of the its different heuristic approaches (i.e. FGH, TLPH, and PMH) is presented. These methods are compared to SPR and PAR under different core network scenarios and the ability of the proposed heuristics (i.e. TLPH and PMH) to provide a solution near the optimum is investigated.

As in the previous section, PAR and PARND-BNBL have been solved by IpOpt and CPLEX, respectively.

Furthermore, as in the previous section, all the methods (also the heuristics) have QoS features (i.e. $rho = 0.95$). It is relevant to note that in the analysis the pure FGH is also considered, i.e. calculated using $\rho = 1$. The results obtained with $\rho = 0.95$ will be indicated as FGH-QoS.

4.5.1 Simulation settings

The performance analysis has been carried out referring to a set of core network scenarios. These network scenarios are nobel-eu, Exodus (US), Ebone (EU), Abovenet (Australia), and Sprintlink (US), the same taken into account in the previous section. Futhermore, a last network scenario has been considered: a large Austrian core topology (ta2) taken from [8] and given by the Telekom Austria (see Figure 4.23).

The statistics of the considered network are the same shown in Table 4.4. In addition, $ta2$ topology has the following statistics: 65 nodes, 108 undirected links, and only one $S.L.N.$.

The components of the considered routers and their power characterization are the same of the previous evaluation. Furthermore the traffic demands and the number of PIC per link are computed in the same way as in the previous section. In particular, the statistics about the number of PICs per link for the $ta2$ topology are the following: $Max = 23$, $Avg = 4.7$, and $\# S.P.L. = 20$. The statistics for the other topologies are shown in table 4.5.

4.5.2 Simulation results

The following parameters has been studied to analyze the performance and the behavior of the different approaches on the various topologies:

- evaluation of TLPH;
- overall power consumption of the network;
- number of powered on links/PICs;
- CPU time.

Evaluation of TLPH

A first study was aimed to evaluate the impact of the multiplicative factor γ on the performance of the TLPH. Table 4.10 shows the power consumption and the CPU time for calculating the solution of TLPH. The results refer to all considered topology and for some values of γ (the symbol ∞ is the case where no time bound is set).

The results highlight that TLPH is very slow when the running time for the solution of the PAR problem is not limited, for example the CPU time of 108163 s the case of the ta2 topology is unacceptable in an actual network scenario.

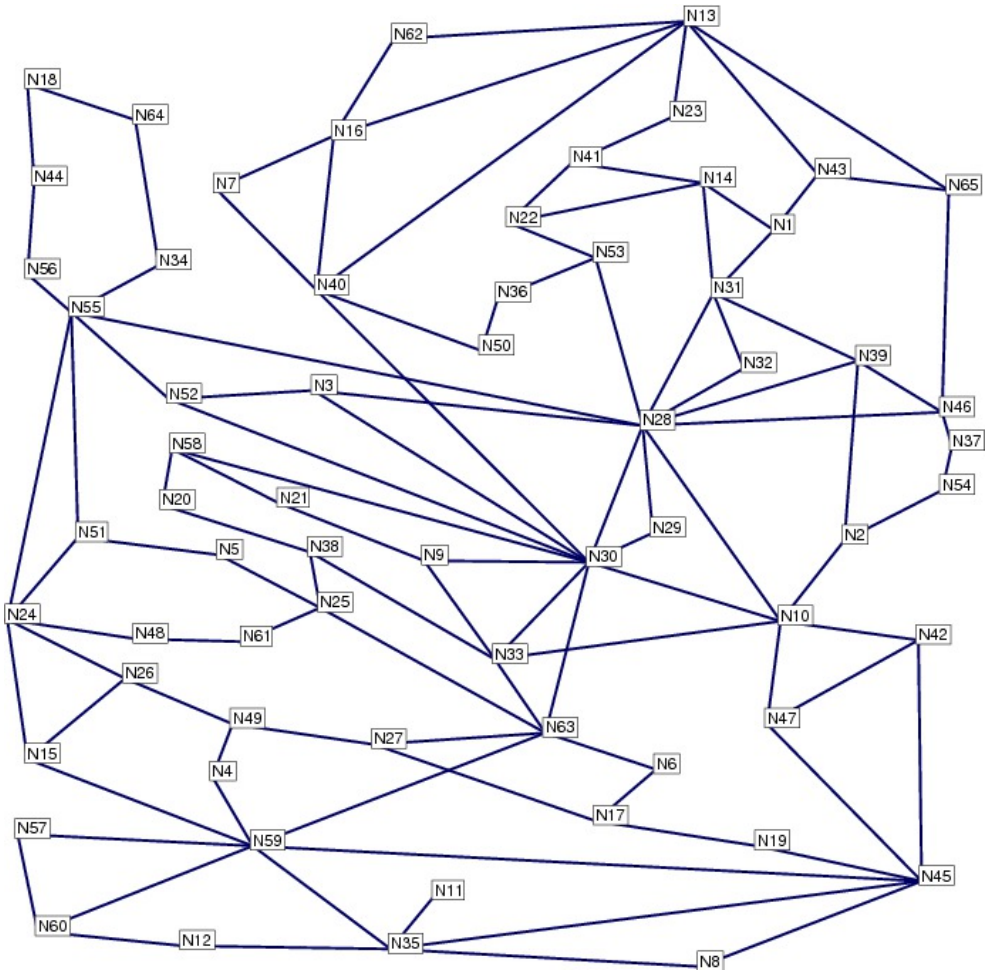


Figure 4.23: *Telekom Austria Topology*

Table 4.10: *Power Consumption and CPU time of TLPH vs* γ

γ	nobel-eu	Exodus	Ebone	Abovenet	Sprintlink	ta2
∞	19816W in 4722s	12839W in 2556s	14580W in 4269s	12426W in 2803s	31135W in 55931s	61317W in 108163s
5	19816W in 627s	12839W in 375s	14580W in 382s	12426W in 387s	31135W in 4575s	61317W in 10694s
4	19816W in 507s	12839W in 303s	14580W in 308s	12426W in 313s	31135W in 4047s	61317W in 8781s
3	19816W in 388s	12839W in 231s	14580W in 235s	12426W in 240s	31135W in 3150s	61225W in 6841s
2	19816W in 267s	12839W in 87s	14580W in 160s	12426W in 165s	31252W in 2183s	61225W in 5161s
1	19816W in 147s	12839W in 87s	14580W in 86s	12426W in 91s	31956W in 1184s	62593W in 2650s

Further, the results show that bounding the time using $\gamma \in \{4,5\}$ leads to obtain the same power consumption obtained with $\gamma = \infty$, but in a time about 10 times smaller. Using smaller γ, the power consumption of the obtained solution is the same, but the CPU time reduces up to 50 times for small topologies (i.e. nobel-eu, Exodus, Ebone, and Abovenet). For the Sprintlink and the ta2 topologies, the differences in terms of the power consumption given by the obtained solution for diverse γ are due to the presence of the false negative events in the PAR computation. Furthermore, in the ta2 topology for $\gamma \in \{3,2\}$ the false negative events leads to a reduction of the power consumption. This phenomenon is due to the fact that, in the initial iterations of the algorithm 6, there was a false negative event, i.e. some PICs of a bundled link remain powered on. In the successive iterations, this event could allow to power off "new" PICs and, consequently, to save more power. In order to better understand this phenomenon, we should take into account that FGH approach used as the basis of the proposed heuristics presents the disadvantage that if the algorithm makes the "wrong" choice by removing a suboptimal PIC, it will never backtrack to correct the mistake.

In summary, TLPH permits to achieve a good trade off between CPU time and power consumption by using $\gamma = 2$. In the following, it will be denoted only as TLPH.

Overall power consumption of the network

The results of the network power consumption (in W) obtained with the different approaches are summarized in Table 4.11. In the case of the ta2 scenario, the utilization of the SPR strategy violates the constraints on the node capacity, since there are a lot of paths traversing some nodes in the middle of the topology. In general, the results show that the power savings produced by PAR with respect to SPR are almost negligible. Indeed, the maximum power savings is about 13.4% (for the ta2 topology), whereas the minimum is about 0.4% (for Abovenet and Exodus scenarios). The low power savings are mainly due to the low load of

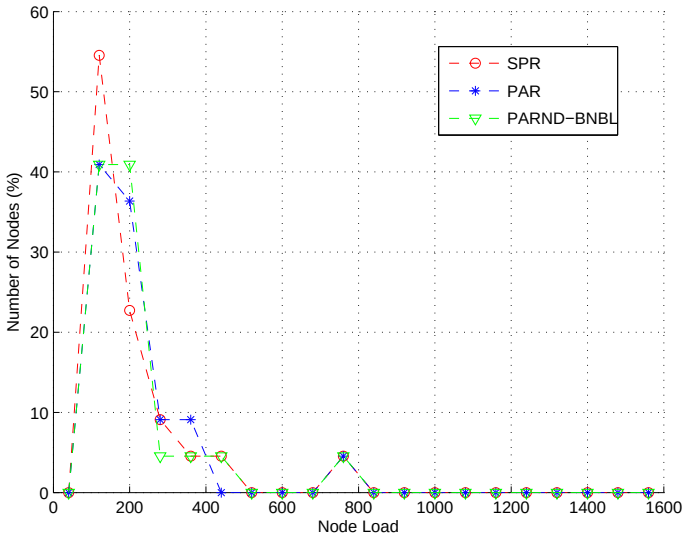


Figure 4.24: *Histograms of the load of nodes with the SPR, PAR and PARND-BNBL approaches - Abovenet Topology*

the networks, which implies that the nodes have on average a throughput value where the function $P^N_{v,T(v)}$ is quite linear and with a low slope.

As an example, Figure 4.24 shows the histograms of the load of nodes when the SPR, PAR and PARND-BNBL approaches are used in the Abovenet scenario. The figure shows that the majority of nodes (above the 95%) have a throughput under 500 Gb/s, whereas the remaining nodes work under the 800 Gb/s. By taking into account the function $P^N_{v,T(v)}$, shown in Figure 4.6, these throughput values are in a quite linear region of the curve, with a very low slope. This observation can explain the low power savings produced by the PAR solution w.r.t. SPR. A similar behavior hes been observed in the other scenarios; these results are omitted for sake of simplicity.

The table clearly shows the power savings introduced by the PARND-BNBL problem and, in the case of large topologies (i.e. Sprintlink and ta2), the distance between the proposed heuristics and the optimal solution. Further, in the ta2 topology, the optimal solution can not be found, but the software produced the error "out of memory" although the PC had a 8GB RAM. On the contrary, the results can be obtained by using the considered heuristics, except

Table 4.11: *Power Consumption (W) vs Topology*

	nobel-eu	Exodus	Ebone	Abovenet	Sprintlink	ta2
SPR	33704	22200	24565	19576	52966	108287
PAR	32856	22120	23706	19504	51166	93820
PARND-BNBL	19336	12841	14497	12307	30761	N/A
FGH	19245	12685	14790	12223	32246	70021
FGH-QoS	19813	12988	14935	12567	32717	71036
TLPH	19816	12839	14580	12426	31252	61225
PMH	N/A	13596	15130	12912	32323	66195

in the case of nobel-eu scenario with PMH strategy. In this case, the heuristic used to solve the PAR problem was unable to allocate some traffic demands, since it over-utilizes the links given that it creates very long paths.

In general, the proposed heuristics have similar performance of the FGH approaches when the topology is small. When the network scenario is more complex, such as in the ta2 topology, the proposed TLPH and PMH algorithms provide a power savings of about 14% with respect to the respect to the simple FGH approaches.

Number of powered on links/PICs

Table 4.12 shows the number of powered on links/PICs after the application of the solutions provided by the different approaches. The results highlight that the proposed TLPH and PMH heuristics produce the highest number of links (and PICs) powered on in all the considered topologies. However, even if the other solutions power off more links and PICs than proposed heuristics, they do not permit to produce more power savings than TLPH and PMH algorithms. This phenomenon confirms that, in the case of router hardware that implements power aware techniques, such as DVS and DFS, the powering off of links/PICs does not always determine a reduction of the overall network consumption.

CPU time

Table 4.13 shows the CPU times required by the different approaches. The measurements have been carried out using a PC with 4 Intel Core i7 CPUs @ 3.07GHz (hyperthreading

Table 4.12: *Number of Powered On Link/PICs*

	nobel-eu	Exodus	Ebone	Abovenet	Sprintlink	ta2
PARND-BNBL	41/87	36/57	34/65	34/49	75/130	N/A
FGH	40/83	34/55	36/64	33/47	74/127	100/227
FGH-QoS	39/87	34/57	35/65	33/49	72/129	99/239
TLPH	40/91	36/57	35/66	35/50	75/133	105/255
PMH	N/A	35/62	37/70	38/54	81/140	108/283

Table 4.13: *CPU times (s)*

	nobel-eu	Exodus	Ebone	Abovenet	Sprintlink	ta2
SPR	0.01	0.01	0.01	0.01	0.09	0.41
PAR	2.20	0.79	0.99	0.57	11.59	17.77
PARND-BNBL	305.94	648.25	111.55	561.27	45437.32	N/A
FGH	4933.20	2240.68	4472.87	2175.40	49450.94	84827.68
FGH-QoS	3981.52	2194.84	3937.29	2412.73	46421.20	108556.36
TLPH	266.52	158.45	160.37	164.31	2014.31	4878.30
PMH	N/A	0.40	0.37	0.46	8.95	57.21

enabled), 8GB RAM, and an ASUS P6T DELUXE V2 Motherboard. The results highlight that the running time for the solution of the PARND-BNBL problem can be unacceptable in some network scenarios, such as Sprintlink or ta2. On the contrary, we can observe the gain in term of CPU times given by the FGH approaches and the very short CPU time required by the PMH approach in finding a solution. The TLPH approaches produces interesting CPU time gains with respect to the FGH approach, although these results are worse then those of PMH. However, this worsening in terms of CPU time represents the cost to obtain a more energy efficient solution with respect to the TLPH, as shown by the results in table 4.11. As an example, in the case of the ta2 scenario, the power consumption obtained by the TLPH solution is about 7.5% lesser than PMH, although it requires a CPU time almost 100 times higher.

4.6 Evaluation of PARND-BL problem and its heuristics

In this section, an extensive simulation study of the PARND-BL and of the HPARND-BL is presented. These methods are compared to SPR and FGH approaches under different network scenarios a sensitivity analysis has been performed by varying several network metrics.

As in the previous sections, PAR and PARND-BL have been solved by IpOpt and CPLEX, respectively, but a newer version of both solvers has been used:

- IBM ILOG CPLEX Optimization Studio V12.3;
- Ipopt version 3.10.0.

Furthermore, as in the previous section, all the methods (also the heuristics) have QoS features (i.e. $rho = 0.95$).

4.6.1 Simulation settings

The performance analysis has been carried out by referring to the same set of core network scenarios taken into account in the previous section. Since the choice of a traffic matrix where each node of the network is both source and destination of traffic would imply that only links (or PICs) could be powered off, the traffic demand has been modified to obtain a set of transit nodes: the 20% of nodes with the highest degree have been selected and the traffic demands from/to them have been deleted.

4.6.2 Simulation results

In this section, the performance and the behavior of the different approaches on the various topologies are analyzed, by studying the following indicators as in the previous sections:

- overall power consumption of the network;
- number of powered on network elements;
- CPU time;
- number of flow paths per origin-destination pair.

Furthermore, an extensive simulation analysis has been exploited to evaluate the behavior of the approaches under different topological and traffic conditions and QoS requirements. In particular,the parameters have been variated as follows:

Table 4.14: *Power Consumption (W) vs Topology*

	nobel-eu	Exodus	Ebone	Abovenet	Sprintlink	ta2
SPR	30974	21422	22962	18533	48325	86673
PARND-BL	11807	9864	11164	9581	23286	N/A
FGH	12149	9734	11203	9587	23389	41264
FGH-QoS	12397	9876	11469	9828	24114	41835
HPARND-BL	12364	9985	11521	9657	23585	40782

- the traffic demand by using a multiplicative factor $\alpha \in \{0.7, 0.8, \ldots, 2.0\}$;
- the number of transit nodes $\{0, 10, 20, 30, 40, 50\}\%$;
- the maximum link utilization $\rho \in \{0.70, 0.75, 0.80, 0.85, 0.90, 0.95, 1.00\}$.

Overall power consumption of the network

Table 4.14 depicts the network power consumptions of the different approaches. The table shows trends similar to those presented is previous section, thus the ability of powering off the nodes does not affect the results. The power savings of power aware approaches with respect to SPR are from the 50% to the 65%. The comparison of the different power aware approaches highlights that the optimal solution of PARND-BL obviously achieves the best performances ever w.r.t. FGH (excluding in Exodus topology), but for large topologies (see ta2 topology) a solution cannot be obtained in an acceptable time. Moreover, HPARND-BL provides a good approximation of PARND-BL, despite that for some small topologies (i.e. Exodus and Ebone) FGH-QoS is more profitable. Instead, for large topologies (i.e. Sprintlink and ta2) HPARND-BL saves further $2 \div 3\%$ w.r.t. FGH-QoS.

Number of powered on network elements

Table 4.15 shows the number of powered on network elements (i.e. nodes, links, and PICs). The table highlights that the powering off of entire nodes is not very profitable, also for this reason there are not remarkable energy savings between FGH approaches and PARND-BL approaches. In conclusion, a power aware topology is a topology with a few powered off nodes and links, but with "thin" links (i.e. high number of powered off PICs per link). Note

Table 4.15: *Number of Powered On Network Elements (Nodes/Links/PICs)*

	nobel-eu	Exodus	Ebone	Abovenet	Sprintlink	ta2
PARND-BL	26/32/48	22/32/40	22/30/48	21/28/38	41/61/102	N/A
FGH	28/38/48	22/33/39	23/30/48	22/29/38	43/65/98	65/96/164
FGH-QoS	28/39/50	22/32/40	23/32/49	22/31/40	43/67/103	65/99/172
HPARND-BL	28/40/50	22/34/41	23/34/50	21/30/39	42/65/104	65/102/185

Table 4.16: *CPU times (s)*

	nobel-eu	Exodus	Ebone	Abovenet	Sprintlink	ta2
SPR	0.01	0.01	0.01	0.01	0.06	0.26
PARND-BL	56.59	236.27	21.49	40.47	87784.79	N/A
FGH	131.21	59.53	66.12	50.82	955.49	4693.43
FGH-QoS	130.13	58.38	65.82	51.39	944.19	4412.29
HPARND-BL	165.15	72.33	91.00	60.54	1420.37	7394.30

that the power consumption due to the route processing can be notable, as shown in ta2 topology where HPARND-BL powers off less links and PICs than FGH and FGH-QoS but however the power consumption is lower.

CPU time

Table 4.16 depicts the CPU times required by the different approaches. The measurements have been carried out using a PC with 4 Intel Core i7 CPUs @ 3.07GHz (hyperthreading enabled), 8GB RAM, and an ASUS P6T DELUXE V2 Motherboard. The results highlight that the running time for the solution of the PARND-BL problem can be unacceptable in some network scenarios, such as Sprintlink or ta2, even by using the new version of CPLEX. Instead, the running times of heuristic approaches are reduced by using the new version of IpOpt, which has fixed the problem investigated in the Section 4.5.2. The CPU times of HPANB-BL are obviously longer than those of FGH approaches, but however both times are widely into the time scale.

Table 4.17: *Number of Paths per Flow (avg/max)*

	nobel-eu	Exodus	Ebone	Abovenet	Sprintlink	ta2
PARND-BL	1.04/2	1.10/3	1.08/2	1.07/2	1.04/3	N/A
FGH	1.35/6	1.44/6	1.29/5	1.24/4	1.37/7	1.40/7
FGH-QoS	1.42/6	1.49/7	1.40/7	1.28/5	1.41/7	1.43/10
HPARND-BL	1.26/4	1.53/5	1.35/4	1.22/7	1.23/5	1.34/7

Number of paths per traffic demand

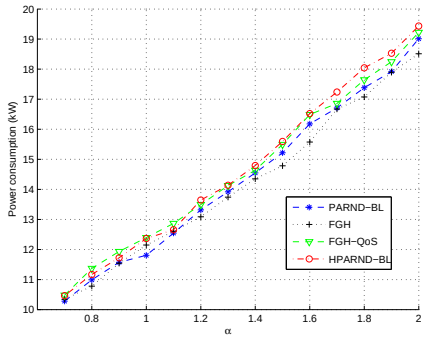
Table 4.17 analyzes the number of path per traffic demand for each approach. The table highlights that, as in the previous study, the number of paths is very low (i.e. 2 or 3) for the optimal solution of PARND-BL by means of CPLEX. Instead, for the heuristic approaches there is an increment of the number of paths (up to 10), this behavior might be due to the different traffic demand or the new version of the IpOpt. From this point of view, HPARND-BL has better performance than FGH approaches.

Impact of traffic demand

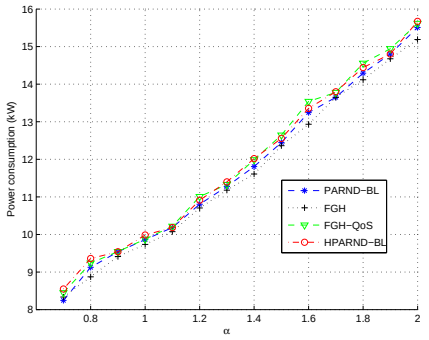
Figure 4.25 depicts the network power consumption of the different approaches by varying the traffic factor α. The figure highlights an increment of the gap among PARND-BL approaches and FGH-QoS in almost all the topologies (excluding nobel-eu and Exodus) when the traffic demand increases. For example, in Ebone topology (see Figure 4.25(c)) the power saving of HPARND-BL w.r.t. FGH-QoS, which was negative at $\alpha = 1.0$, is more than the 2% at $\alpha = 2.0$. Instead, in ta2 topology (see Figure 4.25(f)) the power savings of HPARND-BL firstly increases and successively decreases, with a power saving of about 6% at $\alpha = 1.5$. This trend is due to the ability of HPARND-BL to be aware of the power consumption of route processing, which has a cubic behavior.

Impact of number of transit nodes

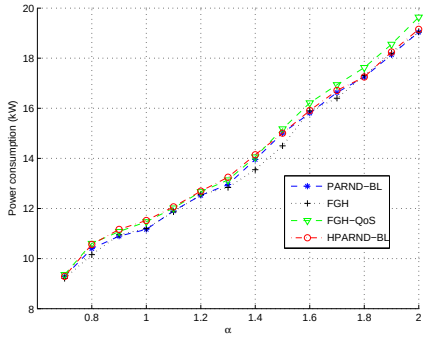
Figure 4.26 shows the behavior of the power aware approaches when the number of transit nodes increases. Obviously, there is a decreasing trend because, when the number of nodes increase, the overall traffic demand decreases (the traffic demand of transit nodes has been deleted). Varying the number of transit nodes, the performance of PARND-BL w.r.t. FGH



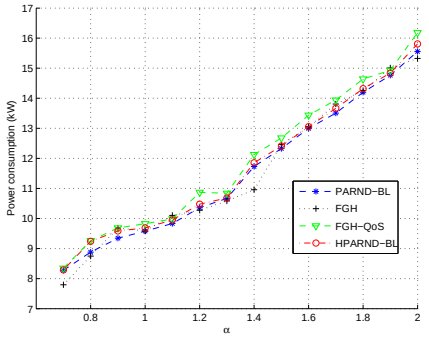
(a) nobel-eu



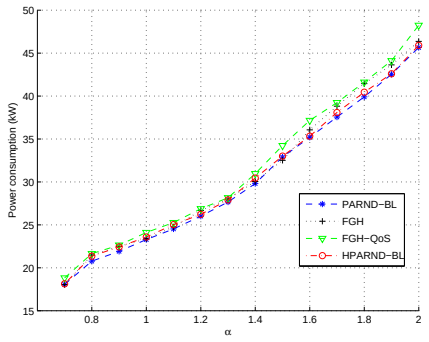
(b) Exodus



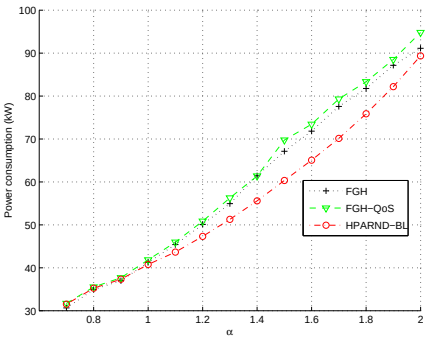
(c) Ebone



(d) Abovenet



(e) Sprintlink



(f) ta2

Figure 4.25: *Power Consumption vs load factor* α

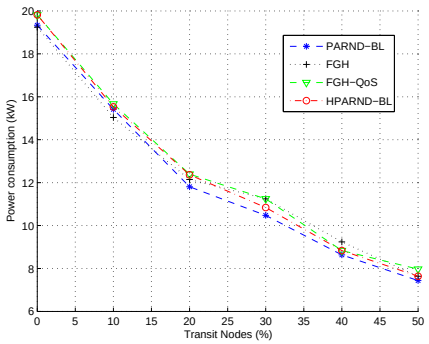
approaches remains constant. Instead, in some topologies (such as Ebone and ta2) the performance of HPARND-BL worses. For example, in the ta2 topology the power savings vary from about the 14% to 0.

Impact of the maximum link utilizations

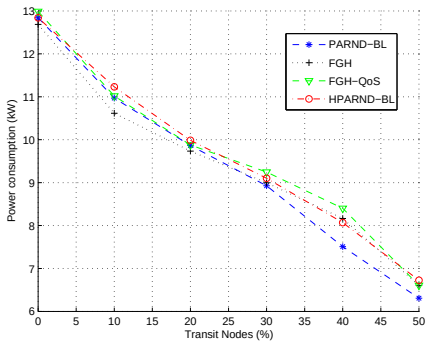
Figure 4.27 shows the network power consumption when the maximum link utilization is varied. The figure highlights a decreasing trend, when the available resources increase, greater ρ, the power consumption decreases. The power savings of PARND-BL approaches w.r.t. FGH-QoS just lightly increase when ρ increases. For example, for the ta2 topology the power savings varies from 1% to 5%.

4.7 Conclusions

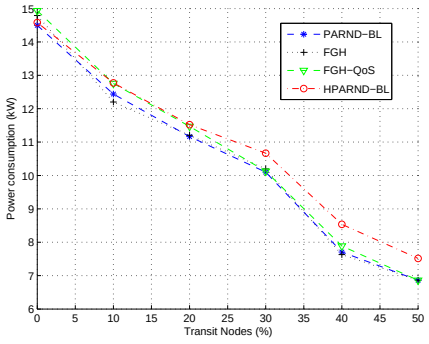
In the previous chapter, some relevant NPM problems, together with related mathematical models, have been presented. The first addressed problem has been a power aware routing problem whose purpose is to route the traffic demands in such a way as to minimize the overall power consumption of the network. This is done by assuming that all the nodes and all the PICs of the network are powered on. For this problem, called *PAR*, a nonlinear multicommodity flow model and a heuristic (called DPRA) have been proposed. The second problem, *PAND*, considers the possibility to power off nodes and links of the network in order to reduce the overall power consumption. This is done by approximating the power behavior of the nodes to the worst case, so leading to a Mixed Integer Linear Programming (MILP) design and routing model. Problem *PARND* generalizes both *PAR* and *PAND*. In fact, it addresses the possibility of powering off nodes and links of the network while defining the routing strategy, by modelling the true, nonlinear power behavior of the nodes. The corresponding model is therefore a mixed integer nonlinear programming model. A heuristic of the PARND problem has been also proposed. Finally the last problem, *PARND-BL*, represents a further level of generalization, since it allows that even single PICs of bundled links are powered off. A hierarchy of NPM models is thus proposed, having *PARND-BL* on the top, models *PAR* and *PAND* on the bottom, and *PARND* at the intermediate level. This is the first time that a nonlinear Power Aware Routing and Design model is investigated and two related heuristics have been proposed. Furthermore, model *PARND-BL* is innovative since the possibility of powering off single PICs of logical links was never investigated in the context



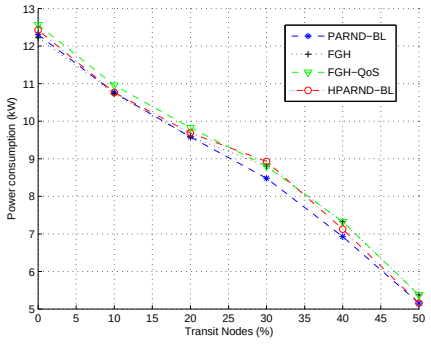
(a) nobel-eu



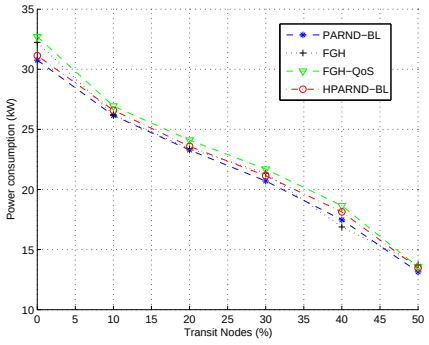
(b) Exodus



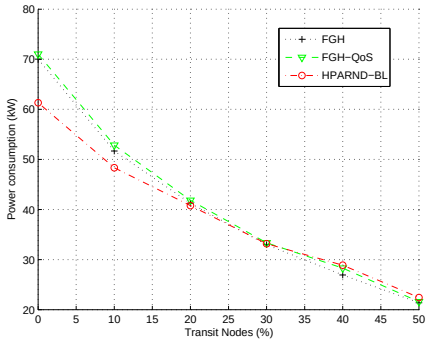
(c) Ebone



(d) Abovenet

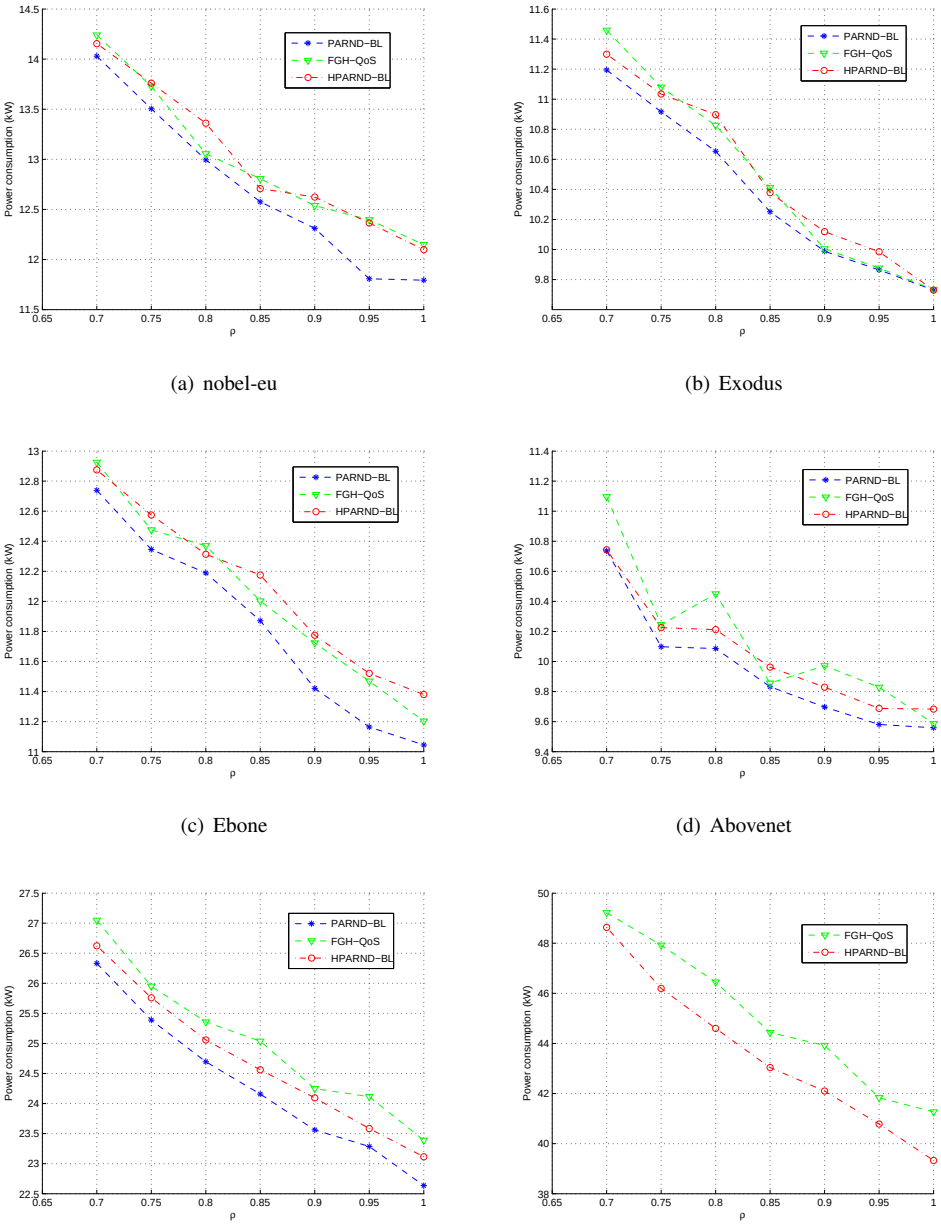


(e) Sprintlink



(f) ta2

Figure 4.26: *Power Consumption vs number of transit nodes*



(a) nobel-eu (b) Exodus

(c) Ebone (d) Abovenet

(e) Sprintlink (f) ta2

Figure 4.27: *Power Consumption vs maximum link utilization*

of power aware routing and network design. In conclusion, the *PARND-BL* and the nonlinear generalized *PARND* models as well as the heuristics of PAR, PARND and PARND-BL have to be considered as an original contribution of this thesis.

An additional contribution of this thesis are the wide computational experimentations, on several real network topologies, presented in this chapter and aimed at investigating the efficiency of the proposed power aware approaches from different perspectives.

The first simulation results shown in this chapter highlight the ability of the proposed heuristic solution of the PAR problem, DPRA, in producing power savings comparable with the exact solution. Furthermore, the PAR solution shows the loss of energy savings when the power behavior of the nodes is roughly approximated. Further, the simulation results show that, in order to achieve an adequate energy savings with respect to the SPR, the network traffic load should lead (in average) the nodes to work in a point of their power curve where the rate of change is appreciable.

The successive simulation study highlights that very high power savings are provided by the joint utilization of power-aware routing strategy and of a network design aimed at minimizing the power consumption and that the proposed heuristic for solving PARND problem is very profitable.

A further evaluation study compares the potential power savings of several approaches (PARND, SP-PARND, and PARND-BL) with respect to two versions of a naive solution (MSC and SP-MSC) in a reference network scenario. The results show that the single-path version of the approaches is more complex than the multi-paths version while the power savings are the same and that PARND can save about the 25% of the power consumed by MSC, instead PARND-BL can save over the 40%.

Afterwards, all the NPM models have been evaluated. The impact on the reduction of the overall network power consumption, the characteristics of the computed solutions, expressed in terms of powered off links and PICs, and the time required to get the optimal solutions have been deeply investigated. Furthermore, relevant Quality of Service (QoS) requisites such as the number of paths per origin-destination pair in the obtained solutions have been analyzed as well. The obtained results are very interesting, showing that the more sophisticated approaches (*PARND* and *PARND-BL*) allow one to substantially reduce the network power consumption in all the considered network scenarios. In particular, an interesting outcome of this experimentation is that, to reduce the overall network consumption, the power consumption due to route processing, when dependent on the node throughput, has to be explicitly addressed in the problem formulation, as modelled in *PARND* and in *PARND-*

BL. In the case of router hardware implementing power aware techniques, such as DVS and DFS, just to power off links/PICs does not necessarily determine a reduction of the network consumption.

Moreover, a simulation study has been presented to highlight the improvements, both in terms of power savings and CPU times, introduced by the TLPH and PMH algorithms to solve a backbone network version of PARND-BL problem. These improvements are particular relevant in the case of complex network scenario.

Finally, an extensive simulation study has been carried on to deeply investigate the behavior of PARND-BL under different network scenarios and evaluate the proposed heuristic of PARND-BL.

CHAPTER 5

CONCLUSIONS AND PERSPECTIVES

In this thesis, the topology design problem of SONs with performance requirements has been addressed taking into account both the traffic demand and the overhead. Through extensive simulations, the performance and overhead of a limited set of well-known overlay topologies and of three new traffic demand-aware overlay topologies, called KSPT, PAC, DAC have been investigated either when the IP-layer network model is flat or hierarchical and in a real network scenario.

The results presented in the Chapter 2 show that MT always performs worse than the other overlay topologies. Moreover, when the size of the flat IP-layer network is large and the number of overlay nodes is small, KSPT is a valuable option. Instead, when the IP-layer network is hierarchical, the AC topology outperforms all the other network topologies, because it takes advantage of the underlay topology knowledge. Moreover, PAC and particularly DAC are two possible alternatives to the AC topology to reduce the node degree. Finally, in a real network scenario with a large amount of available bandwidth, KMST outperforms all the other overlay topologies and MT does not underperform the other overlay topologies as in the previous cases.

In the Chapter 3, some relevant NPM problems have been presented. Each problem is based on different characterizations and power awareness of the network devices, leading either to MILP models or to MINLP models. This has been the first time that a nonlinear Power Aware Routing and Design model is investigated. Furthermore, model PARND-BL is innovative since the possibility of powering off single PICs of logical links was never investigated in the context of power aware routing and network design. In conclusion, the *PARND-BL* and the nonlinear generalized *PARND* models as well as the heuristics of PAR,

PARND and PARND-BL have to be considered as an original contribution of this thesis.

An additional contribution of this thesis are the wide computational experimentations, on several real network topologies, presented in Chapter 4 and aimed at investigating the efficiency of the proposed power aware approaches from different perspectives.

The results show that the more sophisticated approaches, PARND and PARND-BL, allow one to substantially reduce the network power consumption in all the considered network scenarios. In particular, an interesting outcome of this experimentation is that, to reduce the overall network consumption, the power consumption due to route processing, when dependent on the node throughput, has to be explicitly addressed in the problem formulation, as modelled in PARND and in PARND-BL. In the case of router hardware implementing power aware techniques, such as DVS and DFS, just to power off links/PICs does not necessarily determine a reduction of the network consumption. In addition, the solutions obtained via the proposed models satisfy the QoS requisites stated above, and usually they can be determined within a reasonable computational time. Clearly, for very-large network scenarios, PARND and PARND-BL might still require a huge computational time. To overcome this issue, the proposed heuristics are a valuable solution.

In this field, further studies can be realized by implementing the NPM methods on network simulator and/or software router to better evaluate the achievable energy saving and the QoS impact.

BIBLIOGRAPHY

[1] "BONMIN: Basic Open-source Nonlinear Mixed INteger programming," Available at https://projects.coin-or.org/Bonmin.

[2] "BRITE: Boston University Representative Internet Topology Generator," Available from: http://www.cs.bu.edu/BRITE/.

[3] "Datasheet of Juniper T Series Core Routers," Available at https://www.juniper.net/us/en/local/pdf/datasheets/1000051-en.pdf.

[4] "ECR Initiative™," Available at http://www.ecrinitiative.org/.

[5] "IBM ILOG CPLEX Optimizer," Available at http://www.ibm.com/software/integration /optimization/cplex-optimizer/.

[6] "Ipopt: Interior Point OPTimizer," Available at https://projects.coin-or.org/Ipopt.

[7] "Rocketfuel: an ISP topology mapping engine," Available at http://www.cs.washington.edu/research/networking/rocketfuel/.

[8] "Simple Network Description Library (SNDlib)," Available at http://sndlib.zib.de.

[9] "T1600 Core Router PIC Guide," Available at http://www.juniper.net/techpubs/en_US/release-independent/junos/infor mation-products/topic-collections/hardware/t-series/t1600/pics/t1600-pic.pdf.

[10] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall, Feb. 1993.

[11] Y. Amir, C. Danilov, and C. Nita-rotaru, "High performance, robust, secure and transparent overlay network service," in *Proceedings of International Workshop on Future Directions in Distributed Computing (FuDiCo)*, 2002.

[12] D. Andersen, H. Balakrishnan, F. Kaashoek, and R. Morris, "Resilient overlay networks," in *Proceedings of the eighteenth ACM symposium on Operating systems principles (SOSP '01)*, 2001, pp. 131–145.

[13] S. Antonakopoulos, S. Fortune, and L. Zhang, "Power-aware routing with rate-adaptive network elements," in *Proceedings of the 3rd International Workshop on Green Communications (GreenComm3)*, December 2010, pp. 1428–1432.

[14] D. Awduche, L. Berger, D. Gan, T. Li, V. Srinivasan, and G. Swallow, "RFC 3209: RSVP-TE: Extensions to RSVP for LSP Tunnels," *IETF*, 2001.

[15] A. Bianzino, C. Chaudet, F. Larroca, D. Rossi, and J. Rougier, "Energy-aware routing: A reality check," in *Proceedings of the 3rd International Workshop on Green Communications (GreenComm3)*, 2010, pp. 1422–1427.

[16] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, "An Architecture for Differentiated Service," RFC 2475 (Informational), Dec. 1998, updated by RFC 3260.

[17] R. Bolla, R. Bruschi, F. Davoli, and F. Cucchietti, "Energy Efficiency in the Future Internet: A Survey of Existing Approaches and Trends in Energy-Aware Fixed Network Infrastructures," *IEEE Communications Surveys and Tutorials (COMST)*, vol. 13, no. 2, pp. 223–244, May 2011.

[18] R. Braden, D. Clark, and S. Shenker, "Integrated Services in the Internet Architecture: an Overview," RFC 1633 (Informational), June 1994.

[19] A. Capone, J. Elias, and F. Martignon, "Optimal Design of Service Overlay Networks," in *Proceedings of the Fourth International Workshop on QoS in Multiservice IP Networks*, 2008.

[20] J. Chabarek, J. Sommers, P. Barford, C. Estan, D. Tsiang, and S. Wright, "Power awareness in network design and routing," in *Proceedings of the 27th IEEE Conference on Computer Communications (INFOCOM 2008)*, 2008, pp. 457–465.

[21] L. Chiaraviglio, M. Mellia, and F. Neri, "Minimizing ISP Network Energy Cost: Formulation and Solutions," *IEEE/ACM Transactions on Networking*, vol. PP, no. 99, pp. 1–14, July 2011.

[22] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, pp. 269–271, 1959.

[23] R. Doverspike, K. Ramakrishnan, and C. Chase, "Structural overview of ISP networks," *Guide to Reliable Internet Services and Applications*, pp. 19–93, 2010.

[24] Z. Duan, Z.-L. Zhang, and Y. T. Hou, "Service Overlay Networks: SLAs, QoS and Bandwidth Provisioning," in *Proceedings of the 10th IEEE International Conference on Network Protocols (ICNP '02)*, 2002, pp. 334–343.

[25] European Commission DG INFSO, "Impacts of Information and Communication Technologies on Energy Efficiency," Sept. 2008.

[26] J. Fan and M. H. Ammar, "Dynamic Topology Configuration in Service Overlay Networks: A Study of Reconfiguration Policies," in *Proceedings of the 25th IEEE Conference on Computer Communications (INFOCOM'06)*, 2006.

[27] W. Fisher, M. Suchara, and J. Rexford, "Greening Backbone Networks: Reducing Energy Consumption by Shutting Off Cables in Bundled Links," in *Proceedings of the first ACM SIGCOMM workshop on Green Networking*, August 2010, pp. 29–34.

[28] R. G. Garroppo, S. Giordano, G. Nencioni, and M. Pagano, "Energy Aware Routing based on Energy Characterization of Devices: Solutions and Analysis," in *Proceedings of the 4th International Workshop on Green Communications (GreenComm4)*, June 2011, pp. 1–5.

[29] I. Ghamlouche, T. G. Crainic, and M. Gendreau, "Cycle-Based Neighbourhoods for Fixed-Charge Capacitated Multicommodity Network Design," *Operations Research*, vol. 51, no. 4, pp. 655–667, 2003.

[30] Global e-Sustainibility Initiative (GeSI), "Smart 2020: Enabling the low carbon economy in the information age," June 2008.

[31] X. Gu, K. Nahrstedt, R. N. Chang, and C. Ward, "QoS-assured service composition in managed service overlay networks," in *Proceedings of IEEE 23rd International Conference on Distributed Computing Systems*, 2003.

[32] C. Gunaratne, K. Christensen, B. Nordman, and S. Suen, "Reducing the Energy Consumption of Ethernet with Adaptive Link Rate (ALR)," *IEEE Transactions on Computers*, vol. 57, no. 4, pp. 448–461, 2008.

[33] M. Gupta and S. Singh, "Greening of the internet," in *Proceedings of the ACM conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM 2003)*, 2003, pp. 19–26.

[34] R. N. H. Ikebe, N. Yamashita, "Green Energy for Telecommunications," in *Proceedings of the 29th International Telecommunications Energy Conference (INTELEC 2007)*, Sept. 2007, pp. 750–755.

[35] J. Han, D. Watson, and F. Jahanian, "Topology aware overlay networks," vol. 4, 2005, pp. 2554–2565.

[36] R. Hays, "Active/Idle Toggling with Low-Power Idle," in *IEEE 802.3az Task Force Group Meeting*, January 2008, Available at http://www.ieee802.org/3/az/public/jan08/hays_01_0108.pdf.

[37] F. Idzikowski, "Power consumption of network elements in IP over WDM networks," in *TKN Technical Report TKN-09-006*, July 2009.

[38] IEEE Computer Society, "IEEE Standard 802.1AX: Link Aggregation," 2008.

[39] M. Kamel, C. Scoglio, and T. Easton, "Optimal Topology Design for Overlay Networks," 2007, pp. 714–725.

[40] L. Lao, S. S. Gokhale, and J. hong Cui, "Distributed QoS Routing for Backbone Overlay Networks," *Lectures Notes in Computer Science*, vol. 3976/2006, pp. 1014–1025, 2006.

[41] Z. Li and P. Mohapatra, "QRON: QoS-aware routing in overlay networks," *IEEE Jornal on Selected Areas in Communications*, vol. 22, pp. 29–40, 2004.

[42] ——, "On investigating overlay service topologies," *Computer Networks*, vol. 51, pp. 54–68, 2007.

[43] B. D. McBride and C. Scoglio, "Characterizing Traffic Demand Aware Overlay Routing Network Topologies," in *Proceedings of the 26th IEEE Conference on Computer Communications (INFOCOM07)*, 2007.

[44] A. Nakao, L. Peterson, and A. Bavier, "A routing underlay for overlay networks," in *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM'03)*, 2003, pp. 11–18.

[45] J. Restrepo, C. Gruber, and C. Machuca, "Energy Profile Aware Routing," in *Proceedings of the First International Workshop on Green Communications (GreenComm)*, June 2009, pp. 1–5.

[46] E. Rosen, A. Viswanathan, and R. Callon, "Multiprotocol Label Switching Architecture," *IETF*, 2001.

[47] S. N. Roy, "Energy Logic: A Road Map to Reducing Energy Consumption in Telecommunications Networks," in *Proceedings of the 30th International Telecommunications Energy Conference (INTELEC 2008)*, Sept. 2008, pp. 1–9.

[48] L. Subramanian, I. Stoica, H. Balakrishnan, and R. H. Katz, "OverQos: an overlay based architecture for enhancing internet Qos," in *Proceedings of the 1st conference on Symposium on Networked Systems Design and Implementation (NSDI'04)*, 2004, p. 6.

[49] H. T. Tran and T. Ziegler, "A design framework towards the profitable operation of service overlay networks," *Computer Networks*, vol. 51, no. 1, pp. 94–113, 2007.

[50] J. Truffot, C. Duhamel, and P. Mahey, "k-splittable delay constrained routing problem: A branch-and-price approach," *Networks*, vol. 55, no. 1, pp. 33–45, 2010.

[51] B. Vatinlen, F. Chauvet, P. Chretienne, and P. Mahey, "Simple bounds and greedy algorithms for decomposing a flow into a minimal set of paths," *European Journal of Operational Research*, vol. 185, no. 3, pp. 1390–1401, 2008.

[52] S. L. Vieira, "Topology Design for Service Overlay Networks with Bandwidth Guarantees," in *Proceedings of the IEEE International Workshop on quality of Service (IWQoS)*, 2004, pp. 211–220.

[53] B. M. Waxman, "Routing of multipoint connections," *IEEE Journal on Selected Aereas in Communications*, vol. 6, no. 9, pp. 1617–1622.

[54] A. Young, J. Chen, Z. Ma, and A. Krishnamurthy, "Overlay mesh construction using interleaved spanning trees," in *Proceedings of the 23th IEEE Conference on Computer Communications (INFOCOM'04)*, 2004, pp. 396–407.

[55] S. Zeadally, S. Khan, and N. Chilamkurti, "Energy-efficient networking: past, present, and future," *The Journal of Supercomputing*, pp. 1–26, May 2011.

[56] B. Zhai, D. Blaauw, D. Sylvester, and K. Flautner, "Theoretical and practical limits of dynamic voltage scaling," in *Proceedings of the 41st Annual Design Automation Conference (DAC 2004)*, 2004, pp. 868–873.

[57] L. Zhou and A. Sen, "Topology Design of Service Overlay Network with a Generalized Cost Model," in *Global Telecommunications Conference, 2007. GLOBECOM '07. IEEE*, 2007, pp. 75–80.