Energy-Aware IP Traffic Engineering with Shortest Path Routing

E. Amaldi[a], A. Capone[a], L.G. Gianoli[a,b,*]

[a] *Politecnico di Milano, Dipartimento di Elettronica e Informazione Italy*
[b] *École Polytechnique de Montréal, Département de génie électrique, Canada*

Abstract

Internet energy consumption is rapidly becoming an issue due to the exponential traffic growth and the rapid expansion of communication infrastructures worldwide. We address the problem of energy-aware intra-domain traffic engineering in networks operated with a shortest path routing protocol. We consider the problem of switching off (putting in sleep mode) network elements (links and routers) and of adjusting the link weights so as to minimize the energy consumption as well as a network congestion measure. To tackle this multi-objective optimization problem with priority (first minimize the energy consumption and then the congestion), we propose a Mixed Integer Linear Programming based algorithm for Energy-aware Weights Optimization (MILP-EWO). Our heuristic exploits the Interior Gateway Protocol Weight Optimization (IGP-WO) algorithm for optimizing the OSPF link weights so as to minimize the network congestion. The computational results obtained for seven real network topologies and different types of traffic matrices show that it is possible to switch off a substantial number of nodes and links during low and moderate traffic periods, while guaranteeing that network congestion is low enough to ensure service quality. For comparison purposes, we also report the results obtained with (randomized) greedy procedures. The proposed approach is also validated on two networks of emulated Linux routers.

Keywords: Energy-Aware, Traffic Engineering, Shortest Path, OSPF, Weights Optimization

1. Introduction

In the last years, the exponential growth of Internet has been accompanied by a considerable increase in its energy consumption. Data reported in [1] show that in 2007 Internet has been responsible for 5.5% of the total energy consumption in the world and that the annual increment rate can be estimated around 20-25%. Other estimates attribute to the ICT sector a comprehensive energy consumption equal to 156 GW in 2007, with 22 GW consumed exclusively by network equipments (excluding servers in data centers) [2]. Since the annual growth rate is evaluated around 12%, consumptions may reach 95 GW in 2020 [2]. Detailed data reported in [3] about the annual consumptions of very important Internet Service Providers (ISP), show that in 2009 the annual power consumptions of the biggest ISPs overcame the 10 GWh per year. Always in [3] it is also shown that in large-scale and medium size ISPs like Telecom Italia and GRNET the total energy consumptions of access and core networks should exceed 400 GWh per year in 2015. For these reasons, the issues of energy saving in IP networks and of power awareness in network design have recently become of great interest in the scientific community and have attracted the attention of device manufacturers and ISPs [4].

Since the seminal work by Gupta and Singh [5], the research community has started developing technologies for manufacturing energy efficient network devices, methodologies for power aware network design, as well as energy management strategies for reducing energy wastes of networks in operation [6, 7]. In this paper we focus on energy management of the network and its elements.

Energy management in the Internet exploits the fact that networks are designed and dimensioned to serve the estimated peak traffic demand. During network operation, traffic load varies remarkably over time and even during peak hours it is usually well below network capacity [8]. Unfortunately, current network device architectures and transmission technologies make their power consumption almost independent of the traffic load [9, 10], with a static power consumption (measured at zero load) of the same order of the peak power (measured at full load).

A natural approach to improve the Internet energy performance is to switch off or put to sleep mode the unnecessary network devices (routers and links) so as to adapt the network consumption to the traffic load, while guaranteeing that traffic demands can still be routed in the remaining sub-network. The advantages of this approach with respect to those based on load balancing are discussed in [11]. Coordinated energy management strategies can clearly achieve a better performance than those dealing with individual network devices [12].

The routing protocol plays a key role in the energy management and it affects possible strategies. We consider the most widely used Internal Gateway Protocol (IGP) in IP networks, namely the Open Shortest Path First (OSPF) protocol. The link state approach adopted by OSPF to distribute topology information and to perform local calculation of shortest paths is based

*Corresponding author
Email addresses: amaldi@elet.polimi.it (E. Amaldi), capone@elet.polimi.it (A. Capone), gianoli@elet.polimi.it (L.G. Gianoli)

on a set of link weights. This prevents traffic engineering strategies to optimize routing on a per-flow basis (like with MPLS), operation that can be practically very complex when an high number of traffic demands is considered (a dedicated path has to be explicitly assigned to each demand). Therefore the most popular approaches to traffic engineering in OSPF optimize the set of shortest paths by adjusting the link weights (see the seminal work [13] and the surveys [14, 15]).

Current routing devices can be switched in sleep and active mode in relatively short time [16] and most of the manufacturers are currently working to improve the efficiency of devices low energy modes and make their management more agile [4]. Network management platforms commonly adopted by service providers allow to manage remotely both the activation status and the link weights based on energy management policies and expected traffic scenarios, and to gather traffic statistic through monitoring agents.

In this paper we propose a novel approach for off-line intra-domain energy-aware network management based on the optimization of the OSPF link weights: given the daily time periods characterized by different traffic levels (night, morning, lunch break, afternoon), and for which origin destination traffic matrices can be quite accurately forecast [17], we efficiently configure the link weights so as to minimize both energy consumption and network congestion, while guaranteeing that all the traffic demands are routed respecting the QoS constraints[1]. In particular, we present a heuristic for energy-aware OSPF link weights optimization, the *Mixed Integer Linear Programming based algorithm for Energy-aware Weights Optimization* (MILP-EWO). Our method exploits the Interior Gateway Protocol Weight Optimization (IGP-WO) algorithm (proposed in [13] and available from [20]). More in details, IGP-WO is a tabu-search algorithm that, given a traffic matrix, and an IP networks operated with the OSPF protocol with equal cost multi-path (ECMP) (traffic demands equally split among the all the shortest paths), aims at iteratively modifying the OSPF weights so as to minimize a piecewise linear convex cost function related to the link utilization (see Figure 1).

The paper is organized as follows. After summarizing the related work in Section 2, we present in Section 3 our approach for achieving energy-efficient shortest path routing. In Section 4 we describe the multi-objective energy-aware traffic engineering problem under consideration and give a mathematical programming formulation. Since this formulation is very challenging computationally even for small-size instances, in Section 5 we propose our MILP-based heuristic. In Section 6 we report and discuss the computational results obtained on seven real network topologies (both backbone and access network topologies) with different traffic matrices. For comparison purposes, we also report results obtained with randomized greedy procedures. In Section 7 we summarize some experimental evaluations obtained via network emulation, that support the validity and the applicability of our approach and algorithm. Finally, Section 8 contains some concluding remarks.

[1]Preliminary versions of part of this work have been presented in [18, 19]

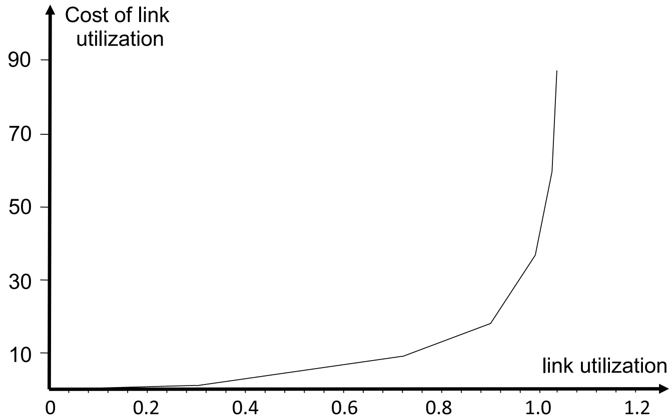


Figure 1: Example of piecewise linear convex link utilization cost function.

2. Related Work on Energy-Aware Traffic Engineering

Although the field of green networking has been attracting a growing attention during the last years [21, 22], a limited number of recent works have been devoted to energy-aware traffic engineering and network design.

Some preliminary studies evaluate the potentialities and the effective applicability of energy-aware routing procedures [23, 24, 25, 26, 11]. Others investigate models of router power consumption in order to define network design strategies that can guarantee at the same time low energy consumption, good performance and resilience to failures [9, 27].

The approach proposed in [28] aims at switching off the line-cards (network links) guaranteeing QoS constraints (maximum utilization and maximum path length constraints) in a scenario where an hybrid MPLS/OSPF scheme is adopted. The approach is based on a MILP formulation where the traffic demands are routed through a set of previously computed k-shortest paths.

In [29] the authors describe some heuristics that, given a traffic matrix and a fully powered network, are able to switch off nodes and links while respecting traffic constraints. In [30] some on-line Energy-Aware Traffic Engineering (EATe) techniques are presented for optimizing links and routers power consumption, by considering their rate-dependant energy profiles. Assuming that the energy consumption of the network elements depends on the different rates, EATe algorithms try to switch off the underutilized elements or to reduce their rate, by re-routing part of the traffic in other network portions without increasing the rate of any element. Unlike in our work, the approaches in [29] and [30] follow a flow-based strategy that is suitable for the Internet geographic backbone where label switching protocols are adopted and not for routing domains based on shortest paths. Other works on energy-aware TE with flow-based routing have been recently presented in [31, 32, 33, 34, 35].

The energy management algorithm for IP networks called Energy Aware Routing (EAR) algorithm and presented in [36] is able to switch off network elements exploiting a modified version of the OSPF protocol. EAR algorithm selects a subset of routers, Importers Router (IR), that do not calculate their own shortest path tree (SPT) but use that of some neighbouring routers, Exporters Routers (ER). In general a small number of active SPTs reduces also the number of links used that can be switched off. There are several important differences between our work and [36]. As in the literature on OSPF traffic engineering [37], we keep the OSPF protocol unchanged and focus on optimizing the link weights, while in the EAR algorithm the weights are assumed to be given and the protocol needs to be modified to implement ERs and IRs. Moreover, we explicitly consider link capacities and minimize the network congestion level in order to guarantee service quality, while in [36] neither traffic load nor network capacity are explicitly considered.

During the past decade considerable attention has been devoted to general intra-domain traffic engineering via link weight optimization, see [14, 15, 38] for surveys. Different objective functions have been considered (e.g., the total cost of link utilization minimization [13], residual capacity maximization and load balancing maximization) and different heuristic methods have been developed (e.g., local search, genetic algorithm, Lagrangian approach). The code of the well-known IGP-WO algorithm [13] is available from the TOTEM toolbox [20].

To the best of our knowledge, [39] is the only work dealing the issue of energy-aware link weights optimization. The algorithms presented in [39] perform a link weights optimization that aims at minimizing the energy consumption of IP networks operated with shortest path routing protocol, by switching off unnecessary links. These heuristics consist into the combination of a local search procedure and a Lagrangian relaxation of an Integer Linear Programming (ILP) formulation. Unlike in our work, it is not possible to switch off (put in a sleep mode) nodes, congestion optimization is not considered and the networks used for testing are quite small (the largest one only contains 28 nodes and 90 links).

Finally, recent contributions on energy-aware network management that adopt different perspectives include: methods for switching off links solely based on network topology features (traffic demands are ignored) [40, 41], distributed algorithms to determine the operating configuration of each node so as to minimize energy consumption [42, 43], an interesting framework for energy-aware admission control and routing [44], MILP models for the joint optimization of network installation and network energetic costs [45], a procedure to optimize both energy consumptions and congestion in networks operated with Carrier Grade Ethernet [46], and new energy-aware protocols [47, 48].

3. Energy Efficient Shortest Path Routing via link weights optimization

In this work we advocate the implementation of the energy-aware management of IP networks operated with the OSPF protocol, by only looking for an efficient configuration of the link weights. Indeed, link weight optimization allows to reduce the network congestion as well as to switch off (put in a sleep mode) network elements. Since a network element can be turned off when no traffic is flowing through it, a link can be switched off by setting its weight to a very large value so as to exclude it from the shortest path trees from all the sources. Similarly, routers can be switched off by sufficiently increasing the weights of all its incident links. Thus we assume that unused network devices (line cards and entire devices) are able to enter in a sleep mode [49], either through centralized management procedures or autonomous mechanisms based on traffic monitoring.

In our scenario we consider the OSPF protocol with the *Equal Cost Multi Paths* (ECMP) rule, where traffic demands are evenly split at nodes where more outgoing links belong to shortest paths to the destination. This feature plays a key role in terms of both link utilization and energy consumption.

As illustrated in Figure 2, we subdivide the day into a few macro-periods characterized by a certain level of traffic (e.g. morning, lunch time, afternoon, evening, night) and we compute off-line a set of energy-aware OSPF link weights for each macro-period.

The link weights are determined by taking into account the traffic forecasts and the constraints on the fraction of the link capacities that can be used. Note that, in normal conditions, traffic profiles (traffic matrices) can be predicted with good accuracy and allow planning network resources allocation in advance with limited uncertainty margins [17]. The choice of the macro periods has to be clearly done by taking into account the shape of the daily traffic profiles. By selecting a suitable value of the maximum link utilization parameter α, with $0 \leq \alpha \leq 1$, the resulting network is likely to support the unpredictable variations of traffic, also when part of the devices are in sleeping mode.

Note that link weights optimization does not interfere with other commonly used network procedures such as, for instance, those related to failure reaction, since it does not modify basic protocol operation.

We do not consider on-line optimization of the link weights for three main reasons: i) frequent weight updates would cause network instability (due to the increase in routing table updates and traffic overhead) , ii) it would require a great computational power and iii) almost optimal energy savings can be obtained by simply applying few optimized network configurations along an entire day [50]. Our main assumption is that network operators consider a limited set of reference traffic scenarios (corresponding for example to different times of the day) that can be selected and defined based on their tools for traffic measurement and analysis and that they optimize the network operation based on the identified scenarios through their network management platform. Note that this approach can be combined with online tools that can react to unexpected traffic situations reactivating sleeping links and devices. Finally, we want to emphasize that since our energy management approach is based on the optimization of link weights used for shortest path calculation, we retain the abilities of the routing protocol to adapt to traffic variations on short time scale and to react to network changes like in traditional approaches based on fixed routing metrics.

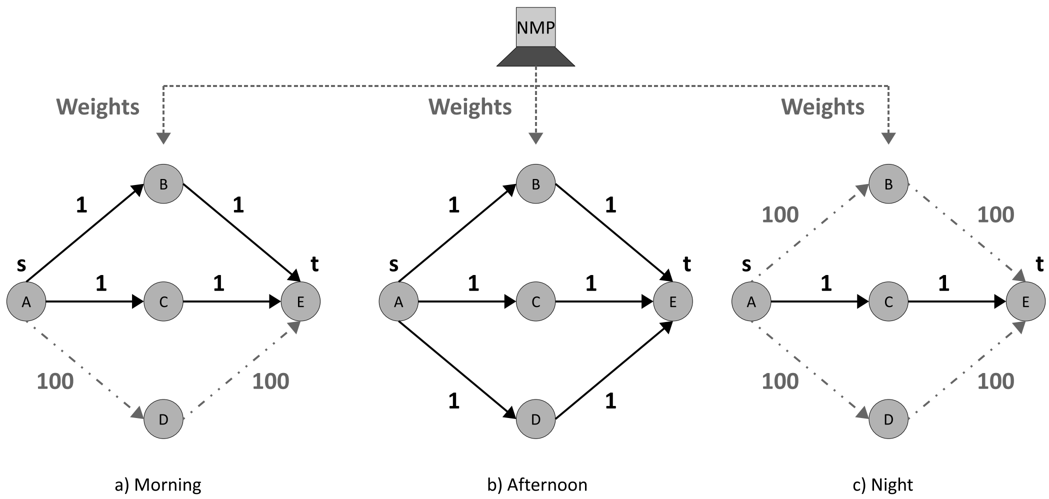


Figure 2: Energy-aware traffic engineering with shortest path routing. Link weights are changed by the network management platform (NMP) according to the traffic level of the different periods of the day. Network elements can be put in sleep mode if excluded from the shortest path trees. With the three weights configuration, we need respectively two (morning), three (afternoon) and one (night) paths to satisfy the demand d_{st}. The weights of the switched-off elements are assigned very high values.

4. MILP Model

Let the directed graph $G = (V, A)$ represent the network topology, where V is the set of nodes and A the set of links. We distinguish two types of nodes: *edge* nodes and *core* nodes. Edge nodes can be both source and destination of traffic demands, while core nodes play only the role of transit routers. Let D denote the traffic matrix, where d_{ij} is the traffic demand for each pair of edge nodes i and j, and $d_{ij} = 0$ for all other pairs of nodes. Let α, with $0 \leq \alpha \leq 1$, be the maximum allowed link utilization fraction. We consider the following extension of the IGP weight optimization problem with ECMP rule for intra-domain Traffic Engineering (see [38]) that we refer to as *Energy-aware Traffic Engineering with Shortest Path routing* (E-TESP).

E-TESP: Given a directed graph $G = (V, A)$, representing the topology of an IP network composed by routers (chassis) and links (line cards) with capacities on the links, and a traffic matrix D, decide which network elements (routers and links) to switch off and determine the link weights so as to minimize the total network energy consumption (primary objective) and a measure of effectiveness of the routing weight configuration (secondary objective), while guaranteeing that all the traffic demands are routed and the maximum utilization fraction of each link is satisfied.

Let in addition p_{ij} and p_k be the power consumption of link (i, j) and node k respectively and let αc_{ij} be the maximum available capacity of the link (i, j) with $0 \leq \alpha \leq 1$. If the binary decision variables x_{ij} and y_k represent the power status (on/off) of links and routers respectively, a Mixed Integer Linear Programming (MILP) formulation of of the E-TESP problem aiming at minimizing the total energy consumption is given by:

$$\min \sum_{(i,j)\in A} p_{ij}\, x_{ij} + \sum_{k\in V} p_k\, y_k \tag{1}$$

s. t.

$$x_{ij} \leq y_i \quad (i, j) \in A \tag{2}$$

$$x_{ij} \leq y_j \quad (i, j) \in A \tag{3}$$

$$\sum_{i\in V} f^t_{it} = \sum_{s\in V} d_{st} \quad t \in V \tag{4}$$

$$\sum_{j\in V} f^t_{vj} - \sum_{i\in V} f^t_{iv} = d_{vt} \quad v, t \in V, t \neq v \tag{5}$$

$$\sum_{t\in V} f^t_{ij} \leq x_{ij}\alpha c_{ij} \quad (i, j) \in A \tag{6}$$

$$0 \leq z^t_i - f^t_{ij} \leq (1 - u^t_{ij}) \sum_{v\in V} d_{vt} \quad t \in V, (i, j) \in A \tag{7}$$

$$f^t_{ij} \leq u^t_{ij} \sum_{v\in V} d_{vt} \quad t \in V, (i, j) \in A \tag{8}$$

$$0 \leq r^t_j + \omega_{ij} - r^t_i \leq (1 - u^t_{ij})M \quad t \in V, (i, j) \in A \tag{9}$$

$$1 - u^t_{ij} \leq r^t_j + \omega_{ij} - r^t_i \quad t \in V, (i, j) \in A \tag{10}$$

$$u^t_{ij} \leq x_{ij} \quad t \in V, (i, j) \in A \tag{11}$$

$$\omega_{ij} \geq (1 - x_{ij})\omega_{max} \tag{12}$$

$$1 \leq \omega_{ij} \leq \omega_{max} \quad (i, j) \in A \tag{13}$$

$$u^t_{ij} \in \{0, 1\} \quad t \in V, (i, j) \in A \tag{14}$$

$$x_{ij}, y_k \in \{0, 1\} \quad (i, j) \in A,\ k \in V \tag{15}$$

$$f^t_{ij} \geq 0 \quad (i, j) \in A,\ t \in V \tag{16}$$

$$r^t_i,\ z^t_i \geq 0 \quad i, t \in V, \tag{17}$$

where M is a large enough constant. The objective function (1) aims at minimizing the total energy consumption of the network. Constraints (2)-(3) ensure that if a node is switched off all incident links are turned off. Obviously a node can be switched off only if there are no traffic demands having it as source or destination. Constraints (4)-(5) are the classical flow conservation constraints, where the (real) positive variable f_{ij}^t indicates the amount of flow routed through the link $(i, j) \in A$ and destined to node $t \in V$. Constraints (6) are the maximum utilization constraints imposing that the total flow through each link does not exceed the maximum link utilization fraction α, with $0 \leq \alpha \leq 1$, and forcing the flow to 0 if the link (i, j) is powered off. The binary variables u_{ij}^t appearing in Constraints (7)-(9) describe the routing configuration: $u_{ij}^t = 1$ if and only if the link (i, j) belongs to one of the shortest paths from node i to node t. Constraints (7) make sure that if $u_{ij}^t = 1$ then the flow f_{ij}^t destined to node t is equal to the (real) variable z_i^t, which is the common value of the flow assigned to all links outgoing from i and belonging to the shortest paths from i to t. Constraints (8) force $f_{ij}^t = 0$ for all links (i, j) that do not belong to a shortest path to node t. Finally, the shortest path routing Constraints (9)-(11) assure that the routing vector u defines shortest paths consistent with the link weight vector ω and forbid switched-off links to belong to a shortest path; moreover the switched-off links weights are put equal to the maximum value w_{max}. For each pair of nodes j and t, the (real) variable r_j^t corresponds to the length of the shortest path from node j to node t.

Unfortunately, the above MILP formulation, which is an extension of that given in [38] for intra-domain traffic engineering, turns out to be very challenging even for small size networks. We tested the E-TESP formulation with six small networks (about 10 nodes and 40 links each) described in [51]. The computational results reported in Table 1 show that the MILP model is competitive only when the traffic load is very low (instances 1% and 10%) and the final solution is a simple tree. For instances with higher traffic (instances 30%), the state-of-the-art solver CPLEX 12.2 running on a Intel i7 processors with 4 core and multi-thread 8x, equipped with 8Gb of RAM, was not able to find a non trivial solution in one day of computation. By trivial solution we mean a network where all elements are active and all weights are equal to 1.

Instance	V-A-D	Non trivial	Optimum	t (s)
abilene-1%	12-15-132	yes	yes	0.6
abilene-10%	12-15-132	yes	yes	0.6
abilene-30%	12-15-132	yes	yes	339.5
dfn-bwin-1%	10-45-90	yes	yes	182.1
dfn-bwin-10%	10-45-90	yes	yes	257838.0
dfn-bwin-30%	10-45-90	no	no	648927.0
dfn-gwin-1%	11-47-110	yes	yes	348.4
dfn-gwin-10%	11-47-110	yes	yes	591561.0
dfn-gwin-30%	11-47-110	no	no	848884.0
di-yuan-1%	11-42-22	yes	yes	2551.5
di-yuan-10%	11-42-22	yes	yes	1867.1
di-yuan-30%	11-42-22	no	no	73835.9*
pdh-1%	11-34-24	yes	yes	11.1
pdh-10%	11-34-24	yes	yes	1630.6
pdh-30%	11-34-24	yes	no	25137.1*
polska-1%	12-18-66	yes	yes	6.8
polska-10%	12-18-66	yes	yes	16.9
polska-30%	12-18-66	yes	yes	29348.5

* Out of memory

Table 1: Computational results obtained solving the E-TESP formulation on six different network topologies. A solution is non-trivial when not all weights are equal to 1 and all network elements are active.

5. Energy-aware Traffic Engineering MILP-based Heuristic

Before presenting our MILP-based heuristic for energy-aware weight optimization, MILP-EWO, we briefly describe the simple greedy procedure used in the pre-processing and post-processing stages.

5.1. Greedy procedure

Consider a given network topology with link capacities, a traffic matrix D and a maximum link utilization parameter α, with $0 \leq \alpha \leq 1$. Suppose we are also given a set of link weights which allow routing of all demands along shortest paths while respecting the maximum utilization constraints for all the links. Such OSPF link weights can, for instance, be obtained by applying IGP-WO [13]. To speed up the algorithm and improve the quality of the solutions found in a given amount of time, we follow the idea proposed in [52], warm-starting IGP-WO by taking as initial link weights the values of the dual variables obtained by solving the following linear programming formulation, that minimizes the link utilization cost function used by IGP-WO while considering a fully splittable per-flow routing:

$$\min \quad \Phi = \sum_{(i,j)\in A} \phi\left(c_{ij}, l_{ij}\right) \tag{18}$$

$$s.\ t.$$

$$\sum_{j\in V} f_{vj}^t - \sum_{j\in V} f_{jv}^t = \begin{cases} -\sum_{s\in V} d_{st} & if \quad v = t \\ d_{vt} & if \quad v \neq t \end{cases} \quad v, t \in V \tag{19}$$

$$l_{ij} = \sum_{t\in V} f_{ij}^t \quad (i, j) \in A \tag{20}$$

$$\phi\left(c_{ij}, l_{ij}\right) \geq \alpha_z l_{ij} - \beta_z c_{ij} \quad (i, j) \in A, z \in Z \tag{21}$$

$$f_{ij}^t \geq 0 \quad (i, j) \in A, t \in V. \tag{22}$$

Obj. function (18) is the same as in the IGP-WO algorithm [13]. Eq. (19) represents the flow conservation constraints, while Eq. (20) is used to compute the total flow on each link. Finally, Eq. (21) defines the link utilization cost function and Constraints (22) define the domain of the flow variables. Note that the optimized link weights are obtained by the dual variables corresponding to Constraints (20).

The greedy procedure proceeds as follows. First we sort the network elements according to some intuitive criteria, then we consider them in that order and try to switch off as many of them as possible. The nodes are considered before the links because of their higher energy consumption.

We use three criteria for sorting nodes. In Least-Link (LL), Least-Flow (LF), and Sum-of-Weights (SW), nodes are sorted in non-decreasing order according to, respectively, the degree (number of incident links), the total amount of traffic flowing through them, and the sum of the weights of all the incident (active) links. We consider two criteria for sorting links. In Least-Flow (LF) and Traffic-Engineering (TE), links are sorted in non-decreasing order according to, respectively, the total amount of traffic flowing through them and their weight. Two of these criteria, Least-Link (LL) and Least-Flow (LF), were already used in a different context in [53]. The six combined node-link sorting policies are summarized in Table 2.

At each step of the greedy procedure, we verify whether the next available active network element according to the sorting order can be turned off. The considered element is actually turned off if the OSPF routing determined on the reduced network by the link weights of the active links, is able to support the traffic matrix (all traffic demands) without exceeding the link maximum utilization limit α. A run terminates when all the network elements have been tested.

	LF	**LL**	**SW**
LF	LF-LF	LL-LF	SW-LF
TE	LF-TE	LL-TE	SW-TE

Table 2: Combinations of sorting criteria for the greedy procedure; the rows correspond to the link criteria and the columns to the router criteria.

The greedy procedure is run six times (once for each combined node-link sorting policy) and the best solution found is returned so as to reduce the total cost of link utilization.

5.2. The MILP-based algorithm for Energy-aware Weights Optimization (MILP-EWO)

Since the energy-aware traffic engineering with shortest path routing problem (E-TESP) is very challenging, we split it into three stages: i) a pre-processing stage , ii) a MILP-based stage aiming at finding a minimum energy sub-network and iii) a post-processing stage (see the flow chart in Figure 3).

The purpose of the pre-processing stage is to rapidly identify which routers and links may be immediately switched off based on simple and local considerations. To find an initial set of link weights W_0, we first run the IGP-WO algorithm by Fortz and Thorup [13] aiming at minimizing the total cost of link utilization. Since we wish to switch off only part of the network elements so as to speed up the following stage without affecting performance, we then apply the greedy procedure with a maximum link utilization fraction equal to $\alpha - 0.1$. Let G' denote the reduced network topology resulting from the pre-processing stage.

The MILP-based stage includes a *Switching Off* step and a *Feasible Routing* step.

In the *Switching Off* step, given the reduced network topology G' and the traffic matrix D, we have to decide which other network elements to switch off so as to further decrease the total energy consumption while routing all the demands and respecting all maximum link utilization constraints. This is achieved

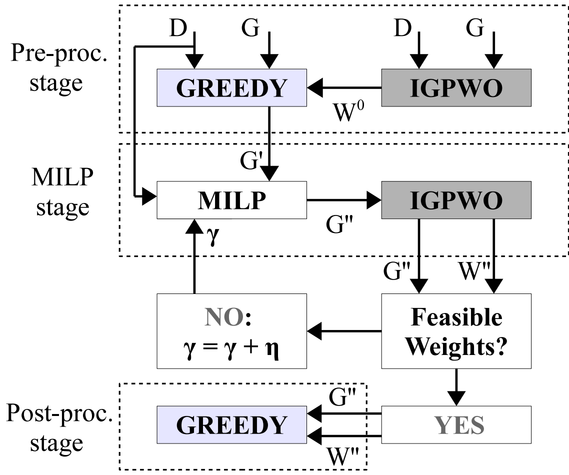


Figure 3: MILP Based Algorithm (MILP-BA) for energy saving

by solving with a 3% gap the following Mixed Integer Linear Program (MILP) that is a relaxation of the E-TESP formulation (1)-(18) where the shortest path constraints are ommitted and traffic demands are assumed to be fully splittable:

$$\min \sum_{(i,j)\in A} p_{ij} x_{ij} + \sum_{k\in V} p_k y_k \tag{23}$$

s. t.

$$x_{ij} \le y_i \quad (i,j) \in A \tag{24}$$

$$x_{ij} \le y_j \quad (i,j) \in A \tag{25}$$

$$\sum_{i\in V} f^t_{it} = \sum_{s\in V} d_{st} \quad t \in V \tag{26}$$

$$\sum_{j\in V} f^t_{vj} - \sum_{i\in V} f^t_{iv} = d_{vt} \quad v, t \in V, t \neq v \tag{27}$$

$$\sum_{t\in V} f^t_{ij} \le x_{ij} \alpha c_{ij} \quad (i,j) \in A \tag{28}$$

$$x_{ij}, y_k \in \{0,1\} \quad k \in V, (i,j) \in A \tag{29}$$

$$f^t_{ij} \ge 0 \quad t \in V\,(i,j) \in A. \tag{30}$$

The objective function (23) amounts to the total network energy consumption. Since the node power consumption p_k is in general much larger (about ten times) than the link power consumption p_{ij}, the priority is to switch off the nodes. Constraints (24)-(30) are identical to Constraints (2)-(6) and (15)-(16).

This formulation falls within the well-known class of minimum cost capacitated multi-commodity flow problems [54]. Due to the complexity reduction achieved by eliminating shortest path variables and constraints from (1)-(18), in this case the state-of-the-art CPLEX 12.2 solver provides in a few seconds feasible solutions within a gap of 3% (with respect to the bound) for sub-networks resulting from the preprocessing of networks with more than 100 nodes and 300 links.

In the *Feasible Routing* step, given the sub-network G'' determined at the *Switching Off* step, we look for a set of link weights W'' that allows to route through G'' all the traffic demands along shortest paths, without exceeding the given maximum link utilization α ($0 < \alpha \leq 1$). To do so, we apply 150 iterations of the IGP-WO algorithm with a maximum weight value of 100. As in the greedy procedure, IGP-WO is warm-started with dual weights inizialization described in [52] for the sub-network G'' found at the *Switching Off* step.

Unfortunately, given the reduced network topology G'' obtained by solving the above MILP formulation, there is no guarantee that there exists a set of link weights W'' allowing feasible shortest path routing of all traffic demands in D. This may occur because the fully splittable routing considered in the formulation (23)-(30) can be hardly reproduced with the OSPF protocol. That means that it may be impossible to respect the maximum utilization constraints in the reduced network topology G'' when a shortest path routing scheme is adopted. For this reason, in case no feasible set of OPSF weights exists (or is found), we slightly increase the original traffic matrix D by multiplying it with a fixed parameter γ, and we repeat the Switch off stage with the increased traffic matrix $D(\gamma)$. At the first iteration $\gamma = 1$ and it is then increased by a parameter η every time IGP-WO fails to find a feasible set of link weights. If the maximum link utilization fraction is larger than α but smaller than $\alpha + 0.1$, $\eta = 0.05$ while, if it is larger than $\alpha + 0.1$, $\eta = 0.1$. The increase of the considered traffic matrix clearly leads to a sub-network G'' with more active elements as input of the *Feasible Routing* stage at the successive iteration. The *Feasible Routing* stage, which is always run with the original traffic matrix D, ends when a feasible set of link weights is found.

To check whether some other elements of the resulting sub-network can still be switched off, we finally perform the post-processing stage: we apply the greedy procedure to the reduced network G'' and the set of feasible weights W'' returned by the MILP-based stage.

6. Computational Results of MILP-EWO and other algorithms

6.1. Network Topologies and Traffic Matrices

We have carried out computational tests on seven real network topologies provided by the Rocketfuel project [55] and on the USA28 network kindly provided to us by the authors of used in [39]. Since our MILP-based heuristic aims at switching off both nodes and links, the main focus is on backbone networks that contain edge nodes as well as core nodes, and whose core nodes may be switched off. For comparison purposes with [28],[30] and [39], we also consider access networks that only contain edge nodes, which cannot be switched off. The characteristics of the eight network topologies are summarized in Table 3. The five Rocketfuel access networks, Abovenet, AT&T, Genuity, Sprint and Tiscali, have been used in [28] and [30] for testing other types of energy-aware traffic engineering approaches (see Section 2).

Network	Type	Nodes	Links	Edge	Core	%Core
Ebone	Backbone	87	322	31	56	64.4
Exodus	Backbone	79	294	38	41	51.9
Sprint	Access	52	168	52	0	0
AT&T	Access	115	296	115	0	0
Abovenet	Access	19	68	68	0	0
Genuity	Access	42	110	42	0	0
Tiscali	Access	41	174	41	0	0
USA28	Access	28	90	28	0	0

Table 3: Network topologies used in the experiments. The first seven ones are real network topologies from the Rocketfuel project [55], while the last one is used in [39].

Unfortunately, the Rocketfuel project data does not include link capacity and network equipment information. For the backbone networks Ebone and Exodus, we assume that all the links have the same capacity, each node is equipped with M10i routers (power consumption p_i of $86.4\,W$), and each link is equipped with a Gigabit Ethernet line card (power consumption p_{ij} of $7.3\,W$ and capacity c_{ij} of $2\,Gbps$). Since a M10i router can support at most eight Gigabit Ethernet line cards, the number of routers in each node directly depends on the degree g_i of the node ($\lceil \frac{g_i}{8} \rceil$ routers in each node). As for the access networks already, we consider the capacity values and the equipment configurations previously used in used in [28, 30, 39]. More specifically, in the Sprint and AT&T networks used in [28] links have capacities of $9920\,Gbps$ or $2480\,Gbps$, while in the Abovenet, AT&T, Genuity, Sprint, and Tiscali networks tested in [30] links have capacities of $100\,Mbps$ or $52\,Mbps$. Finally, in the USA28 network ised in [39] there are only $100\,Mbps$ links.

For the backbone networks (Ebone and Exodus), we have randomly selected the set of edge nodes out of the given set of nodes, making sure that all the leaf nodes are considered as edge nodes and at least one edge node is selected for each city. The remaining nodes are considered as core nodes.

As to the traffic matrices, for access networks we use the same traffic matrices as in [28],[30] and [39]. For the backbone networks Ebone and Exodus, we consider two types of traffic matrices with a nonzero demand between each pair of edge nodes:

i) *Constant* and *Poisson* matrices, which are generated using the Totem toolbox [20], correspond to the maximum load matrices with constant and Poisson traffic distribution that can be supported by the fully active network with OSPF routing when all weights are equal to 1.

ii) *LP-based multicommodity flow* matrices, which are obtained by scaling with a parameter $\beta \in (0, 1)$, a maximum traffic matrix determined by maximizing (via Linear Programming) the sum of traffic demands over all the source-destination pairs subject to the link capacity constraints as well as to lower and upper bounds on each traffic demand (when the given network is fully active).

For the sake of simplicity, in our tests we consider the maximum link utilization parameter $\alpha = 1$.

6.2. Results

The computational experiments have been carried out on an Intel Pentium Duo 3.0GHz with 3.5GB of RAM. The results for backbone networks are reported in Tables 4, 5 and 6, while

Inst					Bound	Greedy		GRASP		MILP-EWO$_{noprep}$		MILP-EWO	
ID	Net	$C-E$	L	$E_c^{tot}\,(W)$	$E_c\,(W)$	$E_c\,(W)$	gap	$E_c\,(W)$	gap	$E_c\,(W)$	gap	$E_c\,(W)$	gap
1	Ex30	41-38	294	9058.2	4546.2	5239.8	15.3%	5124.2	12.7%	4929.5	8.4%	4922.2	8.3%
2	Ex40	41-38	294	9058.2	5131.5	5753.3	12.1%	5738.7	11.8%	5342.0	4.1%	5399.2	5.2%
3	Ex50	41-38	294	9058.2	5536.7	6418.9	15.9%	6519.9	17.8%	6216.9	12.3%	6195.0	11.9%
4	ExC	41-38	294	9058.2	4537.7	5058.5	11.5%	4928.3	8.6%	4682.5	3.2%	4704.4	3.7%
5	ExP	41-38	294	9058.2	4653.3	5346.9	14.9%	5152.2	10.7%	4820.0	3.6%	4805.4	3.3%
6	Eb30	56-31	322	10126.6	5540.4	5677.9	2.5%	5670.6	2.4%	5670.6	2.4%	5569.6	0.5%
7	Eb40	56-31	322	10126.6	5872.6	6248.6	6.4%	6162.2	4.9%	6111.1	4.1%	6096.5	3.8%
8	Eb50	56-31	322	10126.6	6327.7	7136.9	12.8%	7107.7	12.3%	6689.1	5.7%	6667.2	5.4%
9	EbC	56-31	322	10126.6	5865.3	6313.1	7.6%	6298.5	7.4%	6002.8	2.3%	5931.0	1.1%
10	EbP	56-31	322	10126.6	5865.3	6320.4	7.8%	6226.7	6.2%	6096.5	3.9%	6010.1	2.5%

Table 4: Comparison between the energy savings achieved with by the greedy heuristic, GRASP, MILP-EWO$_{noprep}$ and MILP-EWO. The bound on the energy consumption is obtained by solving the MILP relaxation (23)-(30).

Inst	Greedy		GRASP		MILP-EWO$_{noprep}$		MILP-EWO	
ID	N_{off}	L_{off}	N_{off}	L_{off}	N_{off}	L_{off}	N_{off}	L_{off}
1	30 (73.2%)	168 (57.1%)	31 (75.6%)	172 (58.5%)	33 (80.5%)	175 (59.5%)	33 (80.5%)	176 (59.9%)
2	26 (63.4%)	145 (63.4%)	26 (63.4%)	147 (50.0%)	30 (73.2%)	154 (52.4%)	29 (70.7%)	158 (53.7%)
3	21 (51.2%)	113 (51.2%)	20 (48.8%)	111 (37.8%)	23 (56.1%)	117 (39.8%)	23 (56.1%)	120 (40.8%)
4	31 (75.6%)	181 (75.6%)	32 (78.0%)	187 (63.6%)	34 (82.9%)	197 (67.0%)	34 (82.9%)	194 (66.0%)
5	28 (68.3%)	177 (68.3%)	30 (73.2%)	180 (61.2%)	33 (80.5%)	190 (64.6%)	33 (80.5%)	192 (65.3%)
6	34 (60.7%)	207 (60.7%)	34 (60.7%)	208 (64.6%)	34 (60.7%)	208 (64.6%)	35 (62.5%)	210 (65.2%)
7	29 (51.8%)	188 (51.8%)	30 (53.6%)	188 (58.4%)	30 (53.6%)	195 (60.6%)	30 (53.6%)	197 (61.2%)
8	21 (37.5%)	161 (37.5%)	21 (37.5%)	165 (51.2%)	25 (44.6%)	175 (54.3%)	25 (44.6%)	178 (55.3%)
9	28 (50.0%)	191 (50.0%)	28 (50.0%)	193 (59.9%)	30 (53.6%)	198 (61.5%)	31 (55.4%)	198 (61.5%)
10	28 (50.0%)	190 (50.0%)	29 (51.8%)	191 (59.3%)	30 (53.6%)	197 (61.2%)	31 (55.4%)	197 (61.2%)

Table 5: Comparison between the number of elements put to sleep by the greedy heuristic, GRASP, MILP-EWO$_{noprep}$ and MILP-EWO.

Inst	Greedy		GRASP		MILP-EWO$_{noprep}$		MILP-EWO	
ID	$Cong$	$t\,(s)\;(t_{nor})$	$Cong$	$t\,(s)\;(t_{nor})$	$Cong$	$t\,(s)\;(t_{nor})$	$Cong$	$t\,(s)\;(t_{nor})$
1	229%	1159 (1)	264%	16224 (14.00)	318%	4732 (4.08)	300%	2189 (1.89)
2	224%	1146 (1)	239%	21152 (18.46)	263%	10509 (9.17)	352%	2024 (1.77)
3	161%	1252 (1)	162%	25323 (20.23)	174%	10020 (8.00)	171%	4983 (3.98)
4	262%	767 (1)	224%	14632 (19.08)	315%	1939 (2.53)	299%	1863 (2.43)
5	215%	957 (1)	185%	15329 (16.02)	243%	4369 (4.57)	253%	1798 (1.88)
6	270%	2217 (1)	300%	10121 (4.57)	281%	9062 (4.09)	302%	2674 (1.21)
7	217%	1897 (1)	194%	13053 (6.88)	269%	29808 (15.71)	279%	2606 (1.37)
8	190%	1818 (1)	182%	18218 (10.02)	202%	26303 (14.47)	213%	3663 (2.01)
9	211%	1624 (1)	196%	13075 (8.05)	241%	9701 (5.97)	280%	1868 (1.15)
10	209%	1434 (1)	201%	12163 (8.48)	242%	7657 (5.34)	209%	2648 (1.85)

Table 6: Comparison between the computing times and the congestion achieved by the greedy heuristic, GRASP, MILP-EWO$_{noprep}$ and MILP-EWO.

those for the Rocketfuel access networks are shown in Table 7. The results of the performance comparison between our MILP-based algorithm and those proposed in [39] are instead reported in Figure 4.

For comparison purposes, we have also developed a basic greedy heuristic and a Greedy Randomized Adaptive Search Procedure (GRASP) for the E-TESP problem, see e.g. [56] and [57] for surveys on GRASP. The basic greedy heuristic proceeds like the pre-processing stage of MILP-EWO with only two differences: i) the maximum link utilization parameter is equal to α and not $\alpha - 0.1$, and ii) the IGP-WO algorithm is also run after the greedy procedure in order to minimize the total cost of link utilization. At each iteration of our GRASP, the network element to be switched off is randomly selected among the first k% elements of the ordered list derived from the node-link sorting criteria. Since different executions of such a randomized greedy procedure yield different solutions, the best solution obtained over a predefined number of repetitions is returned as an approximate solution.

To evaluate the impact of the pre-processing stage, we also consider a version of MILP-EWO without the pre-processing, that we refer to as MILP-EWO$_{noprep}$. On the one hand, we expect that MILP-EWO$_{noprep}$ is computationally heavier than MILP-EWO because it is based on the MILP formulation (23)-(30) for the original network topology rather than for a subnetwork. On the other hand, we expect an improvement in the solution quality since all decisions are taken from a global perspective, unlike in the greedy pre-processing stage.

The first two columns of Tables 4 and 7 indicates the instance under consideration, by defining the test ID, the network topology and the traffic matrix. The ID field is used to avoid to repeat redundant information in Tables 5 and 6, where only

Instances used in [28] – Link capacities of 9920 $Gbps$ or 2480 $Gbps$										
ID	$Inst$	N	L	$E_c^{tot}\,(W)$	$Cong_{min}$	$E_c\,(W)$	N_{off}	L_{off}	$Cong$	$t\,(s)$
11	Spr1	52	168	24972	28785	11950 (47.9%)	0 (0.0%)	85 (50.6%)	160774 (558%)	578
12	Spr2	52	168	24972	59651	13339 (53.4%)	0 (0.0%)	76 (45.2%)	476098 (798%)	584
13	Spr3	52	168	24972	96214	13795 (55.2%)	0 (0.0%)	73 (43.5%)	412256 (428%)	1241
14	AT&T1	115	296	43344	38990	30504 (70.4%	0 (0.0%)	82 (27.7%)	215903 (553%)	1816
15	AT&T2	115	296	43344	77980	31026 (71.6%)	0 (0.0%)	79 (26.7%)	323802 (415%)	1854
16	AT&T3	115	296	43344	117347	32388 (74.7%)	0 (0.0%)	70 (23.6%)	616849 (525%)	1990
Instances used in [30] – Link capacities of 100 $Mbps$ or 52 $Mbps$										
ID	$Inst$	N	L	$E_c^{tot}\,(W)$	$Cong_{min}$	$E_c\,(W)$	N_{off}	L_{off}	$Cong$	$t\,(s)$
17	Abov1	19	68	7610	1079812	4910 (64%)	1 (5.0%)	35 (51.5%)	2458879 (228%)	59
18	Abov2	19	68	7610	1528712	5200 (68%)	0 (0.0%)	33 (48.5%)	3816638 (250%)	42
19	Abov3	19	68	7610	2118525	5580 (73%)	0 (0.0%)	29 (42.6%)	4659640 (219%)	73
20	AT&T1	115	296	37970	2251536	23130 (61%)	35 (30.4%)	137 (46.3%)	5388692 (240%)	662
21	AT&T2	115	296	37970	279125	23060 (60%)	35 (30.4%)	138 (46.6%)	6069666 (217%)	356
22	AT&T3	115	296	37970	3563969	23270 (61%)	35 (30.4%)	135 (45.6%)	5771727 (162%)	419
23	Gen1	42	110	14000	1841491	10670 (76%)	4 (9.5%)	39 (35.5%)	4960005 (270%)	182
24	Gen2	42	110	14000	2446328	10810 (77%)	4 (9.5%)	37 (33.6%)	4420327 (180%)	251
25	Gen3	42	110	14000	2972444	10810 (77%)	4 (9.5%)	37 (33.6%)	5691743 (191%)	115
26	Spr1	52	168	19560	2691568	13430 (68%)	4 (7.7%)	79 (47.0%)	7724987 (287%)	528
27	Spr2	52	168	19560	3351249	13780 (70%)	4 (7.7%)	74 (44.0%)	8319724 (248%)	298
28	Spr3	52	168	19560	4544579	14130 (72%)	4 (7.7%)	69 (41.1%)	4960005 (175%)	968
29	Tis1	41	174	18330	2534503	11660 (64%)	2 (4.9%)	91 (52.3%)	6459629 (255%)	368
30	Tis2	41	174	18330	2621918	11520 (63%)	2 (4.9%)	93 (53.4%)	4808477 (183%)	270
31	Tis3	41	174	18330	3426815	11590 (63%)	2 (4.9%)	92 (52.9%)	6334003 (184%)	346

Table 7: Computational results obtained with MILP-EWO for the instances used in [28] and [30].

the test ID is used to identify the selected instance. Ex, Eb, Spr, Abov, Gen and Tis, are the abbreviations for, respectively, Exodus, Ebone, Sprint, Abovenet, Genuity and Tiscali, while the letters C and P, and the numbers (30-40-50 for backbone networks, 1-2-3 for access networks) that follow the network acronyms are related to the traffic matrix considered. Specifically, C corresponds to constant matrices, P to Poisson matrices, 30-40-50 to the maximum supported LP matrices scaled by 0.3, 0.4 and 0.5, 1-2-3 for the different traffic levels (respectively low, moderate, high). The columns $C-E$, L, $E_c^{tot}\,(W)$, and $Cong_{min}$ report respectively: the number of core and edge nodes, the number of links, the energy consumption of the fully active network and the optimized congestion obtained with the fully active network. As congestion measure, we use the value of the cost function defined by IGP-WO, namely the total cost of link utilization. The remaining columns are related to the energy performance: $E_c\,(W)$ is the energy consumption of the reduced networks, and gap is the ratio $(E_c - E_c^b)/E_c^b$, where E_c^b is the lower bound on the energy consumption. Note that this energy consumption lower bound is computed by solving the MILP formulation (23)-(30) with fully splittable routing (not shortest path routing) to optimality. $Cong$ reports the congestion levels, $Cong_{\%}$ the ratio $Cong/Cong_{min}$. Finally, N_{off} and L_{off} indicate the number of nodes and links that are switched off, while t and t_{nor} are, respectively, the total computing time and the total computing time normalized w.r.t. the computing time of the greedy heuristic.

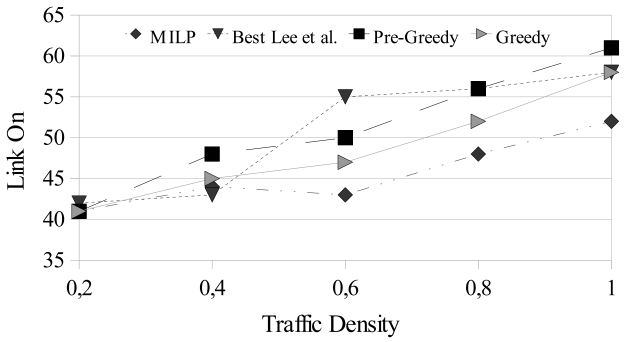


Figure 4: Comparison between the results obtained running our heuristics and the best algorithm proposed in [39] with USA28 topology. On y-axis we report the number of active links remained at the end of the elaborations. For each of the five levels of traffic (from 0.2 to 1), we perform tests with ten different randomly generated traffic matrices. The values reported in the figure are thus the mean values. The code to generate the matrix has been provided by the authors of [39].

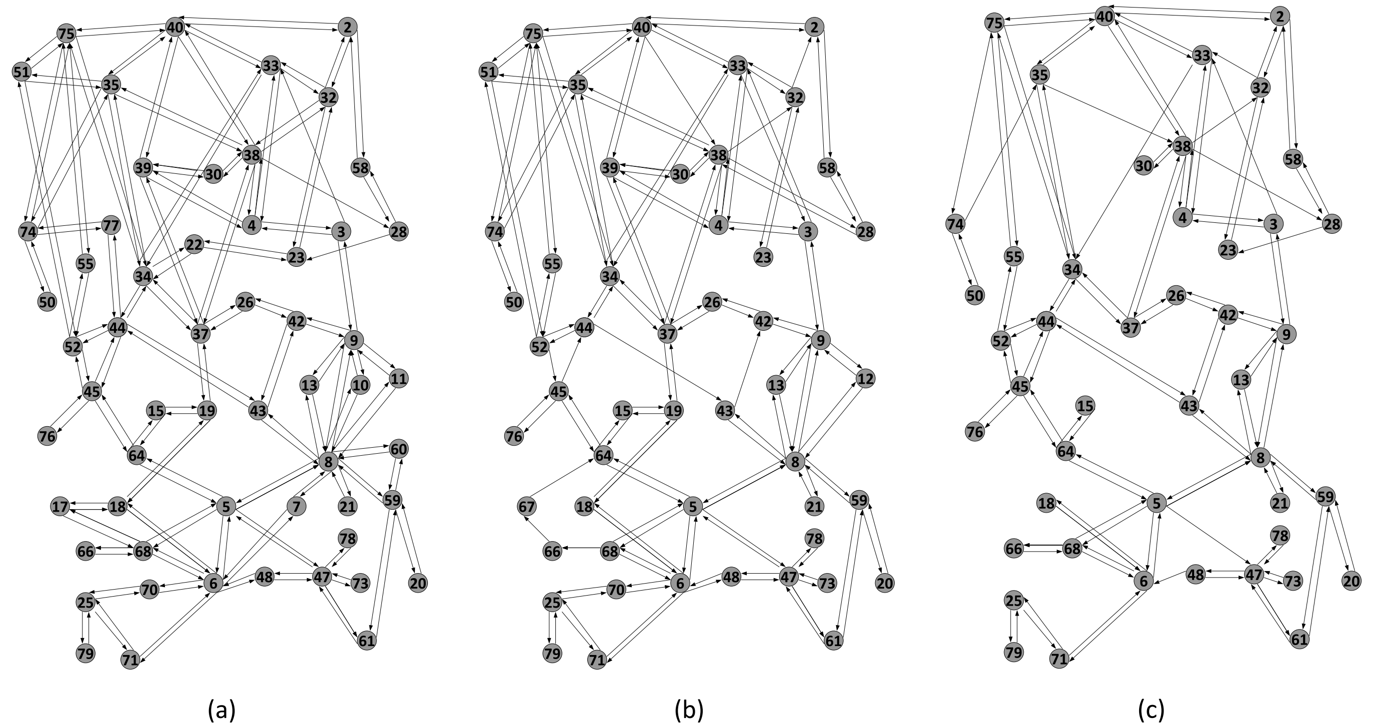


Figure 5: Sub-networks obtained with MILP-EWO for the Exodus network with the LP traffic matrix scaled by (a) $\beta = 0.5$, (b) $\beta = 0.4$ and (c) $\beta = 0.3$.

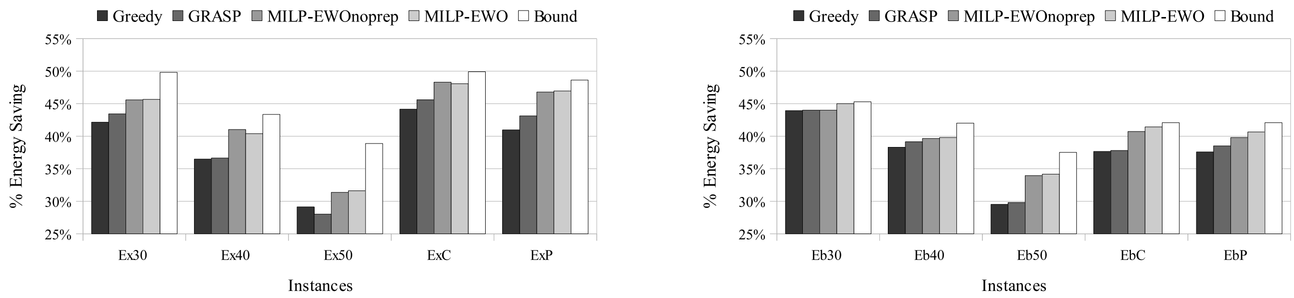


Figure 6: Comparison of the percentage of energy saving achieved by the different heuristics (Greedy, GRASP, MILP-EWO$_{noprep}$, MILP-EWO) with Exodus and Ebone networks.

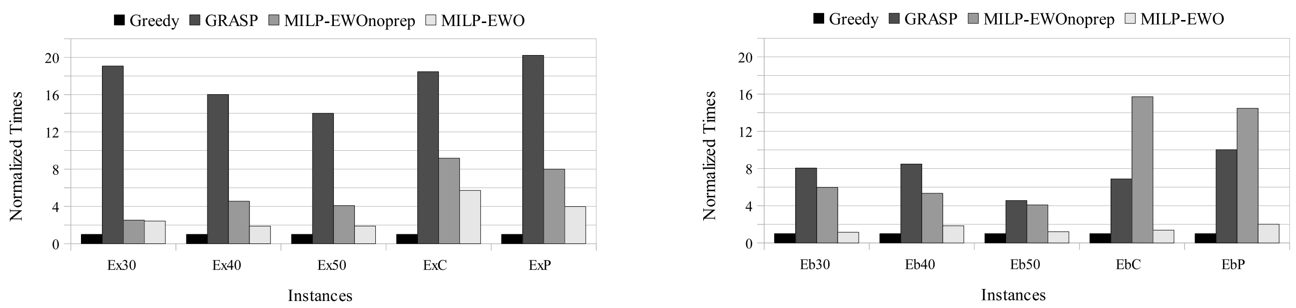


Figure 7: Comparison of the heuristic computing times for Exodus and Ebone networks.

In TableS 4, 5,6 and Figures 6, 7 we report the results obtained with the greedy procedure, GRASP, MILP-EWO$_{noprep}$ and MILP-EWO on the backbone networks.

The solutions found by GRASP are of slightly better quality than those provided by the greedy procedure but computing times are much higher. In one case, the normalized computing time t_{nor} reaches even the value of 20. This is due to the multi-start strategy, where the greedy is repeated 100 times for each one of the sorting policies.

MILP-EWO$_{noprep}$ is very heavy computationally (in the worst cases $t_{nor} = 15$) but the quality of the solutions is much better than those found by GRASP. The gap is small, in average smaller than 5%. However, a computing time of 2 up to 8 hours is needed to solve the Ebone instances with up to 87 nodes (56 core nodes and 322 links).

MILP-EWO turns out to achieve a remarkable trade-off. Computing times are much smaller than those of MILP-EWO$_{noprep}$ (t_{nor} at most 6 and usually smaller than 2), while the solution quality remains the same (except for Ex40). Computing times are generally less than a half-hour, and only about half of the time is used to solve the MILP formulations. For this reasons we think that MILP-EWO could be successfully used for efficiently solve also bigger instances (up to network with 300 nodes).

Concerning the congestion level, it is worth pointing out that in general the total cost of link utilization increases reasonably: the normalized cost of link utilization (see [13]) does not usually exceed 0.3.

Examples of the sub-networks topologies provided by MILP-EWO for the Exodus network with the LP traffic matrix scaled by $\beta = 0.5,\ 0.4,\ 0.3$ are shown in Figure 5. As expect, when the traffic decreases it is possible to switch off a larger number of networks elements.

Table 7 reports the results obtained with MILP-EWO for the different instances used in [28] and [30]. Also in this case, although the presence of heterogeneous link capacities, a large number of the links can be switched off (generally around 40% of the links). The lower percentage of energy saving achievable for AT&T networks (about half compared with that for Sprint networks) is due to the lower link redundancy and the higher number of leafs that characterize the AT&T topology. The largest computing times are of the order of 30 minutes. Note that, although the networks used in [30] are access networks, our algorithms manage to switch off also some nodes, because the traffic matrices that were kindly provided to us only contain demands between some pairs of nodes. In some cases (Spr2) the percentage increase in congestion is much larger than for backbone networks (up to 8 times), but the normalized value remains always under 0.4.

Finally, we compare the performance of our methods with that of the best heuristic proposed in [39] for energy-aware link weights optimization. Recall that in the Lagrangian Relaxation plus Harmonic Series (LR&HS) heuristic [39] only links are switched off and network congestion is not taken into account. As shown in Figure 4, when the traffic increases and the final solution is therefore not a simple tree, MILP-EWO is able to switch off up to 20% more links than LR&HS. It is worth pointing out that even the pre-processing greedy procedure of MILP-EWO yields better quality solutions than LR&HS. Moreover, it is important to recall that in MILP-EWO, unlike in LR&HS, network congestion is also minimized.

7. Experimental Evaluation via Network Emulation

In order to asses the validity and the applicability of our approach in real conditions, we have experimented with the energy-aware weights found by the MILP-EWO$_{noprep}$ algorithm, via simulations with an emulated environment.

7.1. Our test-bed

Our test-bed has been developed using Netkit ([58][59]) a freely available network emulation environment based on User-Mode Linux, and a set of other networking tools (the OSPF daemon Zebra [60], the constant traffic generator Iperf [61] and the traffic monitor tool vnStat [62].

We have used Netkit to reproduce the real network topology called Abilene [63], that we shall refer to as *Original Abilene*. In order to have the possibility to switch off both links and nodes, we have also considered a slightly modified version of Abilene network, the *Extended Abilene*, obtained by adding auxiliary edge nodes around the core nodes. We have not been able to use larger network topologies because machines cannot support more than 30 virtual routers.

As to the traffic matrices, we considered modified versions of the real traffic matrices available at [63]. For the *Original Abilene* topology, we have scaled the real traffic matrices with a parameter equal to 2.5 in order to reach a maximum utilization during the peak hours of around 60%. For the *Extended Abilene* topology, we have also shifted the traffic demands from the core nodes to the auxiliary edge nodes.

7.2. Tests

We have conducted three different types of experiments. First, we have compared the utilization levels recorded along an entire day, running the simulations both with the original weights and the energy-aware link weights obtained by MILP-EWO$_{noprep}$ with the traffic matrices of 6:00 am and 12:00 pm and different values of maximum allowed link utilization α. Then we have analysed the variations in the utilization levels caused by a change of weights when the network begins to be saturated. Finally, we have studied the reaction of the system to the breakdown of a single link.

7.2.1. Original Abilene

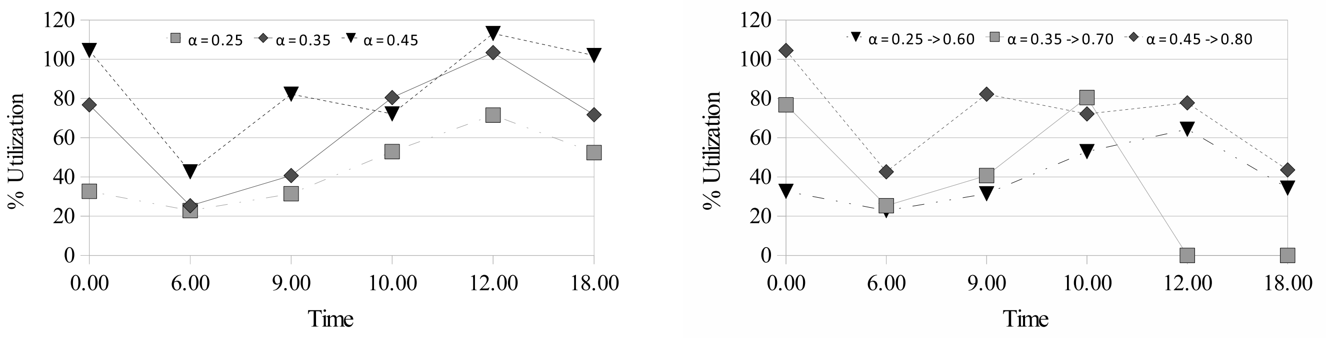
The results obtained running MILP-EWO$_{noprep}$ with the traffic matrix corresponding to 6:00 am of 04/09/04 are reported in Tables 9. As expected, the higher energy saving is achieved when the maximum link utilization parameter is increased. Although the original matrices have been scaled by 2.5 (incremented), the weights obtained with the minimum traffic level maintain the link utilization under an acceptable level ($\leq 70\%$) also during the peak hours. This can be explained by the fact

Network	Nodes	Links	Edge	Core	%Core
Original Abilene	12	32	12	0	0%
Extended Abilene	20	62	8	20	71%

Table 8: Network topologies used for the experimental evaluations.

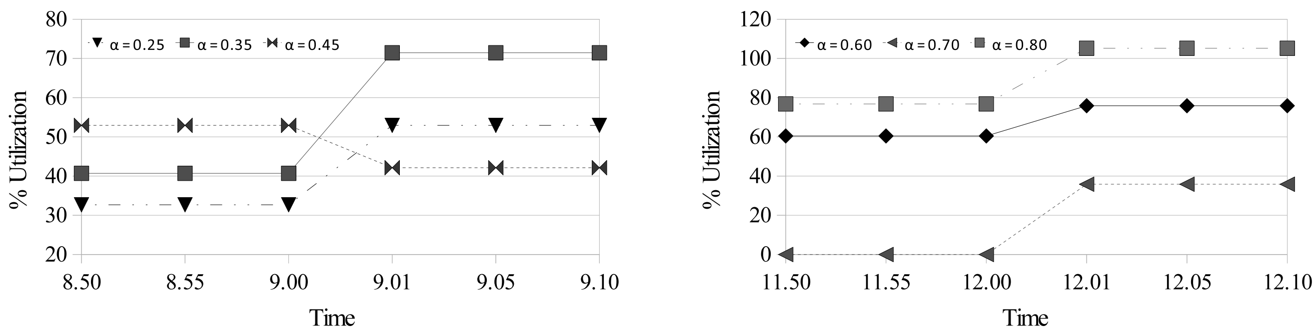
Extended Abilene: Traffic of 04/09/04				
α	L_{off}-N_{off} (6am)	L_{off}-N_{off} (12am)	$\%_{saving}$ (6am)	$\%_{saving}$ (12am)
$\leq$ 20%	infeasible	infeasible	infeasible	infeasible
25%	45-3	infeasible	21,38%	infeasible
30%	47-4	infeasible	28,52%	infeasible
35%	50-5	infeasible	37,45%	infeasible
40%	53-5	infeasible	37,45%	infeasible
45%	54-7	infeasible	53,52%	infeasible
50%	54-7	infeasible	53,52%	infeasible
60%	54-7	46-2	53,52%	12,45%
70%	54-7	42-3	53,52%	19,58%
$\geq$ 80%	54-7	40-3	53,52%	21,38%

Table 10: Number of links and nodes switched off by MILP-EWO$_{noprep}$ when run on *Extended Abilene* with different values of the maximum link utilization α. The traffic values are those recorded at 6:00 and 12:00 am on 04/09/04 (minimum and maximum traffic values).



(a) (b)

Figure 8: *Extended Abilene*: utilization of link Sunnyvale-Kansas when the weights of 6:00 am a) are used for the entire day and b) when they are changed at 10:00 am with those computed with the traffic of 12:00 am.



(a) (b)

Figure 9: *Extended Abilene*: utilization of link Sunnyvale-Kansas when using a) the constant traffic of 9:00 am and the weights of 6:00 am and b) the constant traffic of 12:00 and the weights of 12:00 am.

that Abilene is a study network and for this reason the difference between low and peak traffic periods are smaller than for typical ISP networks.

7.2.2. Extended Abilene

The results obtained running MILP-EWO$_{noprep}$ with the traffic matrix corresponding to 6:00 am and 12:00 am of 04/09/04 are reported in Table 10. As expected, also in this case, the higher energy saving is achieved when the maximum link utilization parameter is increased. In Figures 8, 9 and 10, we report as an example the utilization values recorded on the link Sunnyvale-Kansas during the different simulations.

Unlike in the experiments with *Original Abilene*, the weights found by MILP-EWO$_{noprep}$ for the low traffic periods are not able to support the traffic peak levels. In Figure 8 we show how changing weights at 10:00 am allows to maintain the utilization

Original Abilene: Traffic of 04/09/04		
α	L_{off}-N_{off} (6*am*)	$\%_{saving}$ (6*am*)
5%	9-0	16,25%
10%	15-0	27,08%
15%	18-0	32,50%
$\geq$ 20%	19-0	34,31%

Table 9: Number of links and nodes switched off by MILP-EWO$_{noprep}$ when run on *Original Abilene* with different values of maximum link utilization α. The traffic values are those recorded at 6:00 am of 04/09/04 (minimum traffic values).

of the links under acceptable levels ($\leq$ 70%). Thus, a basic energy efficient strategy could consist in simply computing two sets of energy-aware link weights, one for the night and one for the day. Note that the peak levels registered at 0:00 am are due to the initial routine of Netkit at the beginning of the simulations, and thus they are neglected in the evaluations.

In Figure 9 we illustrate the case of a single link breakdown with constant traffic in both low and high traffic scenarios. As highlighted in Figure 10, 40 seconds are sufficient to the network to converge to a new stable situation (the hello interval is set to 10 seconds). Note that the time interval needed for awakening a link or a router has not been considered. The link breakdown causes in general a natural increment in the utilization of the other links It is important to point out that the link breakdown can be efficiently absorbed by the network only if MILP-EWO$_{noprep}$ is run with a sufficiently low maximum link utilization parameter value α. According to our simulations, α should be of at most 70% in order to avoid capacity saturation when a single breakdown occurs. In some cases the breakdown can also cause the awake of a sleeping link.

8. Conclusions

We have investigated the relevant and challenging problem of energy-aware IP traffic engineering with shortest path routing. We have proposed a MILP-based heuristic which aims at minimizing the energy consumption as well as the total cost of link utilization in backbone and access networks of ISPs. The computational results for eight real network topologies and different types of traffic matrices show that the efficient configuration of link weights allows to achieve remarkable energy saving (up to 50% of savings during low traffic periods) in IP networks with homogeneous and heterogeneous network equipments, by switching off a substantial number of core nodes during low and moderate traffic periods, while guaranteeing the same point-to-point service quality and reasonably increasing the network total cost of link utilization.

MILP-EWO constitutes a good trade-off between the solution quality and computational complexity: it provides solutions with small gaps with respect to the available bound within moderate computing times even for large-size instances (even still bigger instances can be efficiently solved).

The validity of our approach is also confirmed by the evaluations via network emulation (using Netkit) for two network topologies. The experiments indicate that the energy-efficient configuration of link weights provided by MILP-EWO makes the network reliable with respect to breakdowns and unexpected traffic variations.

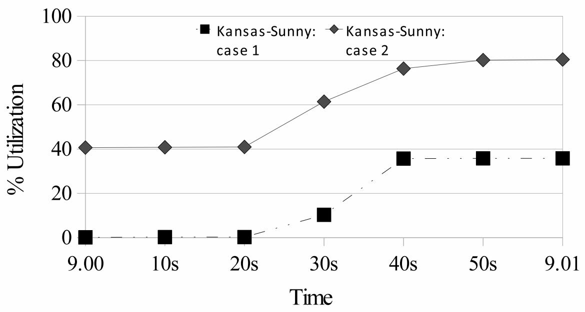


Figure 10: Effect of a topology change (link breakdown or weights change). The routing converges to stability in less than a minute.

Acknowledgment

The authors would like to thank Luca Mascetti and Guido Caldara for helping with the implementation of the greedy algorithms and, respectively, the experimental evaluation via network emulation. Morever, they are greateful to Nedeljko Vasic, Cheng Ye and Steven Lee for providing the network topologies as well as traffic matrices they used in their respective related works.

[1] J.G. Koomey. Estimating total power consumption by servers in the US and the world, 2007.

[2] W. Vereecken, L. Deboosere, D. Colle, B. Vermeulen, M. Pickavet, B. Dhoedt, and P. Demeester. Energy efficiency in telecommunication networks. 2008.

[3] Raffaele Bolla, Roberto Bruschi, Alessandro Carrega, Franco Davoli, Diego Suino, Constantinos Vassilakis, and Anastasios Zafeiropoulos. Cutting the energy bills of internet service providers and telecoms through power management: An impact analysis. *Computer Networks*, 56(10):2320 – 2342, 2012. ¡ce:title¿Green communication networks¡/ce:title¿.

[4] GreenTouch consortium.

[5] M. Gupta and S. Singh. Greening of the Internet. In *Proc. of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 19–26. ACM, 2003.

[6] H. Mellah and B. Sansò. Review of facts, data and proposals for a greener Internet. In *Proc. of Broadnets09*, pages 1–5, sept. 2009.

[7] M. Minami and H. Morikawa. Some open challenges for improving the energy efficiency of the internet. In *Proc. of 3rd International Conference on Future Internet (CFI 2008)*, 2008.

[8] C. Fraleigh, S. Moon, B. Lyles, C. Cotton, M. Khan, D. Moll, R. Rockell, T. Seely, and SC Diot. Packet-level traffic measurements from the Sprint IP backbone. *Network*, 17(6):6–16, 2003.

[9] J. Chabarek, J. Sommers, P. Barford, C. Estan, D. Tsiang, and S. Wright. Power awareness in network design and routing. In *Proc. of INFOCOM 2008. The 27th Conference on Computer Communications.*, pages 457–465. IEEE, 2008.

[10] P. Mahadevan, P. Sharma, S. Banerjee, and P. Ranganathan. A power benchmarking framework for network devices. *NETWORKING 2009*, pages 795–808, 2009.

[11] L. Chiaraviglio, D. Ciullo, M. Mellia, and M. Meo. Modeling sleep modes gains with random graphs. In *Computer Communications Workshops (INFOCOM WKSHPS), 2011 IEEE Conference on*, pages 355–360. IEEE, 2011.

[12] R. Bolla, R. Bruschi, F. Davoli, and A. Ranieri. Performance constrained power consumption optimization in distributed network equipment. In *1st Workshop on Green Communications, GreenCom '09*.

[13] B. Fortz and M. Thorup. Internet traffic engineering by optimizing OSPF weights. In *Proc. of IEEE INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 2, pages 519–528.

[14] A. Altın, B. Fortz, M. Thorup, and H. Ümit. Intra-domain traffic engineering with shortest path routing protocols. *4OR: A Quarterly Journal of Operations Research*, 7(4):301–335, 2009.

[15] A. Bley, B. Fortz, E. Gourdin, K. Holmberg, O. Klopfenstein, M. Pioro, A. Tomaszewski, and H. Umit. Optimization of ospf routing in ip networks. *Graphs and algorithms in communication networks: studies in broadband, optical, wireless and ad hoc networks*, page 199, 2009.

[16] Active/idle toggling with 0base-x for energy efficient ethernet. presentation to the IEEE 802.3az Task Force, Nov. 2007.

[17] A. Mackarel and et al. Study of environmental impact, dn3.5.2, geant project, May 2011.

[18] E. Amaldi, A. Capone, L.G. Gianoli, and L. Mascetti. Energy management in IP traffic engineering with shortest path routing. In *Proc. of World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2011 IEEE International Symposium on a*, pages 1–6. IEEE, 2011.

[19] E. Amaldi, A. Capone, L.G. Gianoli, and L. Mascetti. A MILP-based heuristic for energy-aware traffic engineering with shortest path routing. In J. Pahl, T. Reiners, and S. Voß, editors, *Proc. of INOC*, volume 6701 of *Lecture Notes in Computer Science*, pages 464–477. Springer, 2011.

[20] Totem Project [Online]. `http://totem.run.montefiore.ulg.ac.be`.

[21] A. Bianzino, C. Chaudet, D. Rossi, and J. Rougier. A survey of green networking research. *Communications Surveys & Tutorials*, (99):1–18, 2010.

[22] 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. *Communications Surveys & Tutorials*, (99):1–22.

[23] M. Baldi and Y. Ofek. Time for a greener internet. In *Communications Workshops, 2009. ICC Workshops. IEEE International Conference on*, pages 1–6.

[24] J.C.C. Restrepo, C.G. Gruber, and C.M. Machuca. Energy profile aware routing. In *Communications Workshops, 2009. ICC Workshops. IEEE International Conference on*, pages 1–5, 2009.

[25] A.P. Bianzino, C. Chaudet, F. Larroca, D. Rossi, and J. Rougier. Energy-aware routing: A reality check. In *IEEE GLOBECOM Workshops (GC Wkshps)*, pages 1422–1427, 2010.

[26] R. Bolla, F. Davoli, R. Bruschi, K. Christensen, F. Cucchietti, and S. Singh. The potential impact of green technologies in next-generation wireline networks: Is there room for energy saving optimization? *Communications Magazine, IEEE*, 49(8):80–86, 2011.

[27] A.A. Kist and A. Aldraho. Dynamic topologies for sustainable and energy efficient traffic routing. *Computer Networks*, 55(9):2271–2288, 2011.

[28] M. Zhang, C. Yi, B. Liu, and B. Zhang. Greente: Power-aware traffic engineering. In *Network Protocols (ICNP), 2010 18th IEEE International Conference on*, pages 21–30. IEEE, 2010.

[29] L. Chiaraviglio, M. Mellia, and F. Neri. Minimizing isp network energy cost: Formulation and solutions. *IEEE/ACM Transactions on Networking (TON)*, 20(2):463–476, 2012.

[30] N. Vasić and D. Kostić. Energy-aware traffic engineering. In *Proc. of the 1st International Conference on Energy-Efficient Computing and Networking*, pages 169–178. ACM, 2010.

[31] B. Addis, A. Capone, G. Carello, L.G. Gianoli, and B. Sanso. Energy-aware multiperiod traffic engineering with flow-based routing. In *Communications (ICC), 2012 IEEE International Conference on*, pages 5957–5961, 2012.

[32] R.G. Garroppo, S. Giordano, G. Nencioni, M. Pagano, and Scutellá M.G. Models and heuristic approaches for network power management. In *Proc. GLOBECOM 2011 IEEE Global Telecommunications Conference*, 2011.

[33] G. Athanasiou, K. Tsagkaris, P. Vlacheas, and P. Demestichas. Introducing energy-awareness in traffic engineering for future networks. In *Network and Service Management (CNSM), 2011 7th International Conference on*, pages 1–4. IEEE, 2011.

[34] H.W. Chu, C.C. Cheung, K.H. Ho, and N. Wang. Green mpls traffic engineering. In *Australasian Telecommunication Networks and Applications Conference (ATNAC), 2011*, pages 1–4. IEEE.

[35] F. Giroire, J. Moulierac, T. Phan, and F. Roudaut. Minimization of network power consumption with redundancy elimination. *NETWORKING 2012*, pages 247–258, 2012.

[36] A. Cianfrani, V. Eramo, M. Listanti, M. Polverini, and A. Vasilakos. An ospf-integrated routing strategy for qos-aware energy saving in ip backbone networks. *Network and Service Management, IEEE Transactions on*, PP(99):1–14, 2012.

[37] A. Sridharan, R. Guerin, and C. Diot. Achieving near-optimal traffic engineering solutions for current ospf/is-is networks. *Networking, IEEE/ACM Transactions on*, 13(2):234–247, 2005.

[38] M. Pióro and D. Medhi. *Routing, flow, and capacity design in communication and computer networks*. Morgan Kaufmann Publishers, 2004.

[39] S.S.W. Lee, P.K. Tseng, and A. Chen. Link weight assignment and loop-free routing table update for link state routing protocols in energy-aware internet. *Future Generation Computer Systems*, 28(2):437–445, 2012.

[40] A. Abbagnale, F. Cuomo, and S. Papagna. Esol: Energy saving in the internet based on occurrence of links in the routing paths. In *Proc. of World of Wireless Mobile and Multimedia Networks 2011 (WoWMoM) (workshop SUSATINET), Lucca, Italy, June 20-24, 2011*, 2011.

[41] F. Cuomo, A. Abbagnale, A. Cianfrani, and M. Polverini. Keeping the connectivity and saving the energy in the internet. In *Computer Communications Workshops (INFOCOM WKSHPS), 2011 IEEE Conference on*, pages 319–324. IEEE, 2011.

[42] A.P. Bianzino, L. Chiaraviglio, and M. Mellia. Distributed algorithms for green ip networks. In *Computer Communications Workshops (INFOCOM WKSHPS), 2012 IEEE Conference on*, pages 121–126.

[43] A.P. Bianzino, L. Chiaraviglio, M. Mellia, and J.L. Rougier. Grida: Green distributed algorithm for energy-efficient ip backbone networks. *Computer Networks*, 2012.

[44] Stefano Avallone and Giorgio Ventre. Energy efficient online routing of flows with additive constraints. *Computer Networks*, 56(10):2368 – 2382, 2012. ¡ce:title¿Green communication networks¡/ce:title¿.

[45] F. Idzikowski, L. Chiaraviglio, and F. Portoso. Optimal design of green multi-layer core networks. In *Proc. of the e-Energy 2012*, page 15. ACM.

[46] A. Capone, D. Corti, L. Gianoli, and B. Sansó. An optimization framework for the energy management of carrier ethernet networks with multiple spanning trees. *Computer Networks*, 56(17):3666 – 3681, 2012.

[47] E. Gelenbe and T. Mahmoodi. Energy-aware routing in the cognitive packet network. In *Proc. of International Conference on Smart Grids, Green Communications, and IT Energy-aware Technologies*.

[48] K.H. Ho and C.C. Cheung. Green distributed routing protocol for sleep coordination in wired core networks. In *Networked Computing (INC), 2010 6th International Conference on*, pages 1–6. IEEE, 2010.

[49] R. Bolla, R. Bruschi, and M. Listanti. Enabling backbone networks to sleep. *Network, IEEE*, 25(2):26–31, 2011.

[50] L. Chiaraviglio and A. Cianfrani. On the effectiveness of sleep modes in backbone networks with limited configurations. In *Software, Telecommunications and Computer Networks (SoftCOM), 2012 20th International Conference on*, pages 1 –6, sept. 2012.

[51] S. Orlowski, R. Wessäly, M. Pióro, and A. Tomaszewski. Sndlib 1.0 survivable network design library. *Networks*, 55(3):276–286, 2010.

[52] H. Ümit and B. Fortz. Fast heuristic techniques for intra-domain routing metric optimization. In *Proc. of the 3rd International Network Optimization Conference (INOC 2007), Spa, Belgium*, volume 6, 2007.

[53] L. Chiaraviglio, M. Mellia, and F. Neri. Reducing power consumption in backbone networks. In *Proc. of Communications, 2009. ICC'09. IEEE International Conference*, pages 1–6.

[54] I. Ghamlouche, T.G. Crainic, and M. Gendreau. Cycle-based neighbourhoods for fixed-charge capacitated multicommodity network design. *Operations Research*, 51(4):655–667, 2003.

[55] Rocketfuel Project [Online]. `www.cs.washington.edu/research/networking/rocketfuel`.

[56] M. Resende and C. Ribeiro. Greedy randomized adaptive search procedures. *Handbook of metaheuristics*, pages 219–249, 2003.

[57] M. Resende and C. Ribeiro. GRASP with path-relinking: Recent advances and applications. *Metaheuristics: Progress as real problem solvers*, pages 29–63, 2005.

[58] M. Pizzonia and M. Rimondini. Netkit: easy emulation of complex net-

works on inexpensive hardware. In *Proc. of the 4th International Conference on Testbeds and research infrastructures for the development of networks & communities*, pages 1–10. ICST, 2008.

[59] M. Rimondini. Emulation of computer networks with netkit. *Dipartimento di Informatica e Automazione, Roma Tre University, http://www.netkit. org/, RT-DIA-113-2007*, 2007.

[60] O. Bonaventure. Software tools for networking. *Network*, 18(6):4–5, 2004.

[61] Iperf network testing tool [Online]. `http://iperf.sourceforge.net/`.

[62] vnStat console-based network traffic monitor [Online]. `http://humdi.net/vnstat/`.

[63] Abilene network [Online]. `http://sndlib.zib.de/home.action`.