Dynamic power management in energy-aware computer networks and data intensive computing systems

Ewa Niewiadomska-Szynkiewicz[a,b,*], Andrzej Sikora[a,b], Piotr Arabas[a,b], Mariusz Kamola[a,b], Marcin Mincer[a], Joanna Kołodziej[c]

[a] *Institute of Control and Computation Engineering, Warsaw University of Technology, Nowowiejska 15/19, 06-665 Warsaw, Poland*
[b] *Research and Academic Computer Network, Wąwozowa 18, 02-769 Warsaw, Poland*
[c] *Institute of Computer Science, Cracow University of Technology, Warszawska 24, 31-155 Cracow, Poland*

Abstract

The energy awareness is an important aspect of modern networks and computing systems design and management, especially in the case of internet-scale networks and data intensive large scale distributed computing systems. The main challenge is to design and develop novel technologies, architectures and methods that allow to reduce energy consumption in such infrastructures, which is also the main reason for reducing the total cost of running a network. Energy-aware network components as well as new control and optimization strategies may save the energy utilized by the whole system through adaptation of network capacity and resources to the actual traffic load and demands, while ensuring end-to-end quality of service. In this paper, we have designed and developed a two-level control framework for reducing power consumption in computer networks. The implementation of this framework provides the local control mechanisms that are implemented at the network device level and network-wide control strategies implemented in the central control level. We also developed network-wide optimization algorithms for calculating the power status of energy consuming network components and the energy-aware routing for recommended network configuration. Utility and efficiency of our framework have been verified by simulation and by laboratory tests. The test cases were carried on a number of synthetic as well as on real network topologies, giving encouraging results. Thus, we come up with well justified recommendations for energy-aware computer network design, to conclude the paper.

Keywords: energy-aware network, energy-aware routing, traffic engineering, dynamic power management, data intensive computing, network simulation, testbed implementation

---

[*] Corresponding author, phone: +48 22 xxx xx xx

Email addresses: `ens@ia.pw.edu.pl` (Ewa Niewiadomska-Szynkiewicz), `xxx@ia.pw.edu.pl` (Andrzej Sikora), `parabas@ia.pw.edu.pl` (Piotr Arabas), `mkamola@ia.pw.edu.pl` (Mariusz Kamola), `xxx@ia.pw.edu.pl` (Marcin Mincer), `jokolodziej@pk.edu.pl` (Joanna Kołodziej)

Preprint submitted to Future Generation Computer Systems *July 12, 2013*

1. Introduction to Energy-Aware Networks

The growing energy consumption in ICT sector is unquestionable. It is obvious that in order to support new generation network services and infrastructure, network operators and Internet service providers need a large number of more sophisticated network devices able to perform complex operations in a scalable way and assure expected quality of service. This is one of the reasons for the rapid growth of the energy requirements of wired and wireless modern computer networks. Therefore, the energy consumption trends in the next generation networks have been widely discussed and the optimization of total power consumption in today's computer networks has been a considerable research issue [1, 2, 3, 4]. New solutions both in hardware and software have been developed to achieve the desired trade-off between power consumption and the network performance according to the network capacity, current traffic and requirements of the users. The aim is to reduce the gap between the capacity provided by a network for data transfer and the requirements, especially during low traffic periods. In particular, the energy dissipated in a network can be minimized by switching off idle energy consuming components such as routers, line cards, and communication interfaces, and by reducing the speed of processors and link speed. In general, data transfers should be aggregated along as few devices as possible instead of balancing traffic in a whole computer network. Selectively shutting down routers and links in periods of low demand seems to be a good solution for reducing the energy usage due to the fact that typical networks are usually overprovisioned. The techniques developed for keeping the connectivity and saving the energy can be successfully used for energy-efficient dynamic management in LANs (local area networks), WANs (wide area networks) as well as in computing centers.

Various activities and research projects aimed at developing energy-efficient networks and computing devices have been undertaken. Approaches ranging from "green" network devices and architectures to traffic engineering and routing protocols have been developed and investigated. Novel devices equipped with mechanisms for dynamic power management can operate in a number of modes, which differ in the power usage. Two common techniques for dynamic power management are utilized in energy-aware networks: smart standby and dynamic power scaling. *Smart standby* leverages on the concept of introducing idle mode capabilities, i.e., the whole device or its component is automatically switched off when it is idle – there is no data to transmission. *Dynamic power scaling* adopts the capacity (and thus power consumption) of the devices to the current load utilizing *adaptive rate* (AR) or *low power idle* (LPI) techniques. The AR method reduces the energy demands in a network by scaling the processing capabilities of a given device or the transmission or reception speed of the network interface. The LPI method allows reducing the energy requirements by putting the device or its component into a low power mode. While dynamic power scaling approaches often involve deep modifications in the design of software and hardware components of network devices, the smart standby method requires only coordination among networking nodes to carefully re-route the traffic that results from switching off selected devices or their components.

The control framework for resource consolidation and dynamic power management of the whole network through energy-aware routing, traffic engineering and network equipment activity control has been designed and developed by the ECONET consortium [5]. This control system implements algorithms to exploit both smart standby and dynamic power scaling capabilities of network nodes and links. The implementation of the framework provides the local control mechanisms that are implemented in the network devices level and network-wide control strategies implemented in the central control level. In this paper we focus on the optimization algorithms used by the central dispatcher for calculating the optimal energy settings of all components of a network infrastructure and the optimal routing that minimizes the energy consumption, while ensuring all user quality requirements imposed on a network. We have developed several possible formulations of a network energy saving optimization problem with continuous and discrete variables. The work starts from the complete network management problem assuming full routing calculation and energy-aware state (EAS) assignment to all links in a network, stated in terms of binary variables. Due to the numerical complexity of the complete problem formulation we applied some simplifications. Finally, we proposed to employ heuristics to calculate the optimal energy settings of the devices for more realistic size of networks. The developed control scheme has been validated for various network systems and traffic engineering using MPLS (Multiprotocol Label Switching,cf. [6]) and RSVP-TE (RSVP-Traffic Engineering, cf. [7]) protocols through simulations and the testbed implementation. We compared the performance of all proposed solutions due to the energy saving and efficiency, while providing for adequate transmission quality.

The paper is structured as follows. In Section 2 we discuss the selected approaches to power control in computer networks provided in literature. The description of the control framework that can operate in two variants (centralized and hierarchical) is reported in Section 3. In Section 4 the basic formulation of network-wide optimization of energy consumption and its relaxation and transformation to the simpler problem are presented. The scalability of proposed optimization schemes is discussed in Section 5. The results of simulations and experiments in the testbed are presented in Sections 6 and 7. Finally, conclusions are drawn in Section 8.

2. Related Work

The problem of reducing energy consumption of telecommunication networks has been studied in the recent years by many researchers. First the power needs of networks were assessed and some basic models built [8, 9, 10, 11], then some elementary local strategies, using AR and LPI techniques have been proposed – see e.g. [12, 13]. The industry standard implementing some of these ideas is IEEE 802.3az [14] providing energy efficient Ethernet interfaces.

Apart from improving the effectiveness of network equipment itself, it is possible to adopt energy-aware network-wide control strategies and algorithms to manage dynamically the whole network and reduce its power consumption by appropriate traffic engineering and provisioning. Recent studies concerning networks of Internet Service Providers suggest that such approach can significantly decrease energy consumption of a network [1, 3].

The rationale behind this solution is that network load varies periodically and may be predicted with reasonable probability while network resources stay constant. Furthermore, typically infrastructure is to some degree redundant to provide required level of reliability. To mitigate power consumption some parts of network may be switched off or their performance may be decreased during off-peak periods. To attain a multi-commodity flow optimization problem may be formulated and solved. Such a formulation resembles traditional network design problems [15] or QoS (Quality of Service) provisioning tasks found in e.g. [16], [17], but with energy consumed by all components of the network being major part of the performance index. As energy states of particular devices must be computed and flows routed resulting problems include large number of integer or binary variables and may be solved only for networks of limited size so linear models are prefered to reduce complexity [8, 18, 19, 20]. Some authors try to exploit properties of optical transport layer to scale link rates by selectively switching off fibres composing them [21, 22] or even build two level model with IP layer set upon optical devices layer [23]. Similar decomposition may be found in [24], [25] where switching off the idle devices and modulation of the power supply in computational grids is analyzed.

The major drawback is complexity which has roots in NP-completeness of flow problems formulated as MIP (Mixed Integer Programming). Furthermore, energy consumption models are often non-convex making even continuous relaxation difficult to solve and introducing instability of suboptimal solutions [11]. Also, while some authors – e.g. [19] assume only two states of network equipment – active and switched off, it must be remembered that future green network devices will have ability to independently adapt performance of their subcomponents by setting them to one of a number of energy-aware states. Modeling such situation implies not only larger dimensionality of the problem but also more complicated dependencies among subcomponents. Two-level systems mentioned above 2-level systems [8, 23, 21] may be considered as one of ways to tackle this problem. Some improvement may be attained by aggregating parts of infrastructure like in [19] or constraining search space by preselecting links [21, 26] or paths [27, 28] used for data transmission.

Approaches presented above are centralized ones – i.e. they need complete information about the state of the network and typically operate on a single node controlling the whole infrastructure. Distributing energy-aware control allows better integration with existing mechanisms – typically routing protocols e.g. OSPF and MPLS [29, 30, 31, 32] are extended this way. Important profit of close cooperation with signaling protocols is that the observed state of the network may be used to estimate flows [30] and to reconstruct the traffic matrix [26]. The most important drawback of distributed algorithms is that using reatively simple heuristics for decomposition of fully defined centralized tasks is usually impossible due to interlocking constraints and nonconvexities. An interesting approach, although still giving results far from optimal is substituting energy optimization with specific topology optimization proposed in [33] and [34] which allows easy decomposition and integration with routing algorithms.

Most of the abovementioned solutions were tested by computation or simulation experiments, while real equipment was used mainly when collecting data or building device models – see e.g. [1, 3]. The cause is difficulty in implementing new algorithms in closed devices like routers, optical multiplexers, etc. Some work was, however already done on implementing energy aware mechanisms into open software platforms (e.g. Linux, Zebra/Quagga suite) [10, 13]. The solution adopted within ECONET project is to build testbed network of standard PCs running such a software. As far only one more Linux based testbed is known to exist – see. [35].

3. Control Framework for Energy-Aware Networks

Various optimization and control strategies can be applied for dynamic power management and energy saving in computer networks. We have designed and developed a general control scheme employing energy-aware traffic management and modulation of the energy consumption of network devices. In our approach we assume that network devices and their components can operate in different energy-aware states (EAS), which differ in the power usage. The ACPI (Advanced Configuration and Power Interface) specification [36] defines a number of energy-aware states attained via voltage and clock frequency scaling and idle states in which the processor is in the standby mode. In our control system all routers, line cards and communication ports can operate in active, sleeping and switched off (idle) energy states that are related to the application of standby and dynamic power scaling techniques. All these states are defined as power settings and corresponding throughput. Figure 1 depicts EASs for a router, a card and a port assumed in our control scheme. It can be seen that we consider only two states for a router and a card (active with full power and deep sleep) and a number of active states for communication ports and corresponding links. We tackle the energy saving in a network by putting into low energy states or deactivating selected routers, line cards and ports. The implementation of our control framework requires the presence of a central control unit. The decisions about activity and power status of network equipment are determined by solving the problem of minimizing the energy utilized by the whole system. The optimal network performance is calculated based on the global data about the network topology, load measures and expected demands for a network throughput. The energy optimization problem is formulated as a mathematical programming problem with objective function, various constraints and control parameters. Figures 2 and 3 present the architectures of two variants of the implementation of our control framework. Both of these variants are composed of four main components: OAM, NCP (cNCP or hNCP), LCP and GAL. The objective of OAM (Monitoring and Operation Administration & Management) is to provide tools for network monitoring and configuration in terms of trade-off between energy consumption and network performance. Moreover, this component plays the role of middleware between network equipment. It supports MPLS TE (Multiprotocol Label Switching Traffic Engineering) technology. The significant component of our system – GAL (Green Abstraction Layer) is the standard interface between monitoring and control plane layers and hardware for exchanging data regarding the power status of each network device and all its components. The goal is to hide the implementation details of energy saving techniques, as well as to provide standard interfaces between all components of a system and energy-aware technologies. Therefore, GAL transforms the outcome of LCPs into power-management configuration of a given component of the device – a selected router, card or

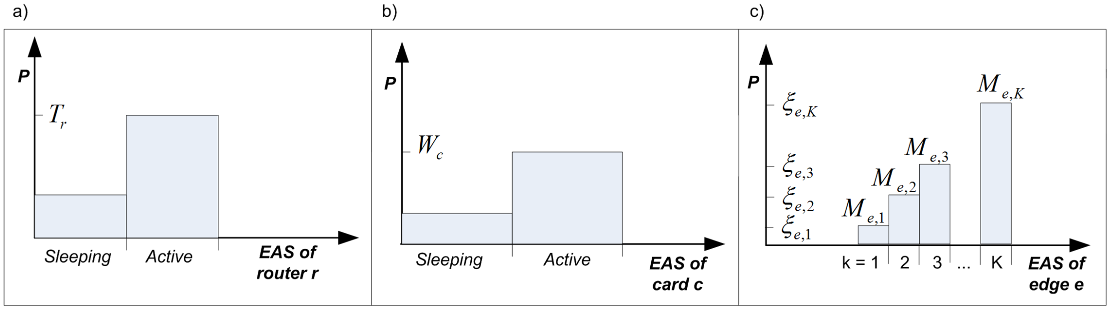

Figure 1: The energy-aware states of router, card, link (K – a number of states).

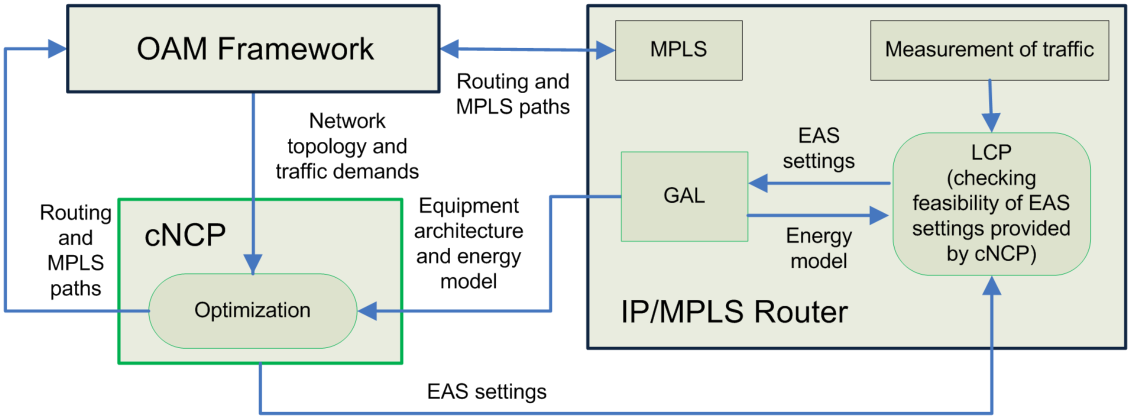

Figure 2: The architecture of the control framework; centralized scheme.

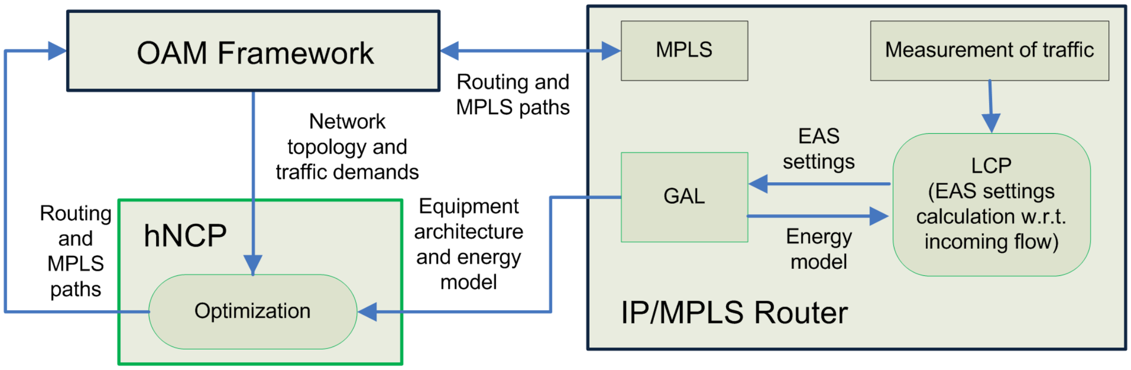

Figure 3: The architecture of the control framework; hierarchical scheme.

port is switched into the recommended energy-aware state. The detailed description of GAL is provided in [37] and [5]. The control plane layer is formed by NCP and LCP components. Each LCP (*Local Control Policy*) component is implemented in a given device and locally calculates the energy efficient configuration of this device due to the incoming traffic load measured by its interfaces. The LCP implements adaptive rate and low power idle techniques for power management. Several technologies for dynamic configuration setting of the energy-saving capabilities of the network devices have been developed by the ECONET consortium [5]. Finally, the NCP (*Network-wide Control Policy*) denotes a central unit whose goal is to optimize the performance of a whole network to reduce power consumption. The optimization problem is formulated and solved for a given network, taking into account its topology and expected demands imposed on a network. The proposed formulations are described in the following section.

Two variants of a network-wide control (NCP) implementation, i.e., cNCP – centralized (Fig. 2) and hNCP – hierarchical (Fig. 3) have been investigated. They produce different outcomes. In the centralized scenario (cNCP) all decisions regarding the operation of a network and configurations of all devices are calculated by the optimization algorithm executed by the central unit. The suggested power statuses of network devices are then sent to LCP units. Furthermore, the routing tables for the MPLS protocol for recommended network configuration are provided to the OAM framework. Hence, in this scenario, the activity of each LCP unit is reduced to comply with the recommendations calculated by the cNCP, taking into account constraints related to current local load and incoming traffic.

In the hierarchical scenario (hNCP) the central unit does not directly force the energy configuration of the devices. The outcome of the hNCP is reduced to routing tables for the MPLS protocol that are used for routing current traffic within a given network. The objective of the LCP algorithm is to optimize the energy setting of each component of a given device in order to achieve the desired trade off between energy consumption and performance according to the incoming traffic load measured by the OAM framework. The remainder of this paper is devoted to the description and evaluation of two optimization schemes that can be applied in the cNCP and hNCP components of our control framework for low energy consumption network.

4. Network-Wide Energy Saving Optimization Problems

We consider various formulations of a network-wide energy saving problem. Starting from an exact MIP formulation, including complete routing and energy-state decisions, we introduced in [38] subsequent simplifications in order to obtain a continuous problem formulation. Therefore, we can define four formulations for energy optimization on network level:

LNPb - Link-Node Problem: a complete network management problem stated in terms of binary variables assuming full routing calculation and energy state assignment to all devices and links in a network;

LPPb - Link-Path Problem: a formulation stated in terms of binary variables assuming predefined paths (simplification of *LNPb*);

LNPc - Link-Node Problem: a complete network management problem stated in terms of continuous variables assuming full routing calculation;

LPPc - Link-Path Problem: a formulation stated in terms of continuous variables assuming predefined paths (simplification of *LNPc*).

In the above formulations the total power utilized in a network for finalizing all required operations is minimized and end-to-end QoS is ensured. All possible energy saving decisions are directly specified, together with decisions concerning traffic assignment to particular links. The main idea is to concentrate network traffic on a minimal subset of network components and put the working devices into such EASs that minimize the energy consumption in a whole network.

Unfortunately, the enumerated approaches are of limited use in real networks. Both *LNPb* and *LPPb* fall into the group of the *NP*-complete problems. The problem complexity strongly grows with the size of a network. Although *LPPb* is easier to solve due to smaller number of constraints, but is still too complex for medium-size networks. Thus, it is difficult to find sufficiently efficient algorithm for determining optimal performance of real networks.

However, the formulations with continuous decision variables (*LNPc* and *LPPc*) are much easier to solve but other problems appear. In case of these approaches the problem is to define an appropriate objective function to be optimized. In practice, it would be very difficult to determine a cost function that may properly account for the costs of operating the routers, and of keeping active or inactive the line cards and communication ports. Moreover, the final optimization problem is nonconvex.

After a preliminary examination of *LNP* and *LPP* schemes, defined both for binary and continuous variables we realized that the only acceptable solution was to develop and employ efficient heuristics to solve the energy saving optimization problem. A network management problem based on a heuristic approach that leads to a continuous optimization was proposed in [20]. Both the original formulation of *LNPb* and its transformation to the continuous problem are described and discussed in the following subsection.

4.1. Network Description

Let us consider a computer network formed by the following components: R routers ($r = 1, \dots, R$), C line cards ($c = 1, \dots, C$) and P communication ports ($p = 1, \dots, P$). The hierarchical representation of a router is assumed, i.e., each router is equipped with a number of line cards, and each card contains a number of communication ports. All pairs of ports from different cards are connected by E links ($e = 1, \dots, E$). All network components can operate in K energy states (EASs) defined as power settings, and labeled with $k = 1, \dots, K$. In the optimization problems stated in this section each router and card can operate in two energy states (EAS): active and sleeping ($K = 2$) and all communication ports can operate in at least two states ($K \geq 2$). Two ports connected by the e-th link are in the same state k. M_{ek} and ξ_{ek} denote respectively, the throughput and the power consumption of the link e in the state k, W_c and T_r denote fixed power levels associated to the card c and the router r, respectively D stands for a number of demands ($d = 1, \dots, D$) imposed on the network and transmitted by means of flows allocated to given MPLS paths under QoS requirements. V_d denotes the volume of the demand d that is associated with a link connecting two ports: s_d and t_d – ports of the source and the destination nodes for the demand d.

4.2. LNPb: Link-Node Problem

Given the notation from section 4.1, we can formulate the energy saving optimization problem. The aim is to minimize the total power utilized by network components for finalizing all network operations while ensuring end-to-end QoS. We begin with the basic link-node formulation (*LNPb*) – a complete network management problem stated in terms of binary variables assuming energy state assignment to all routers, line cards and communication ports in a network and full routing calculation for recommended network configuration.

$$\min_{x_c, y_{ek}, z_r, u_{ed}} \left[F_{LNPb} = \sum_{r=1}^{R} T_r z_r + \sum_{c=1}^{C} W_c x_c + \sum_{e=1}^{E} \sum_{k=1}^{K} \xi_{ek} y_{ek} \right], \tag{1}$$

subject to the constraints:

$$\forall_{e=1,\dots,E} \quad \sum_{k=1}^{K} y_{ek} \le 1, \tag{2}$$

$$\forall_{\substack{d=1,\dots,D,\\ c=1,\dots,C}} \quad \sum_{p=1}^{P} l_{cp} \sum_{e=1}^{E} a_{ep} u_{ed} \le x_c, \tag{3}$$

$$\forall_{\substack{d=1,\dots,D,\\ c=1,\dots,C}} \quad \sum_{p=1}^{P} l_{cp} \sum_{e=1}^{E} b_{ep} u_{ed} \le x_c, \tag{4}$$

$$\forall_{\substack{r=1,\dots,R,\\ c=1,\dots,C}} \quad g_{rc} x_c \le z_r, \tag{5}$$

$$\forall_{\substack{d=1,\dots,D,\\ r=1,\dots,R,\\ p=s_d}} \sum_{c=1}^{C} g_{rc} l_{cp} \sum_{e=1}^{E} a_{ep} u_{ed} - \sum_{c=1}^{C} g_{rc} l_{cp} \sum_{e=1}^{E} b_{ep} u_{ed} = 1, \tag{6}$$

$$\forall_{\substack{d=1,\dots,D,\\ r=1,\dots,R\\ p\neq t_d, p\neq s_d}} \sum_{c=1}^{C} g_{rc} \sum_{p=1}^{P} l_{cp} \sum_{e=1}^{E} a_{ep} u_{ed} - \sum_{c=1}^{C} g_{rc} \sum_{p=1}^{P} l_{cp} \sum_{e=1}^{E} b_{ep} u_{ed} = 0, \tag{7}$$

$$\forall_{\substack{d=1,\dots,D,\\ r=1,\dots,R,\\ p=t_d}} \sum_{c=1}^{C} g_{rc} l_{cp} \sum_{e=1}^{E} a_{ep} u_{ed} - \sum_{c=1}^{C} g_{rc} l_{cp} \sum_{e=1}^{E} b_{ep} u_{ed} = -1, \tag{8}$$

$$\forall_{e=1,\dots,E} \quad \sum_{d=1}^{D} V_d u_{ed} \le \sum_{k=1}^{K} M_{ek} y_{ek}, \tag{9}$$

where variables and constants used in above formulas denote: $z_r = 1$ if the router r is used for data transmission (0 otherwise), $x_c = 1$ if the card c is used for data transmission (0 otherwise), $y_{ek} = 1$ if the link e is in the state k (0 otherwise), $l_{cp} = 1$ if the port p belongs to the card c (0 otherwise), $u_{ed} = 1$ if the path d belongs to the link e (0 otherwise), $a_{ep} = 1$ if the link e is outgoing from the port p (0 otherwise), $b_{ep} = 1$ if the link e is incoming to the port p (0 otherwise), $g_{rc} = 1$ if the card c belongs to the router r (0 otherwise).

In the problem defined as above the conditions (2) assure that each link can be in one energy-aware state, the constraints (3)-(5) determine the number of routers and cards used for data transmission. The constrains (6)-(8) are formulated according to Kirchhoff's law applied for source, transit and destination routers, and the constraint (9) assures that the flow will not exceed the capacity of a given link. The branch-and-bound algorithm can be used to solve (1)-(9) problem.

As stated previously the *LNPb* strategy is an *NP*-complete challenging optimization task, and can appear to be far too difficult to be solved for even medium-size network in a reasonable time due to numerous binary variables. Fig. 4 shows the results of computational *LNPb* complexity estimation for some example networks. The estimated number of variables and number of constraints for varying number of network nodes are depicted.

The results of estimations presented in Fig. 4 show that the complexity grows rapidly with the number of network components. In particular, the number of constraints grows faster than the number of variables. Therefore, one cannot expect to find time efficient algorithm for solving the

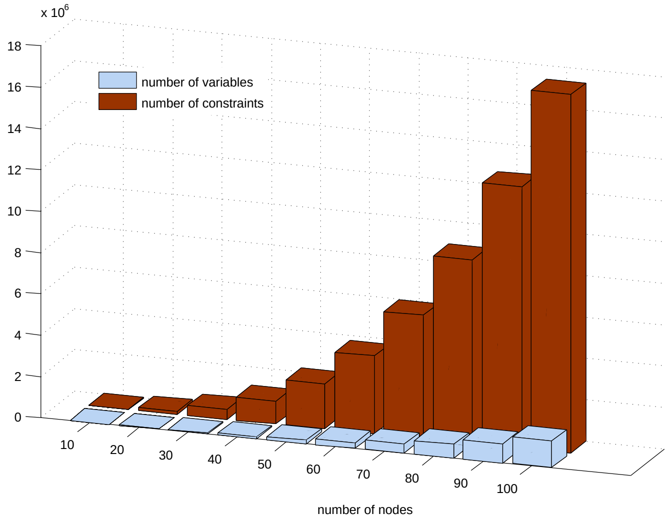

Figure 4: Estimation of *LNPb* computational complexity for various dimensions of a network.

problem (1)-(9) for larger networks. The widely used direction to reduce complexity of a network optimization problem and to develop an algorithm to solve it for more realistic network sizes is to relax the original problem and to employ heuristics to solve it. Hence, we relaxed the basic link-node problem formulation (1)-(9) and transformed it into the problem with continuous variables. Next, we developed an efficient heuristic algorithm to solve such stated optimization task.

4.3. LNRP: Link-Node Relaxed Problem

In the link-node relaxed problem the energy consumption and throughput utilization of the link e in the state k are described in the form of incremental model. The current values of ξ_{ek} and M_{ek} are calculated as follows: $\xi_{ek} = pow_e(k) - pow_e(k-1)$ and $M_{ek} = load_e(k) - load_e(k-1)$; where respectively $pow_e(k)$ denotes power used by the link e in the state k and $load_e(k)$ denotes load of the link e in the state k. Due to the presented relaxation we can transform a linear optimization problem with binary variables to a linear optimization problem with continuous variables x, y, z, u.

$$\min_{x_c, y_{ek}, z_r, u_{ed}} \left[F_{LNRP} = \sum_{r=1}^{R} T_r z_r + \sum_{c=1}^{C} W_c x_c + \sum_{e=1}^{E} \sum_{k=1}^{K} \xi_{ek} y_{ek} \right], \tag{10}$$

subject to the constraints (5)-(9) and additional constraints:

$$\forall_{e=1,\dots,E} \quad y_{e1} \geq y_{ei} \geq \dots \geq y_{eK}, \tag{11}$$

$$\forall_{\substack{e=1,\dots,E,\\ k=1,\dots,K,;;c=1,\dots,C}} \quad \sum_{p=1}^{P} l_{cp} a_{ep} y_{ek} \leq x_c, \tag{12}$$

$$\forall_{\substack{e=1,\dots,E,\\ k=1,\dots,K,;;c=1,\dots,C}} \quad \sum_{p=1}^{P} l_{cp} b_{ep} y_{ek} \leq x_c, \tag{13}$$

In the *LNRP* formulation a given link can operate in more than one energy-aware state. Therefore, the constraint (11) for utilized throughput in various states was added. Moreover, the utilized throughput in subsequent states are sorted. The constraints (12) and (13) force binary values of variables z_r, x_c in case when y_{ek} takes a binary value.

4.4. Link Node Heuristic Algorithm

A two-stage algorithm that employs the heuristic is proposed to solve the problem (10)-(13). The algorithm operates as follows. In the first stage, the preliminary solution is determined by any linear solver. In the second stage, the original problem is modified and calculations are repeated for this modified formulation. The optimization problem (10)-(13) is repetitively modified and solved until all decision variables take binary values, i.e., 0 or 1, and the calculations are terminated. The pseudocode of the algorithm **Algorithm 1** is presented below.

Algorithm 1 Link node heuristic algorithm (LNHA).

```
LNRP denotes the optimization problem (10)-(13). Set the initial values of $\hat{z}_r$, $\hat{x}_c$, $\hat{u}_{ed}$ and $\hat{y}_{ek}$
while $(\hat{z}_r \in (0,1))$ || $((\hat{x}_c \in (0,1))$ || $(\hat{u}_{ed} \in (0,1))$ || $(\hat{y}_{ek} \in (0,1))$ do
    Calculate $\hat{z}_r$, $\hat{x}_c$, $\hat{u}_{ed}$ and $\hat{y}_{ek}$ (use a linear solver to the LNRP)
    if $(\exists \hat{y}_{ek} \in (0,1))$ then
        Create a subset $S_E^* \subset S_E$ of all links $e$ for which $\hat{y}_{ek} \in (0,1)$
        Create a subset $S_{E_{min}} \subset S_E^*$ of all links that operate in the lowest energy-aware state $k^*$
        Select from $S_{E_{min}}$ a link $e^* = arg\{\min_e \hat{y}_{ek}\}$
        Remove $\hat{y}_{e^*k^*}$ from the set $S_E^*$
        Extend the problem LNRP with the constraint $y_{e^*k^*} = 1$
    end if
    if $(\exists \hat{u}_{ed} \in (0,1))$ then
        Create a subset $S_E^{**} \subset S_E$ of all links $e$ for which $\hat{u}_{ed} \in (0,1)$
        Select from $S_E^{**}$ a link $e^{**} = arg\{\max_e \hat{u}_{ed}\}$
        Remove $\hat{u}_{e^{**}d}$ from the set $S_E^{**}$
        Extend the problem LNRP with the constraint $u_{e^{**}d} = 1$
    end if
end while
```

Both *LNPb* and *LNRP* energy saving optimization problem formulations were implemented in our control framework for energy-aware networks. They can support cNCP and and hNCP control strategies.

5. Numerical Validation and Performance Evaluation

Two optimization problems: the basic link-node with binary variables *LNPb* and its relaxation stated in terms of continuous variables *LNRP* were formulated, implemented and solved for small-, medium- and large-size network topologies. The commonly used branch-and-bound solver incorporated from the open source library Lp_solve [39] was applied to the *LNPb* problem while the LNHA algorithm combined with Lp_solve was used to solve the *LNRP* one. To model the power consumption of all network components (routers, cards and ports), we considered power requirements of network devices described in [40]. We assumed that routers and cards could operate in two energy states ($k = 1, 2$): *active* and *sleeping*. Each port could operate in five states ($k = 1, 2, 3, 4, 5$), which differed in power requirements and corresponding throughput. In all experiments we took the overbooking factor equal to 75%.

All tests were done on Intel Core i7-3612QM CPU, 2.1 GHz, 8GB RAM. The aim of the experiments was to check and compare the performance, efficiency and scalability of *LNPb* and *LNRP* strategies to energy saving in computer networks.

5.1. Scenario Description

Net A. The first testing scenario was a small-size synthetic network *Net A* composed of 6 routers connected by 20 links (Fig. 5). The power consumption of each router and card in the *active* and *sleeping* states, and the throughput of each link e and the power consumption in the k-th energy-aware state of each communication port are collected in Tables 1 and 2. The numerical complexities of the *LNPb* and *LNRP* optimization tasks, respectively for number of demands D=3 and D=13 are the following:

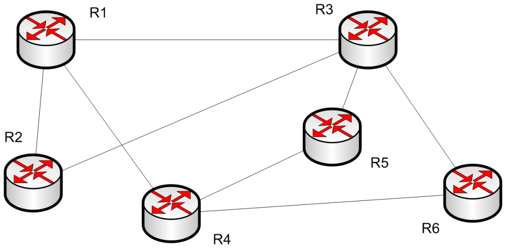

Figure 5: A synthetic network topology *Net A*.

Table 1: A router and a line card: power consumption in *active* and *sleeping* states.

Device	*active* state [W]	*sleeping* state [W]
Router	1 900	100
Line card	90	10

Table 2: A communication port: power consumption and corresponding throughput.

State (k)	1	2	3	4	5
Power consumption (ξ_{ek}) [W]	16	32	48	64	80
Throughput (M_{ek}) [Mb/s]	200	400	600	800	1000

- *LNPb* – D=3: number of variables = 176, number of constraints = 176; D=13: number of variables = 376, number of constraints = 428.
- *LNRP* – D=3: number of variables = 176, number of constraints = 189; D=13: number of variables = 376, number of constraints = 354.

Net B. The second testing scenario was a small-size network *Net B* composed of 12 routers connected by 28 links (Fig. 6). The power consumption of all network components in different states and corresponding throughput are collected in tables 1 and 2. The numerical complexities of the *LNPb* and *LNRP* optimization tasks for number of demands $D = 3$, $D = 6$ and $D = 9$ are the following:

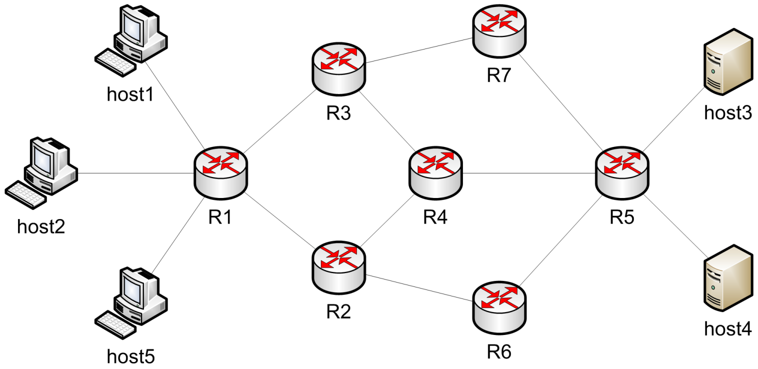

Figure 6: A synthetic network topology *Net B*.

- *LNPb* – $D = 3$: number of variables = 248, number of constraints = 242; $D = 6$: number of variables = 332, number of constraints = 350; $D = 9$: number of variables = 416, number of constraints = 457.
- *LNRP* – $D = 3$: number of variables = 248, number of constraints = 354; $D = 6$: number of variables = 332, number of constraints = 462; $D = 9$: number of variables = 416, number of constraints = 569.

Net C. The third testing scenario was a synthetic network *Net B* composed of 12 routers connected by 42 links (Fig. 7). The power consumption of all network components in different states and corresponding throughput are collected in tables 1 and 2. The numerical complexities of the *LNPb* and *LNRP* optimization tasks for number of demands D=21 are the following:

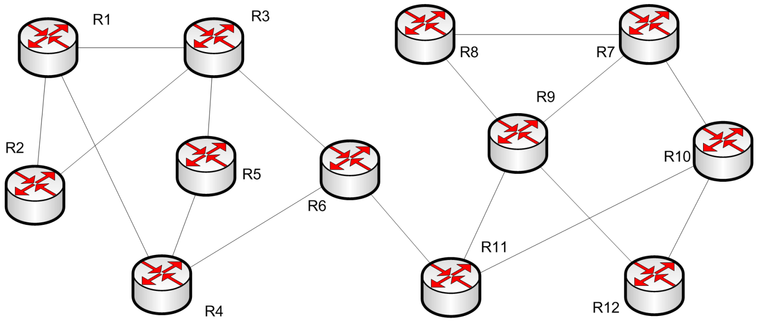

Figure 7: A synthetic network topology *Net C*.

- *LNPb*: number of variables = 1124, number of constraints = 1289.
- *LNRP*: number of variables = 1124, number of constraints = 1956.

Net D. The last testing scenario was an access/metropolitan segment of a traditional telecom operator network used as a benchmark network by many researchers (see [30]). Fig. 8 presents this network topology. It is formed of 21 routers (13 transit, 8 access) connected by 78 links and 1 peering node. The access routers denoted source and destination nodes. The objective of the transit nodes is to perform traffic switching. The peering node provides access to the Internet. The power consumptions of all types of routers and card in the *active* and *sleeping* states are collected in Table 3. Two types of links are considered: *linkT* (thin line in Fig. 8) and *linkB* (bold line in Fig. 8). The throughput and power usages in the *k* state are collected in Table 4. The numerical complexities of the *LNPb* and *LNRP* optimization tasks for number of demands $D = 16$ and $D = 72$ are the following:

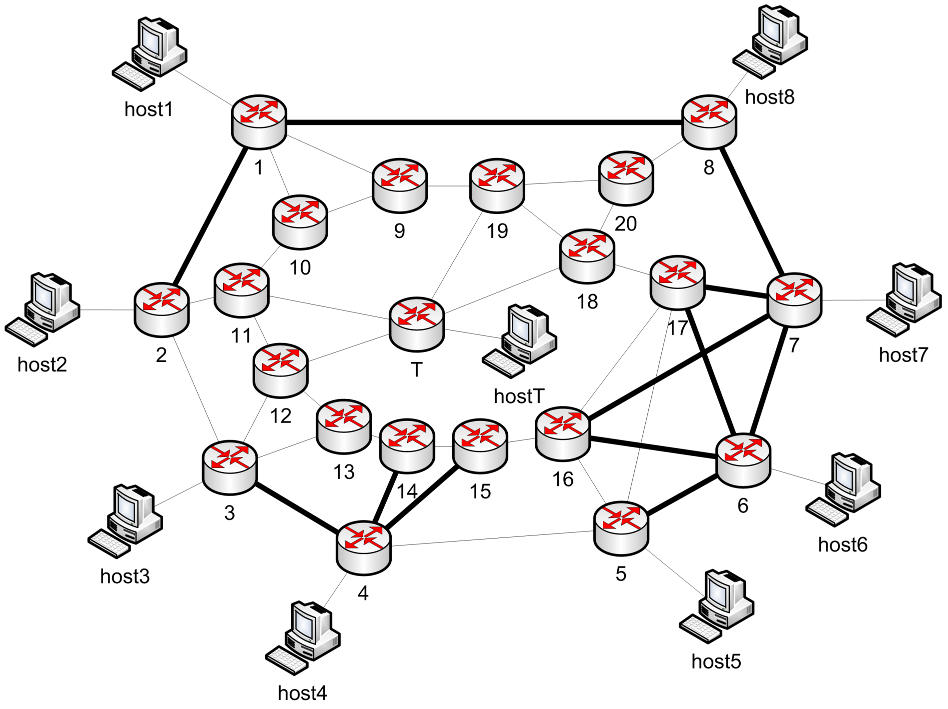

Figure 8: A network topology from a telecom operator *Net D*.

Table 3: A router and a line card: power consumption in *active* and *sleeping* states.

Device	*active* state [W]	*sleeping* state [W]
Access router	1 000	100
Transit router	3 000	100
Peering router	10 000	100
Line card	90	10

- *LNPb* – D=16: number of variables = 1680, number of constraints = 1566; D=72: number of variables = 6048, number of constraints = 5094.

Table 4: A communication port: power consumption and corresponding throughput.

Type of a link *linkT*					
State (k)	1	2	3	4	5
Power consumption (ξ_{ek}) [W]	20	40	60	80	100
Throughput (M_{ek}) [Gb/s]	2	4	6	8	10
Type of a link *linkP*					
	state 1	state 2	state 3	state 4	state 5
Power consumption (ξ_{ek}) [W]	1020	1040	1060	1080	1100
Throughput (M_{ek}) [Gb/s]	2	4	6	8	10

- *LNRP* – D=16: number of variables = 1680, number of constraints = 1956; D=72: number of variables = 6048, number of constraints = 5544.

5.2. Performance Evaluation: Small-Size Networks

The first series of experiments were performed for small-size networks: *Net A* (Fig. 5) and *Net B* (Fig. 6), *LNPb* and *LNPb* energy saving optimization schemes. The results of calculations performed for the *Net A* network and various number of demands D (D=3, D=5, D=7, D=13) are collected in Tables 5 – 7. Table 5 collects the values of calculated optimal energy states and corresponding throughput of links used for transmitting data in a network for the number of demands $D = 13$. Table 6 presents the calculated MPLS routing for $D = 13$. Table 7 shows the number of network devices (routers, cards and links) used for data transmission when routing was calculated, respectively by solving *LNPb* and *LNRP* problems. Moreover, the reductions of energy consumption w.r.t. the total power used by fully loaded network (all devices operating in the highest energy states) and times of calculations for both approaches are presented.

Next, we performed calculations for the *Net B* network. The summary of the results is presented in Table 8.

From the tables 5 – 8 we can observe that both approaches with complete and relaxed optimization problem formulation give similar results. In some cases the cost function is slightly better for the *LNPb*. Therefore, for small-size networks we can recommend to use the *LNPb* method. Unfortunately, the application of this method is limited in case of real networks. The complexity of the problem (1)-(9), and the calculation time grow dramatically with the number of demands. Even in case of a network with only a few nodes and number of demands greater than ten the computation time is noticeable. It can be significantly reduced by applying the *LNRP* problem formulation and heuristic algorithm to solve it.

5.3. Medium-size Networks

The next testing scenario was the medium-size synthetic network *Net C* (Fig. 7) with number of demands $D = 21$. Our experiments showed that complete problem formulation *LNPb* for the *Net C* network and using branch-and-bound to solve this problem involves very high computation overhead; we could not find the solution in a reasonable time. The results confirmed that the *LNP* approach is impractical for even medium-size networks. Therefore, we applied the *LNRP* scheme

Table 5: Energy states of links (*Net A*); methods *LNPb* and *LNRP*, $D = 13$.

Link	Method *LNPb*			Method *LNRP*		
(router/card)	State (k)	Throughput [Mb/s]	Power [W]	State (k)	Throughput [Mb/s]	Power [W]
A1/2 → A2/2	1	200.0	16.0	-	-	-
A1/2 → A3/1	2	400.0	32.0	2	400.0	32.0
A1/2 → A4/1	2	400.0	32.0	2	400.0	32.0
A2/2 → A4/1	2	400.0	32.0	2	400.0	32.0
A2/2 → A3/1	2	400.0	32.0	2	400.0	32.0
A2/2 → A1/2	1	200.0	16.0	-	-	-
A3/1 → A1/2	2	400.0	32.0	2	400.0	32.0
A3/1 → A2/2	2	400.0	32.0	2	400.0	32.0
A3/2 → A5/1	1	200.0	16.0	-	-	-
A4/2 → A5/1	1	200.0	16.0	1	200.0	16.0
A4/2 → A6/1	3	600.0	48.0	3	600.0	48.0
A4/1 → A1/2	2	400.0	32.0	2	400.0	32.0
A4/1 → A2/2	2	400.0	32.0	2	400.0	32.0
A5/1 → A3/2	1	200.0	16.0	-	-	-
A5/1 → A4/2	1	200.0	16.0	1	200.0	16.0
A6/1 → A4/2	3	600.0	48.0	3	600.0	48.0

Table 6: MPLS routing (*Net A*); *LNPb* and *LNRP*, $D = 13$.

Routing (*LNPb*)	Routing (*LNRP*)
A1/2 → A4/1 → A6/1	A1/2 → A4/1 → A6/1
A1/2 → A2/2	A1/2 → A3/1 → A2/2
A1/2 → A4/1 → A6/1	A1/2 → A4/1 → A6/1
A1/2 → A4/1 → A5/1	A1/2 → A3/1 → A2/2 → A4/1 → A5/1
A2/2 → A4/1 → A6/1	A2/2 → A3/1 → A1/2 → A4/1 → A6/1
A2/2 → A4/1 → A6/1	A2/2 → A4/1 → A6/1
A2/2 → A4/1 → A5/1	A2/2 → A4/1 → A5/1
A6/1 → A4/2 → A2/2	A6/1 → A4/2 → A2/2
A3/2 → A5/1 → A4/2 → A6/1	A3/1 → A1/2 → A4/1 → A6/1
A2/2 → A4/1	A2/2 → A4/1
A1/2 → A4/1 → A6/1	A1/2 → A4/1 → A6/1
A1/2 → A2/2	A1/2 → A3/1 → A2/2
A1/2 → A4/1 → A6/1	A1/2 → A4/1 → A6/1

with algorithm LNHA for power management in the *Net C*. The results of calculations are collected in Table 9.

Table 7: Routers, cards and links used for data transmission, reduction of energy consumption and time of calculations (*Net A*).

	LNPb				*LNRP*			
Demands	$D = 3$	$D = 5$	$D = 7$	$D = 13$	$D = 3$	$D = 5$	$D = 7$	$D = 13$
Active routers	4	5	5	6	4	5	5	6
Active cards	5	6	6	7	5	6	6	7
Active links	10	8	10	16	6	10	10	12
Power reduction [W]	5754	3668	3636	1550	5754	3668	3636	1486
Power reduction [%]	41.4	26.4	26.2	11.2	41.4	26.4	26.2	10.7
Time [s]	0.031	0.125	0.327	31.465	0.032	0.078	0.094	0.140

Table 8: Active routers, cards and links, reduction of energy consumption and time of calculations (*Net B*).

	LNPb			*LNRP*		
Demands	$D = 3$	$D = 6$	$D = 9$	$D = 3$	$D = 6$	$D = 9$
Active routers	8	8	9	8	8	9
Active cards	8	8	9	8	8	9
Active links	14	14	16	14	14	16
Power reduction [W]	9368	9368	7346	9368	9368	7346
Power reduction [%]	35.9	35.9	28.1	35.9	35.9	28.1
Time [s]	0.016	0.016	0.089	0.031	0.047	0.058

Table 9: Active routers, cards and links, reduction of energy consumption and time of calculations (*Net C*).

Demands	$D = 21$
Active routers	10
Active cards	14
Active links	22
Power reduction [W]	7188
Power reduction [%]	34.6
Time [s]	0.733

5.4. Large-size Networks

Finally, we performed the experiments for an access/metropolitan segment of an example network of a telecom operator (*Net D*), Fig. 8. The number of demands in our experiments were equal $D = 16$ and $D = 72$. The results of calculations are collected in Table 10. The numerical results presented in Table 10 confirm the observations for the medium-size network (*Net C*) – the calculation time nonlinearly increases with the complexity of the network-wide optimization problem. Summarizing the results of all experiments we can conclude that the simplified formulation of energy saving problem and efficient branch-and-bound implementation supported by heuristics can be successfully applied to medium- and large-size networks.

Table 10: Active routers, cards and links, reduction of energy consumption and time of calculations (*Net D*).

Demands	D=16	D=72
Active routers	11	11
Active cards	11	11
Active links	22	24
Power reduction [W]	50320	48480
Power reduction [%]	58.6	56.5
Time [s]	4.134	33.253

6. Simulation Experiments

Computer simulation is a widely used technique to validate theoretical models of optimization and control systems. Therefore, we validated our control framework for energy-aware computer networks through simulation. We tested the application of centralized variant of our framework (cNCP) to power management in networks with different size and topology. We have designed and developed a green packet-level network simulator for evaluating strategies and algorithms for energy saving in computer networks and checking the compliance of demands.

6.1. Green Network Simulator

The *Green Network Simulator* (GNS) is completely based on OMNeT++ network simulation framework – free software for academic and non-profit use, and widely used platform in scientific and research communities. OMNeT++ (http://www.omnetpp.org/) is an extensible, modular, component-based C++ simulation library and framework, primarily for building wired and wireless network simulators. It offers an Eclipse-based IDE, a graphical runtime environment as well as handy results browser. The GNS simulator extensively uses the open-source communication networks simulation package for OMNeT++ developed by the INET project. The INET framework (http://inet.omnetpp.org/) provides library of models for networking protocols including Ethernet, IP, IPv6, TCP, UDP, MPLS, OSPF, etc., and models of network equipment including routers, switches, line cards, etc. The GNS expands the models of network components implemented in the INET library with green extensions. The schemes for setting energy-aware states, and switching the devices into them were implemented. Moreover, we reformulated the models of routers and cards, and corresponding C++ modules from the INET library to enrich them with basic green abstraction layer (GAL) functionalities developed by the ECONET consortium.

The GNS simulator provides the graphical user interface that was built based on OMNeT++ modules. GUI is organized in a set of nested windows of two kinds: setting and display windows. Setting windows are used to define a scenario of the experiment and introduce all configuration parameters. The other way is to load a network to be simulated from the disc file generated by the optimizer module. Display windows are used to present the simulation results, see Fig. 9.

6.2. Simulation Results

In this paper we present the results of the application of the cNCP control framework to dynamic power management and energy saving in two networks described in Section 5: small-size

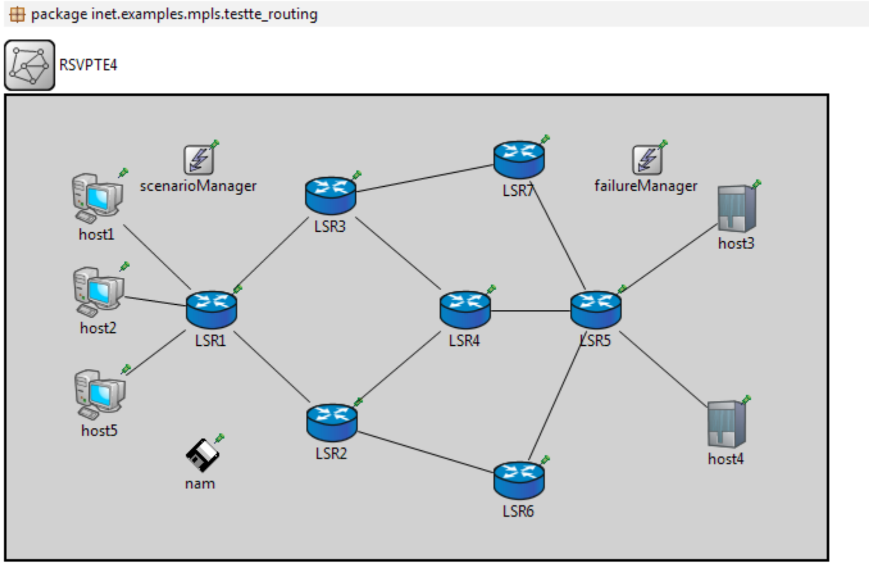

Figure 9: Simulation results presentation.

network *Net B* (Fig. 6) and large-size network *Net D* (Fig. 7). The following number of traffic demands were assumed: $D = 3$, $D = 6$ and $D = 9$ for *Net C* and $D = 16$ for *Net D*. The decisions about power status of devices and routing tables were calculated by solving *LNRP* problem. The goal was to test how power saving strategy reduces total energy consumption and affects the end-to-end QoS. In our experiments we assumed that the size of queue of each router was equal 100 packets (default value in OMNet++). We used traffic UDP generator provided in OMNet++. All experiments were performed using the GNS software.

The cNCP system assumes the complete knowledge of the traffic matrix. Hence, the expected demands have to be estimated. Due to the limited knowledge about future traffic it can be difficult to calculate the forecasts of demands with high accuracy, especially for networks with high volatility of traffic. Therefore, we tested the resilience of our control scheme on the quality of forecasts of demands $\hat{V}_d$. Three series of experiments were performed:

F1 - ***good-quality forecast*** : $\hat{V}_d = V_d$, the forecast of all traffic demands $\hat{V}_d$ was equal to the real traffic within a given network, i.e., used in simulation.

F2 - ***medium-quality forecast*** : $\hat{V}_d = 0.7V_d$ the forecast of traffic demands $\hat{V}_d$ was smaller than the real traffic.

F3 - ***bad-quality forecast*** : $\hat{V}_d = 0.5V_d$ the forecast of traffic demands $\hat{V}_d$ was smaller than the real traffic.

In all experiments we took the overbooking factor equal to 0.75. The results of simulation of 5 seconds of networks *Net B* and *Net D* operation are given in Tables 11 and 12. Both tables present the packets loss caused by the usage of our green control strategy. The simulation results indicate

Forecast	Demands	Total number of packets	Packets loss	Packets lost [%]
F1	$D = 3$	18500	98	0.53
	$D = 6$	23125	144	0.62
	$D = 9$	24050	274	1.14
F2	$D = 3$	26085	1085	4.16
	$D = 6$	32606	1385	4.25
	$D = 9$	33910	3389	9.99
F3	$D = 3$	37000	789	2.13
	$D = 6$	46210	3259	7.05
	$D = 9$	48100	5874	12.2

Table 11: Simulation results for the network *Net B*.

Forecast	Demands	Total number of packets	Packets loss	Packets lost [%]
F1	$D = 16$	57500	162	0.28
F2	$D = 16$	81075	4305	5.31
F3	$D = 16$	115000	9685	8.42

Table 12: Simulation results for the network *Net D*.

that in case of accurate forecasts of expected demands and reasonable overbooking factor we can reduce the total power consumption while ensuring QoS (packet loss is very small). It is obvious that a problem appears for bad quality forecast of demands when application of routing tables generated by the optimizer can result in packets loss (see Tables 11 and 12). To achieve the trade-off between power consumption and a network performance according to the traffic load the hNCP control scheme is proposed. This variant of our control framework that incorporates two-level decision process in which the final energy setting of each network device is calculated by the LCP mechanism due to incoming traffic load measured by its interfaces should be more resistant to the variability of a network traffic.

7. Testbed Implementation and Validation

7.1. Testbed Network Description

A testbed was built and set up to facilitate tests of the green technologies in practice. Instead of dedicated network appliances, PCs with routing capabilities were used to forward layer 3 traffic. The PCs are Dell OptiPlex 9010 MT units with Intel Core i7-3720QM processors operating at 2.6 GHz and with 6 MB cache installed. The operating system is 64-bit Fedora Linux, release 18. Due to various hardships with MPLS patches to Linux kernels, policy routing was used for establishing dedicated paths between network nodes.

The testbed network topology (further referred to as *Net MAN*) is presented in Fig. 10. It is small-size network formed of 7 routers connected by 10 links, and roughly corresponds to the core of *WARMAN* — NASK Warsaw Metropolitan Area Network used mostly by academia and research institutions. The nodes are numbered and marked with the names of the respective NASK POPs. The software routers were equipped with 4-port and 1-port 1 Gbps network cards to support

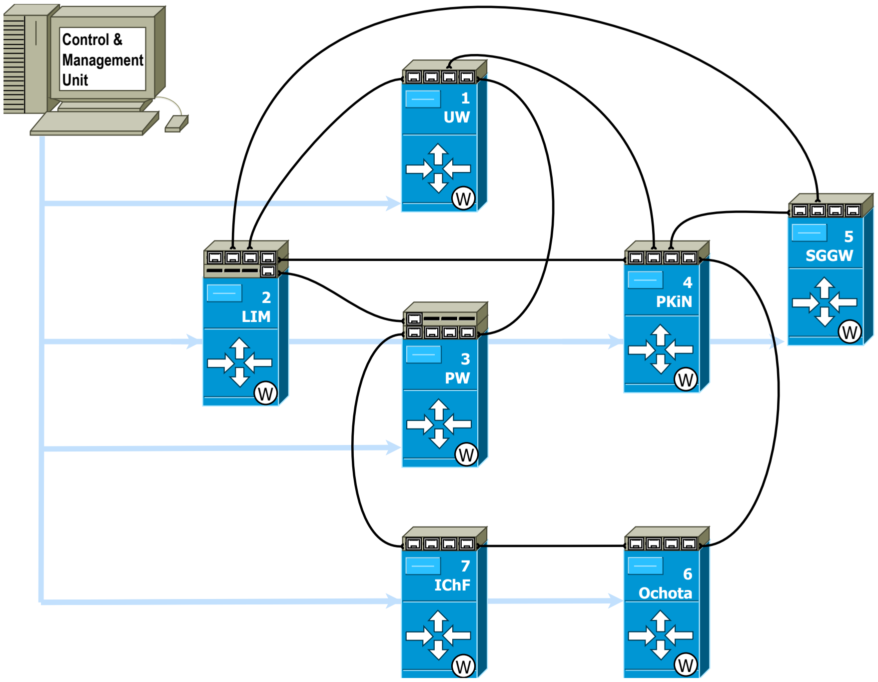

Figure 10: Testbed network (*Net MAN*).

dedicated links to its peers. Thus, software router structure can be considered modular, which is in line with the energy saving optimization problem formulations presented in this paper. Power consumption of the routers is measured individually by industry power meters, and stored periodically in a database. Each router can operate in two states: active and sleeping. All communication ports can operate in one of three possible active states with the following throughput: 10Mb/s, 100Mb/s and 1Gb/s. Due to the application of standard network line cards we could not switch them off or put into sleep mode.

Every PC in a testbed gets controlled via a separate management network (drawn in thick blue lines) from a dedicated control station. The station runs optimization algorithms and reconfigures nodes appropriately. It also keeps monitoring link status and interface statistics. The management network is also necessary to implement specific test scenarios. The traffic in the presented testbed is generated by *iperf* tool.

7.2. Results of Experiments

We calculated the power status of all network devices and energy-aware routing using both *LNPb* and *LNRP* for the *Net MAN* network and the number of demands $D = 10$. The numerical complexities of the *LNPb* and *LNRP* optimization tasks are the following:

- *LNPb*: number of variables = 330, number of constraints = 390.
- *LNRP*: number of variables = 330, number of constraints = 432.

The results of calculations are collected in Table 13. Finally, we have implemented our control

Table 13: Active routers, cards and links, reduction of energy consumption and time of calculations (*Net MAN*).

Method	*LNPb*	*LNRP*
Active routers	6	6
Active cards	6	6
Active links	10	12
Power reduction [W]	425	405
Power reduction [%]	48.57	46.28
Time [s]	0.999	0.310

framework in the testbed network *Net MAN* in our laboratory. In this paper we present the results of the application of the *hNCP* control scheme for dynamic power management and energy saving in the *Net MAN*. The optimization problems for the number of demands $D = 10$ were solved and energy-efficient routing was calculated. The simple local control (LCP), i.e., on-demand frequency scaling governor was used allowing processor clock frequency to be adopted to the load. Thanks to relatively efficient architecture of network drivers used in the testbed the processor load generated during data transmission was far bellow its abilities. This made it possible to reduce frequency and save significant amount of energy.

Two series of experiments were carried out to validate and compare the effectiveness of both optimization schemes: *LNPb* and *LNRP*. The network topology *Net MAN* was used to transmit 10 flows meant to carry 100Mb/s of offered load each. We tested the resilience of the *hNCP* control scheme on the quality of forecasts of demands $\hat{V}_d$. The experiments comprised of five scenarios:

F1 : $\hat{V}_d = 1.15V_d$, the real traffic $\hat{V}_d$ was 15% lower than the forecast V_d.

F2 : $\hat{V}_d = V_d$, the real traffic was equal to the forecast.

F3 : $\hat{V}_d = 0.85V_d$ the real traffic was 15% higher than the forecast.

F4 : $\hat{V}_d = 0.70V_d$ the real traffic was 30% higher than the forecast.

F5 : $\hat{V}_d = 0.55V_d$ the real traffic was 45% higher than the forecast.

The method of traffic generation using *iperf* program allowed some traffic variation resulting in generating higher rates in particular cases – 85.2Mb/s instead of 85Mb/s in scenario F1, 101Mb/s in scenario F2 and 131Mb/s in scenario F4. Then bottleneck that was experienced by the traffic rate was reduced to prevent packet drops.

The results of the experiments are presented in Tables 14 and 15. They collect the values of offered traffic, measured total traffic and their difference. Moreover, the total power used for data transmitting and the reduction of power consumption w.r.t. the network using traditional shortest path routes that consumes 274W are presented. The results of applying the *LNPb* solution to the testbed network are collected in the Table 14. They show that relatively small savings was gained w.r.t. the network using traditional shortest path routes. The reduction is from 10% to 13% and is attained mostly by switching off one of network nodes (router 7). It must be noted, however that loose topology and even distribution of flows prevented switching off many nodes or links. On the other hand packet processing performed by nodes was relatively simple as the routing tables were short so computers did not engage their full processing power – typically it was possible to carry all calculations using one out of four processor cores. Another reason for lower than expected power proportionality of tested network is the application of standard network line cards which does not allow to be switched off or put into sleep mode.

Scenario	Offered traffic [Mb/s]	Measured total traffic [Mb/s]	Traffic difference [%]	Power consumption [W]	Power reduction [%]
F1	850	852.0	+2.4	239.7	12.8
F2	1000	1004.7	+0.5	240.9	12.1
F3	1150	1130.7	-1.7	242.1	11.6
F4	1300	1274.7	-1.9	243.5	11.1
F5	1450	1400.7	-3.4	245.5	10.4

Table 14: Results of testbed experiments for the network *Net MAN* using the *LNPb* method.

Scenario	Offered traffic [Mb/s]	Measured total traffic [Mb/s]	Traffic difference [%]	Power consumption [W]	Power reduction [%]
F1	850	852.0	+2.40	235.9	13.9
F2	1000	999.4	-0.06	239.5	12.7
F3	1150	1111.4	-3.40	239.3	12.6
F4	1300	1239.4	-4.70	241.0	12.0
F5	1450	1351.4	-6.80	241.6	11.8

Table 15: Results of testbed experiments for the network *Net MAN* using the *LNRP* method.

Using the *LNRp* method (see tab 15) surprisingly results in a bit lower power consumption which may be attributed mostly to less processing necessary to look-up shorter routing tables and lower number of routing policies. This is an important result, which could be observed only via experiments as computational models did not cover this aspect of network operation. On the other hand, policy routing was used just to model MPLS which is a far more efficient and scalable technology.

Attempts to send more traffic than forecasted resulted in slightly increased power consumption and reduction of flows traversing links of decreased bandwidth – there is one such link when the *LNRp* solution is used and two when using the *LNRP* method. Respectively, total throughput achieved was lower for the *LNRP*. In both cases the network allowed to send more traffic on most of paths as link capacities were scaled exponentially.

8. Summary and Concluding Remarks

The paper summarizes the results of the research concerned with the design and development of modern energy-aware wired network systems. We have developed a general control framework for low energy consumption network that can operate in two variants: centralized and hierarchical.

This framework incorporates two control levels, i.e., the central unit that implements network-wide control strategies and local units (network devices) that implement local control mechanisms. Two formulations of the network-wide optimization problem have been given, stated in terms of binary and continuous variables. We discussed the limitations of the complete binary problem formulation assuming full routing and energy state of all devices calculation and problems with their application to power control in real networks. We proposed the relaxation of this formulation that leads to a continuous optimization. An efficient branch-and-bound algorithm supported by heuristics to solve it has been proposed.

The control scheme has been validated for various network systems and traffic engineering through simulation and testbed implementation. As a final observation one can say that in general the relaxed optimization problem formulation (*LNRP*) and branch-and-bound implementation supported by heuristics gives slight worse results than the complete problem formulation (*LNPb*) and branch-and-bound implementation. Hence, the *LNPb* is recommended to use in small-size networks. Due to the limitation of the applicability of complete problem formulation to medium- and large-size networks concerned with the high complexity of the optimization problem we recommend to use *LNPb* to obtain energy saving in larger networks. It is obvious that the effectiveness of the proposed control scheme strongly depends on the quality of future traffic forecasting. Hierarchical control scheme *hNCP* should be more resistant to forecasting inaccuracy, and we recommend this approach to be used in networks with high variability of a network traffic or forecast inaccuracy.

Acknowledgment

The work was supported by 7 Framework Program UE grant ECONET, No: 258454.

References

[1] F. Bianco, G. Cucchietti, G. Griffa, Energy consumption trends in the next generation access network – a telco perspective, in: Proc. 29th Inter. Telecommunication Energy Conf. (INTELEC 2007), 2007, pp. 737–742.

[2] R. Bolla, R. Bruschi, F. Davoli, F. Cucchietti, Energy Efficiency in the Future Internet: A Survey of Existing Approaches and Trends in Energy-Aware Fixed Network Infrastructures, IEEE Communications Surveys & Tutorials 13 (2011) 223–244.

[3] S. N. Roy, Energy logic: a road map to reducing energy consumption in telecom munications networks, in: Proc. 30th International Telecommunication Energy Conference (INTELEC 2008), 2008.

[4] L. Wang, S. Khan, Review of performance metrics for green data centers: a taxonomy study, The Journal of Supercomputing DOI 10.1007/s11227-011-0704-3 (2011) 1–18.

[5] ECONET, Project homepage, http://www.econet-project.eu.

[6] E. Rosen, A. Viswanathan, R. Callon, Rfc 3031: Multiprotocol label switching architecture, IETF, January.

[7] D. Awduche, L. Berger, D. Gan, T. Li, V. Srinivasan, G. Swallow, Rfc 3209-rsvp-te: Extensions to rsvp for lsp tunnels, Network Working Group, Dec 64.

[8] J. Chabarek, J. Sommers, P. Barford, C. Estan, D. Tsiang, S. Wright, Power awerness in network design and routing, in: Proc. 27th Conference on Computer Communications (INFOCOM 2008), 2008, pp. 457–465.

[9] A. Qureshi, R. Weber, H. Balakrishnan, Cutting the electric bill for internet-scale systems, in: Proc. SIGCOMM'09, ACM, 2009, pp. 123–134.

[10] R. Bolla, R. Bruschi, A. Ranieri, Green support for pc-based software router: performance evaluation and modeling, in: Proc. IEEE International Conference on Communications (ICC'09), 2009, pp. 1–6.

[11] N. Vasić, D. Kostić, Energy-aware traffic engineering, in: Proc. 1st Inter. Conference on Energy-Efficient Computing and Networking (E-ENERGY 2010), 2010.

[12] S. Nedevschi, l. Popa, G. Iannacone, D. Wetherall, S. Ratnasamy, Reducing network energy consumption via sleeping and rate adaptation, in: Proc. 5th USENIX Symposium on Networked Systems Design and Implementation, 2008, pp. 323–336.

[13] R. Bolla, R. Bruschi, A. Carrega, F. Davoli, Theoretical and technological limitations of power scaling in network devices, in: Proc. 2010 Australasian Telecommunication Networks and Applications Conference (ATNAC), IEEE, 2010, pp. 37–42.

[14] IEEE, IEEE 802.3az Energy Efficient Ethernet Task Force, http://grouper.ieee.org/groups/802/3/az/public/index.html (2012).

[15] M. Pióro, M. Mysłek, A. Juttner, J. Harmatos, A. Szentesi, Topological Design of MPLS Networks, in: Proc. GLOBECOM'2001, 2001.

[16] P. Jaskóła, K. Malinowski, Two methods of optimal bandwidth allocation in TCP/IP networks with QoS differentiation, in: Proc. Summer Simulation Multiconference (SPECTS'04), 2004, pp. 373–378.

[17] K. Malinowski, E. Niewiadomska-Szynkiewicz, P. Jaskóła, Price Method and Network Congestion Control, Journal of Telecommunications and Information Technology 2 (2010) 73–77.

[18] J. Restrepo, C. Gruber, C. Machuca, Energy Profile Aware Routing, in: Proc. 1st International Workshop on Green Communications, IEEE International Conference on Communications (ICC'09), 2009, pp. 1–5.

[19] L. Chiaraviglio, M. Mellia, F. Neri, Minimizing ISP network energy cost: formulation and solutions, IEEE/ACM Transactions on Networking 20 (2011) 463 – 476.

[20] E. Niewiadomska-Szynkiewicz, A. Sikora, P. Arabas, J. Kołodziej, Control system for reducing energy consumption in backbone computer network, Concurrency and Computation: Practice and Experience DOI: 10.1002/cpe.2964.

[21] W. Fisher, M. Suchara, J. Rexford, Greening backbone networks: reducing energy consumption by shutting off cables in bundled links, in: Proc. 1st ACM SIGCOMM workshop on Green networking (Green Networking'10), ACM, 2010, pp. 29–34.

[22] C. Lin, S. Soh, K. Chin, M. Lazarescu, Efficient heuristics for energy-aware routing-based in networks with bundled links, Computer Networks 57 (2013) 1774–1788.

[23] F. Idzikowski, S. Orlowski, C. Raack, H. Rasner, A. Wolisz, Saving energy in IP-over-WDM networks by switching off line cards in low-demand scenarios, in: Proc. 14th Conference on optical network design and modeling (ONDM'10), IEEE, 2010.

[24] J. Kołodziej, S. Khan, F. Xhafa, Genetic algorithms for energy-aware scheduling in computational grids, in: Proc. 6th IEEE International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC2011), 26-28.10.2011, Barcelona, Spain, 2011, pp. 17–24.

[25] J. Kołodziej, S. Khan, L. Wang, A. Zomaya, Energy efficient genetic-based schedulers in computational grids, Concurrency and Computation: Practice and Experience DOI: 10.1002/cpe.2839.

[26] L. Chiaraviglio, M. Mellia, F. Neri, Energy-aware backbone networks: a case study, in: Proc. 1st International Workshop on Green Communications, IEEE International Conference on Communications (ICC'09), IEEE, 2009, pp. 1–5.

[27] M. Zhang, C. Yi, B. Liu, B. Zhang, GreenTE: power-aware traffic engineering, in: Proc. IEEE Inter. Conference on Network Protocols (ICNP'2010).

[28] G. Shen, R. S. Tucker, Energy-minimized desig for IP over WDM networks, Journal of Optical Communications and Networking 1 (2009) 176–186.

[29] A. Cianfrani, V. Eramo, M. Listani, M. Marazza, E. Vittorini, An Energy Saving Routing Algorithm for a Green OSPF Protocol, in: Proc. IEEE INFOCOM Conference on Computer Communications, IEEE, 2010, pp. 1 – 5.

[30] A. P. Bianzino, L. Chiaraviglio, M. Mellia, GRiDA: a green distributed algorithm for backbone networks, in: Online Conference on Green Communications (GreenCom 2011), IEEE, 2011, pp. 113–119.

[31] F. Cuomo, A. Abbagnale, A. Cianfrani, M. Polverini, Keeping the connectivity and saving the energy in the Internet, in: Proc. IEEE INFOCOM 2011 Workshop on Green Communications and Networking, IEEE, 2011, pp. 319–324.

[32] A. Coiro, M. Listani, A. Valenti, F. Matera, Energy-aware traffic engineering: A routing-based distributed solution for connection-oriented IP networks, Computer Networks 57 (2013) 2004–2020.

[33] F. Cuomo, A. Cianfrani, M. Polverini, D. Mangione, Network pruning for energy saving in the Internet, Computer Networks 56 (2012) 2355–2367.

[34] Y.-M. Kim, E.-J. Lee, H.-S. Park, J.-K. Choi, H.-S. Park, Ant colony self adaptive energy saving routing for energy efficient Internet, Computer Networks 56 (2012) 2343–2354.

[35] G. Sakelari, C. Morfopoulou, E. Gelembe, Investigating the tradeoffs between power consumption and quality of service in a backbone network, Future Internet 5 (2013) 2–281.

[36] Hewlett-Packard Corp., Intel Corp., Microsoft Corp., Phoenix Technologies Ltd., Toshiba Corp., Advanced Configuration and Power Interface Specification, Revision 5.0, 2011.

[37] D. Reforgiato, A. Lombardo, F. Davoli, L. Fialho, M. Collier, P. Donadio, R. Bolla, R. Bruschi, Exporting Data-Plane Energy-Aware Capabilities from Network Devices toward the Control Plane:, in: Proc. 17th European Conference on Network and Optical Communication (NOC 2012), 2012, pp. 1–5.

[38] P. Arabas, K. Malinowski, A. Sikora, On formulation of a network energy saving optimization problem, in: Proc. of 4th Inter. Conference on Communications and Electronics (ICCE 2012), 2012, pp. 122–129.

[39] Lpsolve, solver homepage, http://lpsolve.sourceforge.net/5.5/.

[40] R. Bolla, *et al.*, Econet deliverable d2.1 end-user requirements, technology, specifications and benchmarking methodology, https://www.econet-project.eu/Repository/DownloadFile/291 (2011).