

EWA: an Adaptive Algorithm for Energy Saving in IP-over-WDM Networks

Filip Idzikowski,^a Luca Chiaraviglio,^b Edoardo Bonetto^b

a) Technische Universität Berlin, TKN, Berlin, Germany

b) Department of Electronics and Telecommunications, Politecnico di Torino, Torino, Italy

Email: filip.idzikowski@tu-berlin.de, {luca.chiaraviglio, edoardo.bonetto}@polito.it

Abstract—We propose Energy Watermark Algorithm (EWA), an adaptive algorithm to reduce power consumption in Internet Protocol (IP)-over-Wavelength Division Multiplexing (WDM) networks. Our solution wisely adapts the set of powered on line cards to the actual traffic demand. In particular, EWA trades between network power consumption and Quality of Service (QoS) by changing the level of overprovisioning of the lightpaths.

We evaluate the performance of EWA over a set of realistic scenarios. Results show that EWA adapts the power consumption to the traffic variation, while limiting the amount of rerouted traffic. We also show that a proper setting of the input parameters allows to avoid overload introduced by the power saving.

I. INTRODUCTION

Core networks consume a non-negligible amount of power [1] due to the fact that core devices exchange huge amount of data. Power consumption of an IP-over-WDM network can be reduced by switching off network components in low demand hours [2]. This is a promising approach since currently power consumption of core network devices in the IP layer hardly depends on the load [3].

Switching network devices on and off introduces changes in the network configuration, which may influence the QoS. A constraint on the maximum link utilization has been often used in the literature to guarantee the QoS, i.e., when devices are switched off the maximum link load is below a given threshold [4]. However, this is not sufficient to guarantee a realistic QoS, because switching off a network device is a complex task and it may introduce rerouting of traffic demands. In particular, the switch-off process has to be coordinated with other devices in order to move traffic from the current device to the other ones that remain powered on. However, the traffic rerouting is not instantaneous, leading to potential losses of data and consequently QoS degradation. Additionally, even the switch-on process is not instantaneous, since it may take some time to power on the circuits and recover from a suspended state. For all these reasons, the energy-aware process has to be carefully planned in order to limit the QoS deterioration.

In this work, we propose EWA, an algorithm to reduce the power consumption of IP-over-WDM networks, in which Line Cards (LCs) can be switched off and switched on based on the traffic variation over time. We search for appropriate settings of the EWA parameters in order to reduce power consumption of the network, eliminate overload, and limit the frequency of changing of power state of each device leading to rerouting of traffic. Specifically, we consider the amount of traffic which

has to be reconfigured when some devices in the network change their power state.

EWA takes an adaptive approach and utilizes the network configuration from the previous time period. The algorithm is based on [5], but it is more aggressive in switching off devices in order to save energy, and it is also able to adapt to higher variations of traffic between two consecutive time periods. We evaluate EWA over different scenarios, assuming as realistic assumptions as possible. Our results show that EWA is able to wisely adapt power consumption while reducing the amount of reconfigured traffic.

The rest of the paper is organized as follows. The network model and metrics are presented in Section II. EWA is described in Section III. Results of an energy evaluation study based on realistic input data are presented in Section IV. Finally, conclusions are drawn in Section V.

II. NETWORK MODEL

We consider an IP-over-WDM network, where Optical Cross-Connects (OXCs) are interconnected by fibers in the WDM layer, and IP routers are interconnected by lightpaths in the IP layer. IP routers have a modular structure and are composed of one Line Card Shelf (LCS) or of several LCSs interconnected by one or more Fabric Card Shelves (FCSs) [6]. LCs are located in LCSs and they are the end-points of lightpaths. A lightpath is a concatenation of WDM channels and is terminated by a transmitter and a receiver, both located in a LC of a router. Bundles of parallel lightpaths form a logical link and all the logical links together with IP routers constitute the Logical Topology (LT) of the network.

Notation More formally, the LT is modeled as a directed graph $H = (V, L)$ where V is the set of all nodes in the network and L is the set of supplied logical links on which lightpaths can be established. Each lightpath has bitrate C . Power consumed by a LC, LCS and FCS is denoted by \mathcal{P}^{LC} , \mathcal{P}^{LCS} and \mathcal{P}^{FCS} , respectively.

We consider a set of time periods T consisting of past and future time periods ($T = T_{past} \cup T_{fut}$, $T_{past} \cap T_{fut} = \emptyset$). A Traffic Matrix (TM) $D(t)$ for each time period $t \in T$ contains traffic demands between the nodes $(a, b) \in V \times V$ with values $d^{ab}(t)$. Traffic exchanged during $T_{past} \subset T$ is used to determine the set of installed devices. We call this procedure Static Base Network (SBN) design, and use the Genetic Algorithm (GA) [7] with the objective of Capital

Expenditures (CapEx) minimization for this purpose. The SBN is dimensioned to satisfy the maximum TM D_{SBN} , based on the set of past time periods T_{past} :

$$d_{SBN}^{ab} = \max_{t \in T_{past}} d^{ab}(t), \quad \forall a, b \in V \quad (1)$$

The EWA is executed at each t out of the set of future time periods $T_{fut} \subset T$. The duration of each time period $t \in T_{fut}$ is denoted as Δt . Since EWA is executed at each $t \in T_{fut}$, we introduce variables that are updated every Δt . The flow variables $f_{ij}^{ab}(t) \in \{0, 1\}$ denote whether the traffic demand originated at node a and targeted to node b traverses the logical link from i to j at time t . Single-, shortest-path routing of traffic demands over the LT is assumed. Moreover, the variables $y_l(t) \in \mathbb{Z}_+$ determine the number of lightpaths established on the logical link $l \in L$ at time t , which in turn determine the powered on LCs. Finally, $x_i^{LC}(t) \in \mathbb{Z}_+$ is the number of LCs powered on at each node i at time t , which is bounded by the number of installed LCs X_i^{LC} in each node of the SBN.

Using the terms explained above, we define the network configuration as the set of network nodes V with installed LCs X_i^{LC} (powered on or off), established lightpaths forming logical links $y_l(t)$ and IP routing of traffic demands $f_{ij}^{ab}(t)$. The network model is similar to [2]. The main difference is that we use directed graphs and traffic demands in this work.

Metrics A trade-off between reconfigured traffic and the power consumption is investigated in this work. Therefore we look at the following metrics.

Power consumption of all LCs active in the network (as a function of time) is defined as:

$$P^{LC}(t) = \mathcal{P}^{LC} \sum_{i \in V} x_i^{LC}(t) \quad (2)$$

Power consumption of active LCSs and FCSs is determined by the number of active LCs. The number of LCSs used at each node is expressed as:

$$x_i^{LCS}(t) = \lceil x_i^{LC}(t) / U_{LCS} \rceil \quad (3)$$

where U_{LCS} is the capacity (in terms of LCs) of a LCS. The number of FCS is determined by the number of LCSs:

$$x_i^{FCS}(t) = \begin{cases} 0 & \text{if } x_i^{LCS}(t) \leq 1 \\ \lceil x_i^{LCS}(t) / U_{FCS} \rceil & \text{otherwise} \end{cases} \quad (4)$$

where U_{FCS} denotes the maximum number of LCSs that a FCS can interconnect. Power consumption of active LCSs in the network is given by:

$$P^{LCS}(t) = \mathcal{P}^{LCS} \sum_{i \in V} x_i^{LCS}(t) \quad (5)$$

Similarly, the power consumption of active FCSs is defined as:

$$P^{FCS}(t) = \mathcal{P}^{FCS} \sum_{i \in V} x_i^{FCS}(t) \quad (6)$$

The total power consumption of the whole network is hence defined as:

$$P^{TOT}(t) = P^{LC}(t) + P^{LCS}(t) + P^{FCS}(t) \quad (7)$$

We consider the traffic that needs to be rerouted in order to reduce power consumption of the network. Therefore, let us define as $r_{ij}^{ab}(t) \in \mathbb{R}_+$ the amount of reconfigured traffic between a and b on the logical link from i to j at time t with respect to time $t-1$, with $t > 0$ belonging to the set of future time periods T_{fut}

$$r_{ij}^{ab}(t) = \begin{cases} d^{ab}(t) \cdot f_{ij}^{ab}(t) - d^{ab}(t-1) \cdot f_{ij}^{ab}(t-1) & f_{ij}^{ab}(t) > f_{ij}^{ab}(t-1) \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

We introduce the reconfiguration ratio over all subsequent pairs of time periods in T_{fut} with $t > 0$ as:

$$\xi = \frac{\sum_{t \in T_{fut}} \sum_{i \in V} \sum_{j \in V} \sum_{a \in V} \sum_{b \in V} r_{ij}^{ab}(t)}{\sum_{t \in T_{fut}} \sum_{a \in V} \sum_{b \in V} d^{ab}(t)} \quad (9)$$

This metric captures the amount of traffic which is reconfigured over all time periods in T_{fut} , normalized by the total amount of traffic which is exchanged in the network. The ξ may be greater than 1 since the reconfigured traffic is counted multiple times if it passes through multiple logical links from the source to the target.

Finally, we define the overload ratio metrics to capture overload traffic in all periods $t \in T_{fut}$. We look at the overload ratio at each $t \in T_{fut}$ in the network before reconfiguration determined by EWA (Eq. (10)), and in the network after the reconfiguration determined by EWA (Eq. (11)). Flow variables $f_{ij}^{ab}(t-1)$ at previous time period $t-1$ (before reconfiguration) are used in Eq. (10), and flow variables $f_{ij}^{ab}(t)$ at current time period t (after reconfiguration) are used in Eq. (11). Both ϕ^{PRE} and ϕ^{POST} should be as low as possible (ideally 0) to prevent service disruptions and loss of QoS. Note that we take into account each hop that a demand passes through when calculating the overload, even though traffic can be dropped only once in the network. Therefore, ϕ^{PRE} and ϕ^{POST} may also be greater than 1. Since EWA adapts the network configuration to the current traffic demands, $\phi^{PRE} \geq \phi^{POST}$.

III. ENERGY WATERMARK ALGORITHM

We first present the general idea of the EWA, then we detail the algorithm and finally we show the differences with respect to previous work.

A. General Idea

The EWA adapts the network to current traffic situation in order to save energy on one hand, and limit the load on logical links in order to ensure certain QoS on the other hand. EWA uses a low and a high watermark (W_L and W_H) defined as thresholds on the utilization of the last lightpath on a logical link. Exceeding the W_H triggers attempts to establish additional lightpath(s) in order to avoid overload of the network. Exceeding the W_L triggers attempts to release lightpath(s) in order to switch off idle LCs and save energy. EWA makes sure that the maximum utilization of last lightpath on a logical link ψ is not exceeded when trying to release lightpaths.

$$\phi^{PRE} = \frac{\sum_{t \in T_{fut}} \sum_{i \in V} \sum_{j \in V} \max \left(\sum_{a \in V, b \in V} d^{ab}(t) \cdot f_{ij}^{ab}(t-1) - \sum_{l \in L_{(i,j)}} C y_l(t-1), 0 \right)}{\sum_{t \in T_{fut}} \sum_{a \in V} \sum_{b \in V} d^{ab}(t)} \quad (10)$$

$$\phi^{POST} = \frac{\sum_{t \in T_{fut}} \sum_{i \in V} \sum_{j \in V} \max \left(\sum_{a \in V, b \in V} d^{ab}(t) \cdot f_{ij}^{ab}(t) - \sum_{l \in L_{(i,j)}} C y_l(t), 0 \right)}{\sum_{t \in T_{fut}} \sum_{a \in V} \sum_{b \in V} d^{ab}(t)} \quad (11)$$

Alg. 1 shows the main pseudocode of EWA. Details of its subroutines are presented in the next section. The algorithm takes as input the network configuration in previous time period $t-1$ (network nodes V with installed LCs X_i^{LC} , established lightpaths forming logical links $y_l(t-1)$ and IP routing of traffic demands $f_{ij}^{ab}(t-1)$), TM $D(t)$ for the current period t , capacity of a lightpath (WDM channel) C , W_L , W_H and ψ . Updated network configuration is returned as output of the algorithm.

EWA first checks whether all the demands in the current network configuration are routable, and iteratively tries to establish additional lightpath(s) for the unroutable demands (if any), starting from the largest ones (line 1). The logical links on which watermarks are exceeded are identified next (line 2), and violation of the W_H is checked, starting from the logical links with the highest utilization of the last lightpath (line 3). For each overloaded logical link (from the most overloaded to the least overloaded), the algorithm first tries to increase the capacity of the logical link if a demand with the same source and target flows through it. If this is not the case, attempts are made to establish lightpath(s) for the possibly biggest demand flowing through the overloaded logical link.

Once load is lower than W_H for all logical links, or it is impossible to reduce overload anymore, violation of the W_L is checked starting from the least loaded logical links (line 4). One lightpath per iteration is tried to be released making sure that ψ is not exceeded.

B. Main Steps

We then summarize the main steps of EWA. We refer the reader to [8] for the detailed algorithm description.

1) *Ensuring Routability of Demands*: The main requirement on an energy-saving algorithm is not to influence the connectivity of the network. This means that given a set of

traffic demands $D(t)$, there must be at least one path in the LT determined by $y_l(t)$ to route each traffic demand. Therefore the first step of EWA is to ensure that all the demands are routable. In particular, starting from the current network configuration the unroutable demands are identified and sorted in descending order according to the demand value. Then, iteratively for each unroutable demand, a logical link is tried to be established. The routine iterates until all the demands are routable or it is impossible to establish additional logical links. Finally, updated network configuration is returned.

2) *Establishing Necessary Lightpaths*: During this step the algorithm ensures that the capacity of the network is able to satisfy the current traffic demands $D(t)$. In particular, given the updated network configuration and the sorted list of logical links where the high watermark is exceeded, the algorithm iteratively tries to add lightpaths. Initially, the set of logical links is ordered in descending order of utilization of the last lightpath. Then, for each logical link violating the high watermark, attempts to establish new lightpath(s) are made. The algorithm tries to offload the current logical link by establishing additional lightpaths and shifting the demands from the current logical link to the new lightpaths. In particular, two cases are identified based on the demands flowing through the current logical link: a) if a demand with the same source and target nodes as the current logical link flows through the current logical link, and its value is higher than the capacity of the logical link, parallel lightpaths are added to the logical link (subject to availability of transmitters and receivers), until the current demand is satisfied, b) otherwise, a new logical link is established directly between the source and destination of a demand flowing through the current logical link if transmitter(s) and receiver(s) are available. Note that the demands satisfying condition b) are sorted in decreasing order, starting from the biggest one. The idea is to fill the newly established lightpaths to a high extent (preserving the W_H), which corresponds to efficient usage of the power of the corresponding LCs. The algorithm iterates over the set of demands until the high watermark constraint is ensured (or no more demands remain). The procedure is repeated for all logical links violating the high watermark.

3) *Releasing Unnecessary Lightpaths*: The last step of the EWA is to attempt to release unnecessary lightpaths in order to save energy. This subroutine takes as input the current network configuration and the set of logical links violating the low watermark. Then, the algorithm iteratively selects a logical link, starting from the one having the lowest utilization of the last lightpath. A single lightpath of the current logical link is

Algorithm 1 Pseudo-code of EWA.

Input: netConfig from period $t-1$, current TM $D(t)$, C , W_L , W_H , ψ

Output: Updated netConfig

- 1: ensureDemandsRoutability(netConfig, $D(t)$, C);
 - 2: sortedLLsExceedingWMs = getSortedLLsExceedingWMs(netConfig, $D(t)$, W_L , W_H);
 - 3: establishNecessaryLpaths(netConfig, $D(t)$, C , sortedLLsExceedingWMs, W_L , W_H);
 - 4: releaseUnnecessaryLpaths(netConfig, $D(t)$, sortedLLsExceedingWMs, W_L , W_H , ψ);
-

attempted to be released keeping the constraints on routability as well as the constraints on ψ in the whole network. Then, the procedure is repeated for all logical link violating W_L , until it is possible to release lightpaths.

C. Differences to related work

To the best of authors' knowledge, no heuristic based on watermarks triggering establishing and releasing of lightpaths has been investigated so far in the literature focusing on power consumption. Our work was inspired by [5]. The main difference to [5] (apart from the focus) is that EWA does not forbid releasing lightpath(s) in the case when some lightpath(s) have already been established in the same run of the algorithm. This allows more attempts to release lightpaths (more aggressive energy saving) and is not critical for the operation of the network, since releasing of lightpaths is performed in the last step of the algorithm (line 4 of Alg. 1). Lightpaths are released only after establishing the new ones, so that the necessary rerouting can be performed in a controlled way. Moreover, differently from [5], EWA is able to add or delete more than one lightpath during the execution of the algorithm. Preliminary investigations have shown that adding or deleting of a single lightpath is insufficient to adapt the network to the traffic changes based on input data originating from measurements [9]. Finally, the parameter ψ is added in order to trade between QoS and energy saving.

Another work which uses the concept of watermarks is [10]. The authors of [10] introduce the load balance indicator (bound on maximum lightpath load of the new logical topology at reconfiguration point), which is similar to ψ , however defined on a lightpath, and not on (the last lightpath of) a logical link. Moreover, the watermarks are used only to trigger solving the optimization problem, which does not use the watermarks themselves, but only the load balance indicator. The optimization problem gives the possibility to redistribute the load in the network by rerouting only, without adding or deleting lightpaths, however solving an optimization problem is time consuming and therefore impractical in real operation. The authors of [10] propose also a so-called softbound approach, where reconfiguration is done only after three observation periods since the previous reconfiguration if in between there is no violation of the high watermark. Eventually an approximate mathematical model is proposed in [10], where the added and deleted lightpath(s) are chosen among some candidate lightpaths. Algorithms for the selection of the candidate lightpaths are also described in [10]. They perform the routing and wavelength assignment in order to reduce the complexity of the mathematical formulation.

IV. PERFORMANCE EVALUATION

We evaluate the EWA on the Géant network (22 nodes and 36 bidirectional physical supply links) using corresponding traffic data originating from measurements available at [9]. Details of the adopted traffic scenarios and power models are reported in [8]. In brief, we use realistic CapEx [6] and power consumption values corresponding to Cisco CRS-1 [11]

assuming capacity C of a WDM channel equal to 40 Gbps. The maximum utilization of the logical links allowed during the SBN design was set to 0.5. T_{past} spans over a month, and EWA is evaluated over a day (T_{fut}) with $\Delta t = 15$ min. Measured traffic [9] is rescaled to meet current behavior so that the total demand (sum of all elements of a TM) of D_{SBN} is equal to 6.6 Tbps (300 Gbps per node). We run EWA on a personal computer equipped with a Dual Core CPU at 2.4 GHz and 2 GB of RAM using a Java framework. We consider different algorithm parameters (W_L and W_H , and $\psi = W_H$), and different performance metrics (power consumption, reconfiguration ratio and overload ratios) as explained in Section II.

Switching off the Line Cards We assume that it is easier to switch off LCs than to switch off LCSs and FCSs so we first look at the power consumption of this type of devices according to Eq. (2). Results for different settings of W_L and W_H are presented in Fig. 1. Fig. 1(a) reports the results for $W_L = 0.1$ and different values of W_H . When W_H is low, the active LCs tend to consume more power than when it is high. This is due to the fact that the low value of the low watermark ($W_L = 0.1$) rarely triggers attempts to release lightpaths, while low values of W_H and ψ introduce overprovisioning and thus require many devices to be powered on. To give more insight we extend our analysis considering $W_L = 0.3$ (Fig. 1(b)) and $W_L = 0.5$ (Fig. 1(c)). The power consumption decreases as W_L increases, since the algorithm becomes more aggressive in releasing lightpaths. Moreover, the increase of W_H further reduces the power consumption. Regardless of the setting of the EWA parameters, power consumption of LCs follows the variation of the traffic depicted in Fig. 1(c).

Breakdown of the total power consumption We then extend our analysis considering the case in which also LCSs and FCSs can be switched off. Fig. 2 details the power consumption of active LCs, LCSs and FCSs in the network for different values of W_L and W_H . Activation and deactivation of FCSs and LCSs occur less frequently than activation and deactivation of LCs. However, the impact of switching a shelf on or off on the total power consumption of the network is significant due to high values of power consumption of single components (see [8]). The power consumption of all active LCs is higher than the power consumption of all active FCSs and slightly lower than the power consumption of all active LCSs (Fig. 2(a)), even though a single LC consumes significantly less power than a single LCS or FCS. This is due to the fact that there are much more LCs in the network than LCSs and FCSs. With increasing values of W_L and W_H (Figures 2(b) and 2(c)), the share of LCs in the total power consumption of the network decreases. More aggressive attempts to release lightpaths (higher W_L) result in longer periods when some FCSs and LCSs are powered off. The length of these periods is related to the granularity of capacity provided by each component. Since the capacity of LCSs is of finer granularity than the one of FCSs, adaptation of these network components to traffic is more accurate (compare Figures 2(a), (b) and (c) with the total demand curve in Fig. 1(c)).

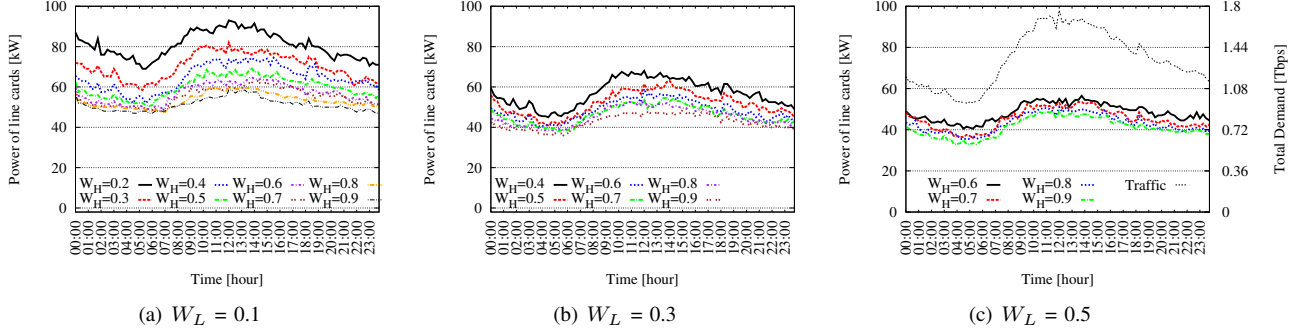


Fig. 1. Power consumption of LCs in the Géant network together with the total demand in (c).

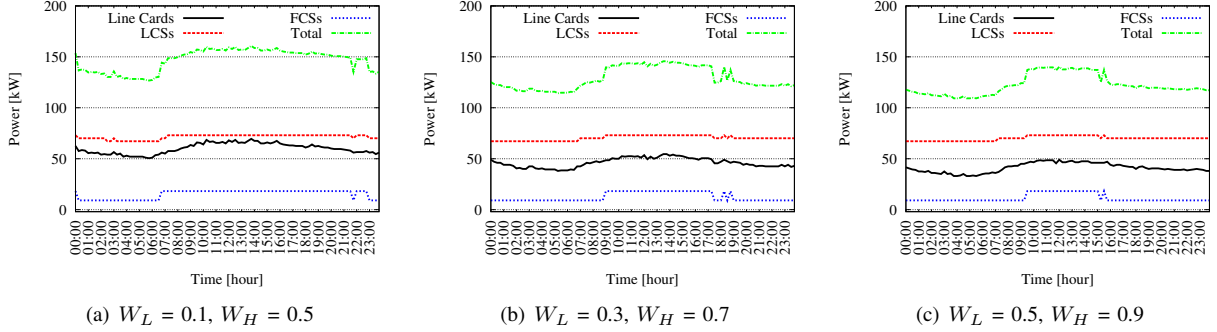


Fig. 2. Power consumption in the Géant network.

EWA allows to reduce power consumption of all types of devices during low demand hours. We quantify the power savings at 05:45 GMT with respect to the high demand hour (13:45 GMT) assuming $W_L = 0.1$ and $W_H = 0.5$ (see Fig. 2(a)). Switching off LCs, LCSs and FCSs allows to save respectively 27%, 8% and 50% of power in the low demand hour. The 50% results from the fact that one out of two FCSs used in the high demand hour is switched off in the low demand hour. The total power consumption of the network is reduced by 21% in the low demand hour with respect to the high demand hour.

Although evaluation of the power saving in comparison to the SBN network is not the main focus of this work, since it highly depends on the assumed SBN, we report that the power consumption of all LCs installed in the SBN is equal to 276 kW, all installed LCSs consume 201.48 kW, and all FCSs consume 163.8 kW. Assuming again $W_L = 0.1$ and $W_H = 0.5$, the power saving in the low demand hour is equal to 82%, 67%, 94% and 80% for LCs, LCSs, FCS and for the whole network, respectively. During the high demand hour the corresponding values are 75%, 64%, 89% and 75%.

The high power savings can be explained by the fact the SBN was designed using the maximum traffic out of a period of one month (see Eq. (1)) and overprovisioned. The large savings achieved by switching off FCSs can be explained by balancing between using zero and one FCS at each node.

Reconfiguration costs Finally, we evaluate the performance of EWA considering the reconfigured traffic and the network overload before and after the reconfiguration determined by

EWA (Fig. 3). The reconfiguration ratio (Eq. (9)) is typically lower than 0.12. In particular, setting the W_L to 0.1 results in the lowest reconfiguration ratio. This is due to the fact that, with this configuration, the algorithm is less aggressive in releasing lightpaths. On the contrary, as W_L increases, the reconfiguration ratio tends to increase showing the trade-off between reducing power consumption (high W_L) and reducing the reconfigured traffic (low W_L). Moreover, the reconfigured traffic does not consistently depend on the values of W_H . The overload ratio before reconfiguration (Eq. (10)) is shown in Fig. 3(b), and the overload ratio after reconfiguration (Eq. (11)) is shown in Fig. 3(c). While ϕ^{POST} is always equal to zero in the considered scenario, ϕ^{PRE} takes very low, but positive values for some settings of EWA. In particular, there is no overload if W_H and ψ are less or equal to 0.4 for all considered values of W_L . The border line for the overload of the network is W_H equal to 0.5. This is inline with the maximum utilization of the logical links assumed during the SBN design. However, even very aggressive setting of EWA ($W_L = 0.1$ and $W_H = 0.9$) does not cause big overload in the network before reconfiguration, i.e. ϕ^{PRE} is equal to 0.0018 in this case. Another way to understand the metric ϕ^{PRE} is to consider the knowledge of traffic in the network. Namely, one of the input parameters of EWA is a TM (see Alg. 1). Although traffic in core networks is much smoother than in the access networks, it may vary within one observation period (corresponding to Δt equal to 15 min. in this work). EWA triggers establishing and releasing lightpaths using the level of utilization of the last lightpath on a logical link. Considering ϕ^{PRE} is equivalent to

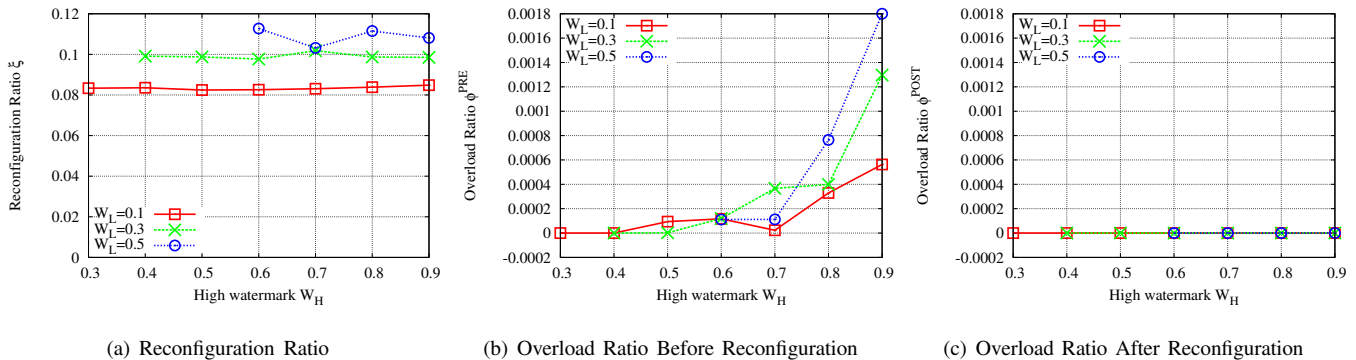


Fig. 3. Variation of Reconfiguration Ratio and Overload Ratios with different EWA parameters.

the assumption that only traffic information collected in the past is used to dynamically determine the configuration of the network

We evaluated the EWA also on the Abilene network (12 nodes and 15 bidirectional physical supply links). The obtained results were similar to the ones obtained on the Géant network. We refer the reader to [8] for more details. An interesting observation stems from the fact that the first period of the considered Abilene data (in contrast to Géant) is a high demand hour. EWA parameterized with very low values of W_H and ψ may fail to find a network configuration which does not violate these parameters, and hence no deactivation of LCs in the whole network is performed in order to minimize the violation. Since EWA has an adaptive character, introduction of this algorithm into the SBN needs to be performed carefully.

V. CONCLUSION

We proposed an adaptive algorithm which dynamically determines configuration of IP-over-WDM networks according to current traffic demands. Results show that it is possible to switch off several network devices while taking care of QoS (constraints on the maximum utilization of logical links), and limiting the amount of reconfigured traffic.

High values of both watermarks result in high power savings. However low value of the low watermark W_L still allows saving significant amount of power, while limiting the amount of reconfigured traffic. EWA manages to eliminate overload in the considered network scenario if the current traffic demands and load on all logical links are known. More conservative setting of EWA parameters need to be chosen in order to eliminate the overload also in the case when only information about traffic situation in the past is known. We are currently working on comparison of the EWA with other heuristics reducing power consumption of IP-over-WDM networks. We are considering various networks, traffic patterns and load of the network.

As future work, we plan to integrate the physical constraints to EWA including the maximum length of a lightpath, routing and wavelength assignment, as well as limited number of wavelengths available on a physical link. Although EWA makes sure that transition from the old network configuration

to a new one is feasible due to activation of necessary lightpaths before releasing the unnecessary ones, we would also like to validate how the activation and deactivation times of network devices as well as the control mechanisms influence the amount of power that can be saved using EWA.

ACKNOWLEDGEMENTS

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement n. 257740 (Network of Excellence “TREND”).

REFERENCES

- [1] J. Baliga, R. Ayre, K. Hinton, W. Sorin, and R. Tucker, “Energy consumption in optical IP networks,” *Journal of Lightwave Technology*, vol. 27, pp. 2391–2403, July 2009.
- [2] F. Idzikowski, S. Orłowski, C. Raack, H. Woesner, and A. Wolisz, “Dynamic routing at different layers in IP-over-WDM networks – maximizing energy savings,” *Optical Switching and Networking, Special Issue on Green Communications*, vol. 8, pp. 181–200, July 2011.
- [3] J. Chabarek, J. Sommers, P. Barford, C. Estan, D. Tsang, and S. Wright, “Power awareness in network design and routing,” in *Proc. of the INFOCOM, Phoenix, USA*, April 2008.
- [4] L. Chiaraviglio, M. Mellia, and F. Neri, “Minimizing ISP Network Energy Cost: Formulation and Solutions,” *IEEE/ACM Transactions on Networking*, vol. 20, pp. 463–476, July 2011.
- [5] A. Gençata and B. Mukherjee, “Virtual-topology adaptation for WDM mesh networks under dynamic traffic,” *IEEE/ACM Transactions on Networking*, vol. 11, pp. 236–247, April 2003.
- [6] R. Hülsermann, M. Gunkel, C. Meusburger, and D. A. Schupke, “Cost modeling and evaluation of capital expenditures in optical multilayer networks,” *Journal of Optical Networking*, vol. 7, pp. 814–833, September 2008.
- [7] A. Ahmad, A. Bianco, E. Bonetto, D. Cuda, G. G. Castillo, and F. Neri, “Power-aware logical topology design heuristics in wavelength-routing networks,” in *Proc. of the ONDM, Bologna, Italy*, February 2011.
- [8] F. Idzikowski, E. Bonetto, and L. Chiaraviglio, “EWA – an adaptive algorithm using watermarks for energy saving in IP-over-WDM networks,” Tech. Rep. TKN-12-002, Technical University of Berlin, Telecommunication Networks Group, May 2012.
- [9] “SNDlib: library of test instances for Survivable fixed telecommunication Network Design.” <http://sndlib.zib.de/home.action>, February 2012.
- [10] P. N. Tran and U. Killat, “Dynamic reconfiguration of logical topology for WDM networks under traffic changes,” in *Proc. of the IEEE Network Operations and Management Symposium (NOMS), Salvador, Brazil*, pp. 279–286, April 2008.
- [11] F. Idzikowski, “Power consumption of network elements in IP over WDM networks,” Tech. Rep. TKN-09-006, Technical University of Berlin, Telecommunication Networks Group, July 2009.