Evaluating the Impact of Energy-Aware Routing on Software-Defined Networking Performance

Adriana Fernández-Fernández, Cristina Cervelló-Pastor, Leonardo Ochoa-Aday
Department of Network Engineering,
Universitat Politècnica de Catalunya
Esteve Terradas, 7, 08860, Castelldefels, Spain.
{adriana.fernandez, cristina, leonardo.ochoa}@entel.upc.edu

Abstract—**Increasing power consumption and CO$_2$ emissions generated by large data networks have become a major concern over the last decade. For this problem, the emerging paradigm of Software-Defined Networks (SDN) can be seen as an attractive solution. In these networks an energy-aware routing model could be easily implemented leveraging the control and data plane separation. This paper investigates the impact of energy-aware routing on SDN performance. To that end, we propose a novel energy-aware mechanism that reduces the number of active links in SDN with multiple controllers, considering in-band control traffic, i.e. links are shared between data and control plane traffic. The proposed strategy exploits knowledge of the network topology combined with traffic engineering techniques to reduce the overall power consumption. Therefore, two heuristic algorithms are designed: a static network configuration and a dynamic energy-aware routing. Significant values of switched-off links are reached in the simulations using real topologies and demands data. Moreover, obtained results confirm that crucial network parameters such as control traffic delay, data path latency, link utilization and TCAM occupation are affected by the performance-agnostic energy-aware model.**

Keywords—**Software-Defined Networking, energy-aware routing, in-band control traffic, heuristic algorithms.**

I. INTRODUCTION

Recently, the growing energy consumption of Information and Communication Technologies (ICT) has attracted the attention of the networking researchers. According to [1] in 2012, close to 4.7% of the world's electrical energy was consumed by ICT, releasing into the atmosphere roughly 1.7% of the total CO$_2$ emissions. Moreover, recent studies state that energy demand of ICT sector is growing faster than the overall one and the power consumption of the global Internet could rise to more than 10% of the world's electricity consumption by 2025 [2]. This implies that the reduction of power consumption in Internet Service Provider (ISP) backbone networks is crucial to accomplish significant energy savings in this sector. At the same time, increasing the energy efficiency in data

networks can also substantially reduce the environmental impacts of other sectors.

Given that energy consumption of network equipment is only slightly influenced by their traffic load, an effective strategy to minimize the consumption of data networks is to reduce the number of active elements [3]. This feature can be implemented by putting into a low-power state (sleep mode) elements such as line cards or port interfaces that are not in use. Although turning off entire interconnection devices enables greater energy savings, in this work we do not consider this possibility because of resiliency concerns in case of network events. However, due to typical over-provisioning considered in the design and operation of backbone networks, considerable energy savings could be reached changing the status of network interfaces to sleep mode whenever a link is not transferring data.

Within this context, a promising solution for this problem is the use of Software-Defined Networking (SDN) [4]. The basic idea of SDN is to decouple control and data planes to make network environments more manageable. The logically centralized control plane in SDN has a global knowledge of network state information. Furthermore, it can manage network tasks and perform device programming without the need of additional software or hardware in each one of the switching elements. Meanwhile, network devices only forward traffic according to the rules set by the controller. This feature can be leveraged to perform an energy-aware routing that determines, in a coordinated and centralized way, the switch interfaces that should be put to sleep mode. Therefore, an energy-aware solution could be easily implemented in the control plane.

Despite consistent efforts to improve the network power efficiency, these techniques lead to performance degradations when QoS requirements are neglected. Inspired by this reality, this paper introduces a new energy-aware strategy and evaluate its impact on crucial performance metrics. Instead of restricting the path selection and po-

tential improvements in terms of energy efficiency to meet some specific metric bound, this work aims to quantify performance concerns of a fundamental research topic in recent communication networks.

Throughout this work we consider a SDN architecture with multiple controllers and, similar to our previous works, [5] and [6], in-band control traffic. This means that control messages are exchanged using the same links that data traffic without the need of additional edges. In this way, the energy-aware routing performance can be analyzed when, for physical and cost-related restrictions, implementing a dedicated control network is not feasible. Furthermore, this is a more realistic scenario for large backbone networks, where additional links dedicated to transfer the control messages between controllers and forwarding devices are impractical and cost-inefficient.

Specifically, the major contributions of this paper are as follows:

- We develop a novel energy-aware mechanism that reduces the number of active links in SDN with multiple controllers, considering that links are shared between data and control plane traffic.
- Two solution modules were conceived, exploiting knowledge of the network topology and traffic engineering techniques to reduce the overall power consumption.
- Using real topologies and traffic demands, we provide a performance comparison analysis of our proposal with another routing approaches.

The rest of this paper is structured as follows. In Section II we further discuss previous studies about different strategies to tackle the problem of power consumption. In Section III we explain the main characteristics of our energy-aware approach together with the description of its two comprised modules. The simulations strategies and the obtained results are presented and analysed in Section IV. Finally, in Section V we conclude our work and outline future research guidelines.

II. RELATED WORKS

Energy-aware techniques that reduce the number of active elements in the network can be divided into traffic-based and topology-based solutions, according to the elements considered in the model. In this section we analyze in more details works that deal with each one of these two approaches.

A. Traffic-Based Solutions

Under some assumption of expected traffic behaviour, traffic-based solutions are routing mechanisms that aggregate traffic over a subset of links and devices in over-provisioned networks, in order to switch off the unused network components.

For instance, Zhang et al. in [7] propose GreenTE, an intra-domain, centralized traffic engineering mechanism that finds a set of links that can be turned off under a given traffic load or matrix. The approach is based on a Mixed-Integer Linear Programming (MILP) formulation where the traffic demands are routed through a set of previously computed k-shortest paths. Performance requirements such as maximum link utilization (MLU) and network delay are considered as constraints in the problem. However, the implementation of such coordinated strategy is a difficult task given the distributed nature of network control in traditional networks.

More recently, in [8] the authors introduced a state-of-the-art study of energy efficiency strategies in SDN. This paper addresses the importance of implementing green routing methods in SDN, taking advantage of the flexibility given by dynamic configuration and centralized network view capabilities. A summary of some existing energy-aware techniques in SDN with their key properties (benefits and drawbacks) is presented.

The authors of [9] addressed the problem of saving energy in partially deployed SDN. They formulated an optimization problem for finding minimum power network subsets in these hybrid networks. Giroire et al. [10] proposed an energy-aware routing approach that takes into account the limited rule space of TCAM in SDN devices. An ILP model is presented as well as an efficient heuristic. The authors of [11] provided two greedy algorithms for minimizing the power of integrated chassis and line-cards used. To achieve this they considered an expanded network topology according to the connections between forwarding devices. Nevertheless, in all these works, dedicated links between the controller and SDN nodes were considered.

In [12] the authors proposed a model for controller-switch associations that aims to maximize the energy efficiency of the network. Although the routing of control traffic is considered in this work, they assume that controllers act as well as forwarding devices, i.e. data plane communications are routed through network controllers. Therefore, only links that belong to control paths are activated and data traffic demands are routed using these links until a MLU bound is reached. We argue that data plane traffic should not pass through network controllers, since this will represent an additional load in these devices.

The work in [5] addressed the problem of minimizing the number of required links in large-scale SDN with in-band control traffic. To accomplish this, an ILP model and a heuristic algorithm are presented, integrating the routing requirements for data and control traffic. This model also determines an optimal distribution of switches between controllers in terms of energy efficiency and load balancing. In [6] a distributed routing algorithm that optimizes the power consumption in large-scale SDN with multiple domains is proposed. The solution, called DEAR (Distributed Energy-Aware Routing), tackles the problem of minimizing the number of links that can be used to satisfy a given traffic matrix. Despite being efficient models, the complexity of considering the entire topology for the selection of the most suitable routes can be very expensive in networks with major path redundancy. By the contrary, in this work, after pruning the network topology, the number of paths and the consequent computation complexity are significantly reduced.

B. Topology-Based Solutions

The lack of awareness of traffic conditions in typical operative networks has led to several research works that, in order to reduce the number of active links, are oriented to control the network topology. Basically, these approaches modify the existing topology considering different requirements such as the resulting connectivity.

In [13] the authors present an OSPF-based routing mechanism that considers the topological information exchanged among routers. The proposed EAR algorithm is based on the definition of the "exportation" mechanism where a Shortest Path Tree (SPT) is shared between neighbour nodes. The routers with the highest node-degree, called "exporters", calculate the SPTs that are used to route the traffic and force the use of these paths to all their neighbors, so that the overall set of active links can be reduced. The exportation mechanism is enhanced in [14], where the concept of "move" was introduced turning the energy saving routing problem into a formulation of the well-known Maximum Clique Problem in an undirected weighted graph.

Authors in [15] propose a routing algorithm called Energy Saving based on Algebraic CONnectivity (ESACON), using the algebraic connectivity as a metric to control the resulting network topology. Based on this metric, ESACON is able to identify and switch off the network links that less affect the network connectivity, keeping this value over a given threshold.

Similarly, the topology-based solution reported in [16] also takes into account the algebraic connectivity as a requirement to preserve the overall network connectivity. This work also considers the edge betweenness as a metric to measure the links role in the network, placing the links least frequently used as the first candidates to be pruned. However, this approach is conceived to be implemented in a distribute way into each IP router.

The work in [17] also aims to improve the energy efficiency reducing the number of active links. For this purpose, the authors propose four different versions of the Energy Saving based on Occurrence of Links (ESOL) algorithm that show the tradeoff between complexity and efficiency in powering off a great number of links. The parameters used in this approach to select the network interfaces to be switched off are the occurrences of nodes and links in network paths.

The analysis of including QoS requirements in an energy-aware topology-based solution is discussed by the authors of [18]. Their approach, called Energy Saving IP Routing (ESIR), is also based on the concepts of SPT exportation and move but constrained to a maximum load boundary on network links.

All the previously described works mainly tend to minimize the number of active network elements in the current topology restricting the path selection to meet some specific metric bound or connectivity rate and fail to extensively examine the impact of energy-aware routing on SDN performance. Moreover, their lack of awareness about the requirements of incoming connection requests
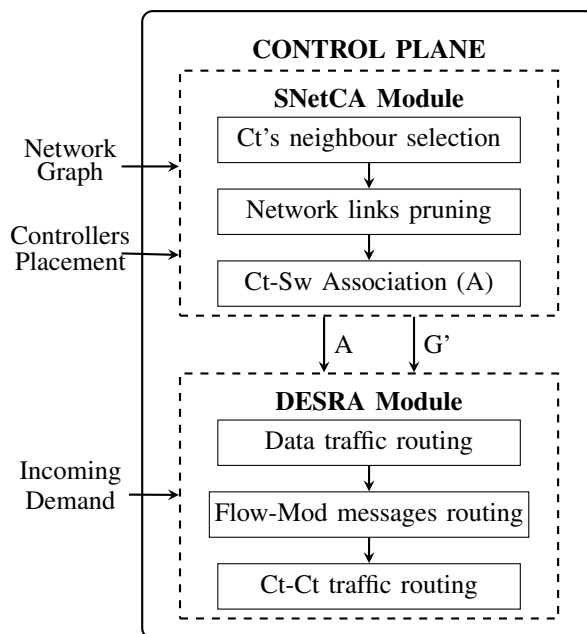

Fig. 1. Illustrative diagram of the proposed approach.

can lead to performance degradations, which is highly undesired.

Different from the aforementioned works, the aim of this paper is to provide a low-complexity energy-aware strategy which will be used to evaluate its impact on crucial performance metrics, considering a SDN architecture with multiple controllers and in-band control traffic.

III. ENERGY-AWARE APPROACH

In this work, we present a hybrid solution for the energy efficiency problem in SDN comprising the main advantages of the two aforementioned approaches. More precisely, we exploit specific network topological properties combined with the use of traffic engineering to reduce the overall power consumption. An illustrative diagram of this strategy is shown in Fig. 1. The first component, denoted as Static Network Configuration Algorithm (SNetCA), is a topology-based solution intended to be statically activated as a planned operation. On the other hand, the traffic-based module, denoted as Dynamic Energy Saving Routing Algorithm (DESRA), is activated by the arrival of each incoming traffic demand. Therefore, an accurate prediction of incoming traffic is not needed.

In essence, this approach finds the routes between network elements that minimize the number of active links used, considering that links are shared between data and control plane traffic (i.e. in-band mode). Therefore, control paths between controllers and switches (in both senses) and between controllers are also established.

Additionally, given the controllers placement in the network topology, our model determines the ideal distribution of switches between controllers in terms of energy efficiency, considering as well a balanced load between controllers. In our energy-aware approach, the routing of additional traffic load through the controllers is avoided,

i.e., admissible control paths do not pass through any other controller that is not the source or target of the traffic and data plane communications cannot be routed through these devices.

The two main parts enclosed within the proposed energy-aware approach are described in more details in the following subsections.

A. Static Network Configuration Algorithm

In the proposed scheme the network topology can be modeled as a directed graph $G = (V, E, C)$, where V, E and C denote the set of nodes, links and controllers respectively, being $C \subset V$. We define the set of interconnection devices as $S = \{n \mid n \in V \wedge n \notin C\}$. We use X to denote the set of active links $X \subseteq E$ and U to store the utilization of network links.

By considering the typical link redundancy of backbone networks, we design a Static Network Configuration Algorithm, denoted as SNetCA, which aims to prune as many links as possible in order to stress the importance of energy saving. Additionally, the most favorable switch-controller associations in terms of energy efficiency and load balance, are determined in this stage.

The algorithm, described in the Algorithm 1 pseudocode, is composed of three steps:

1) selecting one of the controller's neighbours, as the node that will remain connected to it in the outcome topology;
2) identifying the links that do not disconnect the graph to be put into sleep mode;
3) associating each node with one controller and computing the control path between them.

The input of the algorithm is the network topology with controllers placement and its outputs are a pruned network with a reduced number of links, denoted as G', an array keeping the controller-switch associations, denoted as A and the control paths from each node to its controller, denoted as P_{sc}.

In the first step, the algorithm iterates over the set of network controllers in order to evaluate each one of its neighbours. The selection of one neighbour node for each controller is based on the betweenness centrality (B_n), which measures the intermediary role of a node in the network. In the proposed approach, we use a simplified version of this metric considering only the shortest paths from a controller to every switch.

In particular, after computing the shortest paths from one controller as single source, the algorithm determines whether a neighbour node belongs to each path and increases the B_n associated with that node (lines 6-16). For each controller a list of neighbour devices, sorted in decreasing order of B_n, is stored in L. This list is used to identify the available neighbour with the highest betweenness centrality. This node is associated with the considered controller in the current iteration and stored in A, as long as it has not been already attached to another controller.

For the remaining nodes in L, the links between them and the controller are removed from the resulting network

---

Algorithm 1 SNETCA

Require: $G = (V, E, C)$ network graph with controller placements

Ensure: $G' = (V, E', C)$ network graph with reduced number of links, A controller-switch associations, P_{sc} switch-controller control paths

1: $N_c \leftarrow$ Set of neighbours of controller $c \in C$
2: $G' \leftarrow G$
3: **for** $c \in C$ **do**
4:      $B \leftarrow$ NULL   ▷ Array of betweenness values
5:      $SP_c \leftarrow$ Set of shortest paths from controller $c \in C$
6:      **for** $n \in N_c$ **do**
7:          **if** $n \in C$ **then**
8:              continue
9:          **end if**
10:         $B_n = 0$
11:         **for** $p \in SP_c$ **do**
12:            **if** $n \in p$ **then**
13:               $B_n = B_n + 1$
14:            **end if**
15:         **end for**
16:      **end for**
17:      $L \leftarrow B\_Sorted$
18:      **for** $s \in L$ **do**
19:          **if** s and c not already in A **then**
20:              $A = A \cup (s, c)$
21:          **end if**
22:          Remove links (s, c) and (c, s) from G'
23:      **end for**
24: **end for**
25: **for** $i, j \in E'$ **do**
26:      **if** $i \in C$ or $j \in C$ **then**
27:          continue
28:      **end if**
29:      $G'' \leftarrow G'$
30:      Remove controllers $c \in C$ from G''
31:      Remove link i, j from G''
32:      **if** G'' remains strongly connected **then**
33:          Remove link i, j from G'
34:      **end if**
35: **end for**
36: **for** $s, c \in A$ **do**
37:      PATHSELECTOR(s, c)
38:      Update P_{sc}, X, U
39: **end for**
40: **for** the rest of $s \in S$ **do**
41:      PATHSELECTOR(s, C)
42:      Update P_{sc}, A, X, U
43: **end for**

---

graph. This means that they are put into sleep mode in the original graph. Notice that when a controller's neighbour is another controller, the link between them is not considered as a candidate to be pruned (lines 7-9).

In the next step, the algorithm iterates over the set of directional links in the pruned network that do not have any controller as its extreme nodes. At each iteration the

---

Algorithm 2 PATHSELECTOR(a, b)

1: $L \leftarrow \infty$
2: $SeP \leftarrow None$
3: **for** $p \in$ Get_All_Adimissible_Paths(G', a, b) **do**
4:     **if** $b = C$ **then**
5:         **if** p is to an already loaded controller **then**
6:             continue
7:         **end if**
8:     **end if**
9:     $off \leftarrow$ number of links in p that are not in X
10:     **if** $off < L$ and p has sufficient bandwidth **then**
11:         $L \leftarrow off$
12:         $SeP \leftarrow p$
13:     **end if**
14: **end for**

Algorithm 3 DESRA

Require: G', A, d incoming traffic request
Ensure: P_{ss}, P_{cs}, P_{cc} data and control paths, X active links, U links utilization
1: $Ct_1 \leftarrow A[s_d]$
2: $p =$ PATHSELECTOR(s_d, t_d)
3: Update P_{ss}, X, U
4: **for** $n \in p$ **do**
5:     $Ct \leftarrow A[n]$
6:     PATHSELECTOR(Ct, n)
7:     Update P_{cs}, X, U
8:     **if** $Ct_1 \neq Ct$ **then**
9:         PATHSELECTOR$(Ct1, Ct)$
10:         Update P_{cc}, X, U
11:     **end if**
12: **end for**

---

algorithm attempts to increase the number of switched-off edges.

A new link is removed only when the resulting graph remains being strongly connected, i.e. at least one path exists between every pair of nodes in the network. To accomplish this, a temporal graph without any controller, denoted as G'', is created. This graph is used to check the required connectivity between all the forwarding devices. After validating that the possibility to reach any node in the network is not affected, the considered link is removed from the resulting graph.

The last step of the algorithm is intended to determine a control path from each forwarding device to one controller. To achieve this goal, the algorithm starts evaluating the pairs of controller-switch associations already stored in A (line 36). For each pair, an admissible control path minimizing the number of active links is selected using the method PATHSELECTOR described in Algorithm 2, which will be further explained below. As stated previously, admissible control paths do not pass through any other controller that is not the source or target of the traffic. The remaining forwarding devices are then considered. Notice that in this case the algorithm takes into account the control paths to all controllers in the network. Precisely, the path computed by the PATHSELECTOR in this step defines the controller for the rest of forwarding devices.

Using this initial control plane configuration, switches send to the controller packet_in requests when a new traffic flow arrives, as well as statistics and failure notifications. Consequently, there is an initial set of active links in the network before the ingress of traffic flows as well as some link utilization.

The PATHSELECTOR method, described in Algorithm 2, performs the energy-aware path selection. In essence, this function is used to select the best admissible route between a pair of nodes, in terms of minimizing the number of active links in the network. The key idea of this function is to perform a low-complexity greedy evaluation between all the admissible paths to select the most suitable route in terms of energy-efficiency, while guaranteeing a balanced load of switches between controllers and capacity con-

straint of links. Since this method works over the pruned network with a reduced number of links, (i.e. G'), the set of admissible paths considered is significantly smaller than in the original topology and the solution can be found with fewer iterations. When this function is called for determining the path between each forwarding device and one controller (i.e. using the set of controllers as the traffic destination) the controller load is considered (line 4 to 8). In addition, the path only can be selected if it has sufficient link capacity to route the required traffic volume.

B. Dynamic Energy Saving Routing Algorithm

When a new traffic demand arrives, a routing request is sent from the incoming node to its associated controller using the path between both devices previously computed during the static network configuration phase. Based on its global knowledge of the network topology, this controller calculates the required data path minimizing the number of links that need to be activated for this connection request and creates the flow forwarding rules. Given the multidomain scenario considered, the nodes traversed by the data traffic may be associated with different controllers.

The proposed dynamic energy-aware routing is shown in Algorithm 3. For an incoming demand d from source s_d to destination t_d, the algorithm starts storing in Ct_1 the controller associated with the source node. This controller is the main responsible of managing this traffic request. Using the PATHSELECTOR method, the most favorable admissible data path in terms of energy consumption is computed. This is done considering that admissible paths do not pass through any controller in the network. Then, a loop is used to consider the required control plane communications for each node along this path.

After determining the controller associated with each node in the data path, a control path is computed between them. These paths are used to set the flow forwarding rules in each switch using the flow_mod messages. When a node is not associated with Ct_1 an additional control message is sent from this controller to the other, in order to inform the second controller of the flow forwarding rule that need to be installed in one of its managed nodes.
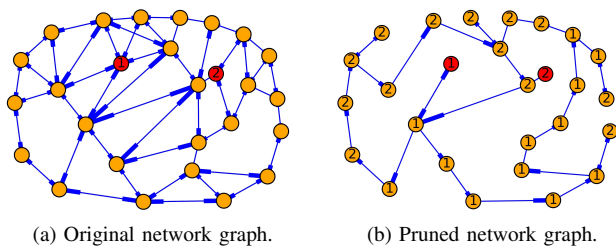
(a) Original network graph.   (b) Pruned network graph.

Fig. 2.   SNetCA performance on the Norway topology.

Fig. 3.   Average number of pruned links in the three topologies varying the number of controllers.

IV. SIMULATIONS AND RESULTS

In this section we describe the evaluation of our energy-aware approach and analyze the results obtained. The proposed control framework described in Section III was implemented using the programming language Python to develop the heuristic algorithms. All computations were carried out on a computer equipped with 3.30 GHz Intel Core i7 and 16 GB RAM. We conducted our simulations using real network topologies and traffic demands collected from SNDlib [19], considering each router in the network as a SDN node or as a possible controller placement.

Specifically, we use three of the most link-redundant network topologies in SNDlib in order to asses the effectiveness of the proposed scheme. The mentioned topologies are: Geant ($|V| = 22$; $|E| = 72$), New York ($|V| = 16$; $|E| = 98$) and Norway ($|V| = 27$; $|E| = 102$). For the control traffic we assume an average rate of 1.7 Mbps [20]. To analyse the performance of our energy-aware approach we present the following evaluations for different amount of controllers in the network.

A. SNetCA performance

In order to evaluate the effectiveness of the proposed topology-based module, Fig. 2 shows an example of the performance of SNetCA on the Norway topology, considering two network controllers placed at nodes denoted as 1 and 2 and emphasized with a different color in the figure. The distribution of switches between controllers is depicted through the use of labels in each node, indicating the controller number to which the node is associated.

A comparison between the original network and the resulting graph illustrated in Fig. 2(a) and Fig. 2(b) respectively, shows a difference of 67 edges, which represents more than 65% of total network links. These links are pruned by our algorithm guaranteeing that the resulting graph remains being strongly connected and avoiding additional traffic load through network controllers.

Additionally, as a result of applying SNetCA on the Norway topology, switches are distributed between controllers minimizing the number of required active links and ensuring a balanced controllers load. For instance, 12 switches are associated with controller 1 while the remaining 13 are managed by controller 2.

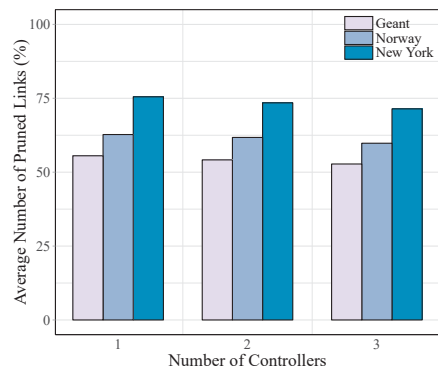To provide a more general perspective, Fig. 3 shows, for the three considered topologies, the average number of links pruned by SNetCA. In this analysis we consider all the admissible placements of 1 to 3 controllers. Notice that a controller placement is admissible when the assumptions established in this proposal to avoid the routing of additional traffic load through network controllers can be kept (i.e., the network graph without any controller is strongly connected). As it is shown, a high number of links is pruned in all the topologies considered, which contributes directly with the energy efficiency achieved by this proposal. In general, the more redundant the network, the higher number of links can be put to sleep mode applying this strategy.

B. Impact of DESRA on Network Performance

It is to be emphasized that in our energy-aware approach quality of service (QoS) constraints and performance metric boundaries are not taken into account. This is not a limitation but a choice; since we intend to measure the impact of our proposal on the network performance metrics as a trade-off with the energy saving improvements. In fact, we are presenting an effective and easy to implement green routing mechanism that emphasizes the importance of energy efficiency in the operation of current data networks.

In order to assess the impact of our energy-aware approach on the network performance, we adapt two well-known state-of-the-art routing algorithms: Shortest Path Routing (SPR) and Load Balancing (LB) for their use in the considered in-band SDN environment. Being the rule space a significant issue of concern in SDN, an algorithm balancing the number of rules installed in each forwarding device, denoted as TCAM Occupation Balancing (TOB), is also included in this analysis. In essence, these algorithms are greedy heuristics that prioritize some performance metric such as: traffic latency, link utilization or TCAM occupation used in our evaluation as baselines for comparison purposes. All of them follow the assumptions established in this proposal to avoid the routing of additional traffic load through network controllers. SNetCA is still used to determine the distribution of switches between controllers.

Due to space limitation, for the different performance metrics, we may focus our attention on some specific network, but similar results have been obtained for all the considered topologies.
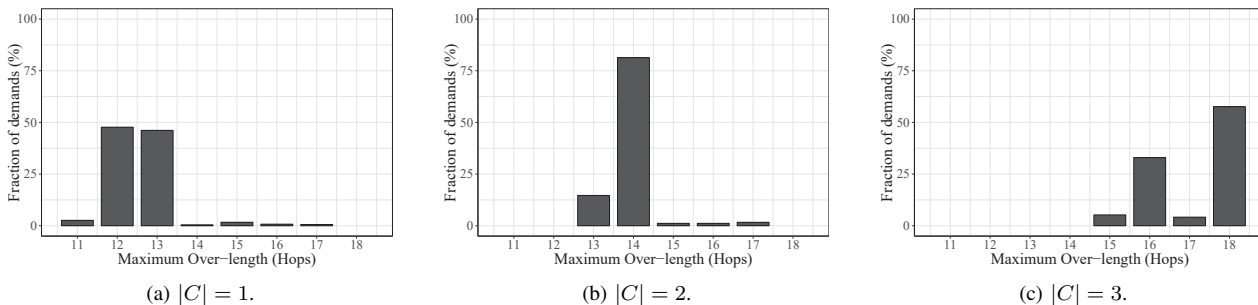
(a) $|C| = 1$.

(b) $|C| = 2$.

(c) $|C| = 3$.

Fig. 4. Distribution of maximum control traffic over-length in the Norway topology for different amount of controllers.

(a) Geant Topology.
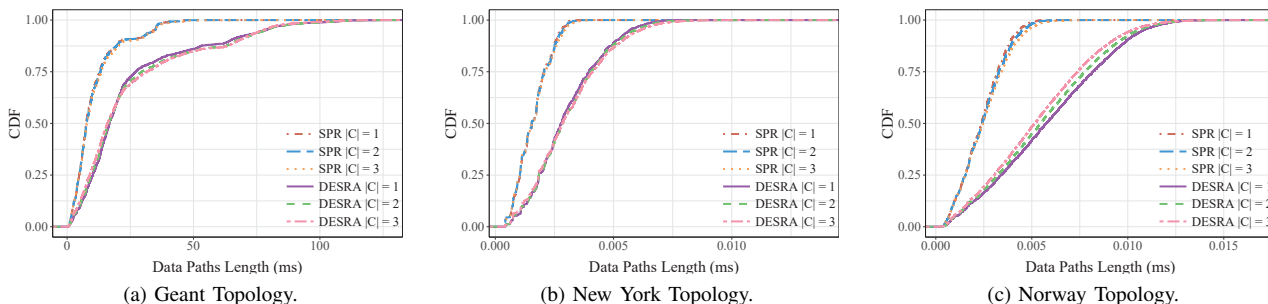
(b) New York Topology.

(c) Norway Topology.

Fig. 5. CDF of data paths latency for three different topologies varying the amount of controllers.

1) Traffic Latency: In a first set of simulations, we analyze how the data and control paths latency is affected by the routing decisions made.

To evaluate the impact of our algorithm on control path delay, we collect, for each traffic demand, the length of its associated control paths and the corresponding shortest path. The notation Maximum Over-length is used to denote the maximum difference (in number of hops) between the length of the routing solution and the shortest path. Fig. 4 shows this behaviour for the Norway topology considering all possible placements for different amount of controllers. As it is shown, when the number of controllers grows, the control traffic is routed using a larger number of hops for a higher fraction of demands.

To take a closer look at the data plane, we draw in Fig. 5 the cumulative distribution function (CDF) of data paths latency for three different topologies considering all possible locations of one to three controllers. As shown in Fig. 5(a), Fig. 5(b) and Fig. 5(c), the CDFs of data paths latency for different amount of controllers are quite similar. However, we can see that under the energy-aware routing, the control path delay is affected in order to minimize the number of active links. For instance, in Fig. 5(a), only 87% of data paths exhibit delays lower than 50 ms, meanwhile all control paths in the SPR case are under this value. In general, the larger the network (in terms of geographic length), the more increase in latency is incurred.

2) Links Utilization: The selection of routing paths minimizing the energy consumption has a direct influence in the traffic load of all the network links. To better showcase this situation, we use the Geant topology and the LB algorithm. Fig. 6 provides the CDF of link utilization

under both algorithms considering all possible locations of one to three controllers in this topology. As expected, the fairness of traffic distribution is altered by the energy-aware routing, since there is a subset of active links that is more overloaded than the others. Nevertheless, even in the more loaded cases the link utilization is under 60% in this topology.

3) TCAM Occupation: Intuitively, an energy-aware routing would affect the allocation of flow rules since traffic flows are redirected to minimize the number of active links. In Fig. 7, we evaluate the impact of our approach on TCAM occupation with respect to the TOB algorithm using the Geant topology and all possible locations of three network controllers. As expected, the number of installed rules is raised by the energy-aware routing in almost all the network devices (18 out of 22 nodes), being in some cases more than twice the value obtained by the TOB algorithm. However, the DESRA performance in this topology is still physically acceptable considering that a routing table can support from 750 to few thousands of rules [10].

4) Energy savings: To get a sense of the energy saving values achieved by our approach, Fig. 8 shows the average energy performance of all the considered routing models in the New York topology for the case of one centralized controller in the network. The energy savings were computed as the number of links in sleep mode over the total amount of network links. As expected, in all cases energy saving decreases while the number of demands grows, since new paths need to be established. Furthermore, the proposed routing algorithm outperforms SPR, LB and TOB in terms of energy saving. In general, DESRA achieves significant energy savings but bigger improvements with respect to the other approaches are reached when the traffic grows.
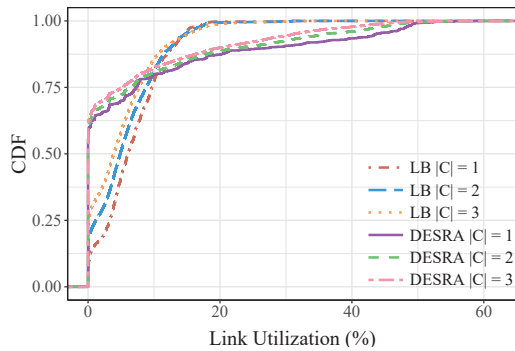
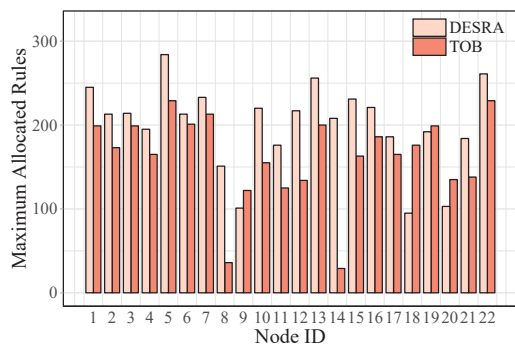Fig. 6.   CDF of link utilization in the Geant topology.

Fig. 8.   Average energy savings in the New York topology with $|C| = 1$.

Fig. 7.   Average TCAM occupation in the Geant topology with $|C| = 3$.

V. CONCLUSION

In this paper, we evaluate the performance impact of applying an energy-aware routing on SDN with in-band control traffic. To achieve such goal, we have proposed a hybrid approach comprising two heuristic algorithms: a static network configuration and a dynamic energy-aware routing. Apart of providing an effective power-aware scheme able to achieve notable improvements in terms of energy saving, the most significant added value of the proposal is the quantitative analysis presented of such significant networking concern. Extensive simulations using real topologies and traffic matrices validate that crucial network parameters such as control traffic delay, data path latency, link utilization and TCAM occupation are affected by the performance-agnostic energy-aware model. These findings confirm that energy-aware routing schemes should be designed considering specific traffic requirements and performance metric bounds. As future work, we want to provide an analysis on the impact of reducing the number of active network elements on SDN reliability, considering in-band control traffic.

ACKNOWLEDGMENT

REFERENCES

[1] E. Gelenbe and Y. Caseau, "The Impact of Information Technology on Energy Consumption and Carbon Emissions," *ACM Ubiquity*, vol. 2015, no. June, pp. 1–15, 2015.
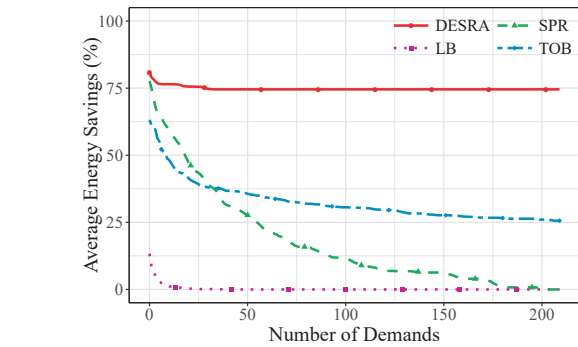
[2] R. S. Tucker, "Energy consumption in telecommunications," in *Proc. Optical Interconnects Conference*, May 2012, pp. 1–2.
[3] M. Gupta and S. Singh, "Greening of the Internet," in *Proc. ACM SIGCOMM'03*, 2003, pp. 19–26.
[4] D. Kreutz, F. M. Ramos, P. Verissimo, C. Esteve Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-Defined Networking: A Comprehensive Survey," *Proceedings of the IEEE*, vol. 103, pp. 14–76, 2015.
[5] A. Fernandez-Fernandez, C. Cervello-Pastor, and L. Ochoa-Aday, "Achieving Energy Efficiency: An Energy-Aware Approach in SDN," in *IEEE GLOBECOM'16*, Dec. 2016, pp. 1–7.
[6] ——, "Energy-Aware Routing in Multiple Domains Software-Defined Networks," *ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal*, vol. 5, no. 3, 2016.
[7] M. Zhang, C. Yi, B. Liu, and B. Zhang, "GreenTE: Power-Aware Traffic Engineering," in *Proc. IEEE ICNP'10*, 2010, pp. 21–30.
[8] B. G. Assefa and O. Ozkasap, "State-of-the-art Energy Efficiency Approaches in Software Defined Networking," in *Proc. SoftNetworking'15*, Apr. 2015.
[9] H. Wang, Y. Li, D. Jin, P. Hui, and J. Wu, "Saving Energy in Partially Deployed Software Defined Networks," *IEEE Trans. Comput.*, vol. 65, no. 5, pp. 1578–1592, 2016.
[10] F. Giroire, J. Moulierac, and T. K. Phan, "Optimizing Rule Placement in Software-Defined Networks for Energy-Aware Routing," in *Proc. IEEE GLOBECOM'14*, 2014, pp. 2523–2529.
[11] R. Wang, Z. Jiang, S. Gao, W. Yang, Y. Xia, and M. Zhu, "Energy-Aware Routing Algorithms in Software-Defined Networks," in *Proc. IEEE WoWMoM'14*, June 2014, pp. 1–6.
[12] A. Ruiz-Rivera, K. W. Chin, and S. Soh, "GreCo: An Energy Aware Controller Association Algorithm for Software Defined Networks," *IEEE Commun. Lett.*, vol. 19, no. 4, pp. 541–544, 2015.
[13] A. Cianfrani, V. Eramo, M. Listanti, M. Marazza, and E. Vittorini, "An Energy Saving Routing Algorithm for a Green OSPF Protocol," in *Proc. IEEE INFOCOM'10*, Mar. 2010, pp. 1–5.
[14] A. Cianfrani, V. Eramo, M. Listanti, and M. Polverini, "An OSPF enhancement for energy saving in IP networks," in *Proc. IEEE INFOCOM WORKSHOPS'11*, Apr. 2011, pp. 325–330.
[15] F. Cuomo, A. Abbagnale, A. Cianfrani, and M. Polverini, "Keeping the connectivity and saving the energy in the internet," in *Proc. IEEE INFOCOM WORKSHOPS'11*, Apr. 2011, pp. 319–324.
[16] F. Cuomo, A. Cianfrani, M. Polverini, and D. Mangione, "Network Pruning for Energy Saving in the Internet," *Comput. Netw.*, vol. 56, no. 10, pp. 2355–2367, July 2012.
[17] F. Cuomo, A. Abbagnale, and S. Papagna, "ESOL: Energy saving in the Internet based on Occurrence of Links in routing paths," in *Proc. IEEE WoWMoM'11*, June 2011, pp. 1–6.
[18] A. Cianfrani, V. Eramo, M. Listanti, M. Polverini, and A. V. Vasilakos, "An OSPF-Integrated Routing Strategy for QoS-Aware Energy Saving in IP Backbone Networks," *IEEE Trans. Netw. Service Manag.*, vol. 9, no. 3, pp. 254–267, Sept. 2012.
[19] S. Orlowski, M. Pióro, A. Tomaszewski, and R. Wessäly, "SNDlib 1.0-Survivable Network Design Library," *Networks*, vol. 55, no. 3, pp. 276–286, 2010.
[20] J. Li, J.-H. Yoo, and J. W.-K. Hong, "Dynamic control plane management for software-defined networks," *International Journal of Network Management*, vol. 26, no. 2, pp. 111–130, 2016.