Exploiting Segment Routing and SDN Features for Green Traffic Engineering

Chung-Horng Lung
Department of Systems and Computer Engineering
Carleton University, Ottawa, Canada
chlung@sce.carleton.ca

Hesham ElBakoury
Futurewei Technologies
San Jose, CA, USA
helbakoury@gmail.com

Abstract—Energy efficiency for network devices becomes an important topic, as they consume a significantly amount of energy. Various techniques have been proposed to address energy-aware traffic engineering (TE), including Segment Routing (SR) and Software-defined Networking (SDN), which provide lower operational complexity and higher flexibility. However, existing approaches have not exploited some evolving SR and SDN features for efficient TE, e.g., path computation, sub-50 *msec* protection, and local/global segments. Consequently, those approaches result in higher complexity or extra overhead. This paper provides a holistic view of green TE using evolving SDN and SR-specific features without adding much additional computational tasks, and also considers SR segment processing overhead for energy efficiency. The proposed approach can simplify green TE by reusing SR features and improve energy efficiency and robustness.

Keywords— Segment routing, traffic engineering, software-defined networking, energy efficiency, path calculation element

I. INTRODUCTION

Internet traffic has increased rapidly and the trend is growing in a fast pace. As a result, a large number of network devices have been deployed and the processing power of those devices have also increased considerably. Further, redundant network devices and links have been deployed in the field for robustness. Consequently, the network devices consume a huge volume of energy, which raises concerns for the environment [1].

Many energy-aware or green traffic engineering (TE) approaches have been proposed to deal with the energy consumption problem for different types of networks, e.g., wide area networks (WANs) [2] [3], Ethernet [4] [5], data center networks [6] [7] [8], using techniques at different layers, e.g., physical layer and network layer. This paper focuses on routing techniques at the network layer for WANs. One popular approach to the problem area is to reroute traffic to reduce active ports (links) or nodes. The idea of this approach is backed up by two main factors: (i) networks are often overprovisioned and link utilizations even during peak hours are not high, e.g., < 50% most of the time [2]; and (ii) line cards consume ~43% of the total power of the network devices like routers or switches [3].

Energy-aware TE has also been investigated for different network routing technologies, e.g., Interior Gateway Protocol (IGP)/IP [3], Multi-Protocol Label Switching (MPLS) [9] [10], and Segment Routing (SR) [2] [5] [11]. In addition, Software-defined networking (SDN) has been adopted in some works, e.g., [5] [11] [12] [13]. Our paper focuses on green TE using SR and SDN. The main reasons for choosing SR and SDN are: (i) SR supports explicit routing like MPLS without the complex operations of RSVP protocol [14] at the control plane; and (ii) SDN central controller provides a global view of the network, which reduces the complexity of distributed computing.

Network TE is a critical topic. Various factors need to be considered, including: (i) monitoring of traffic demands and network planning; (ii) modeling and control of flows for dynamic traffic; and (iii) minimizing the maximum link utilization (MLU) to accommodate traffic bursts [15]. Based on data traces, there are diurnal traffic patterns in the field [2] [3] [16]. Some ports, links, or even nodes could be put into sleep or powered off if the demand drops below a threshold. The concept is not new; variations of the model have been devised by most energy-aware TE approaches. Some green TE approaches also exploiting the traffic patterns by calculating a subset of the full topology to start the process [2] [3]. The problem can be formulated as a multi-commodity minimum cost flow problem and is known to be NP-hard, though heuristics have been adopted by some approaches [3] [11].

This paper presents a holistic view of energy-aware TE using SR and SDN. We also make use of existing traffic patterns and explicit routing for better traffic control and less rerouting. We advocate that such an approach to be first evaluated during off- or mid-peak hours, though it is not limited to those periods. Our proposed approach does not rely on a subset of the full topology, since diurnal traffic patterns exist, but traffic may still fluctuate significantly these days due to social media, e.g., sport games, a release of a popular video, or a sudden unexpected event.

In addition to managing switch on-off of network elements based on monitored utilization values that have been used for most existing approaches, we also exploit emerging SR-specific features, as SR has evolved quickly and becomes popular. Notably, those SR-specific features considered in our paper include global segment for shortest path (*SP*) forwarding, local segment for explicit routing, SR policy flexibility, backup path calculation for resilience (see Section III), Equal-Cost Multi-Path (ECMP) [17], and segment reduction and consolidation.

The rest of this paper is organized as follows. Section II describes related work. Section III discusses the main model and assumptions adopted for our proposed approach. Section IV presents algorithms. Finally, Section V is the conclusion.

II. BACKGROUND AND RELATED WORKS

This section highlights some works in TE, e.g., traditional MPLS, emerging SR, SDN, and energy-aware TE approaches.

A. *MPLS-TE and SDN*

MPLS has been widely used for TE for a number of years [18]. MPLS-TE assigns network traffic flows using dynamically defined labels to steer the traffic along pre-computed paths. Green TE based on MPLS has been proposed [9] [10]. However, the control plane signalling protocol, i.e., RSVP-TE [14], for path computation is complicated and inefficient as the paths are created via a hop-by-hop manner for each path and changes of existing path reservations also need to be re-signalled.

SDN has attracted a great deal of attention. One key feature of SDN is its physical separation of the control and data planes for network devices, which enables fast development of network functions and greatly reduces the operational complexity compared to the traditional network. Using SDN, RSVP is no longer needed, as the central controller can calculate paths using Path Computation Element (PCE) [19], which has much lower overhead. In addition, PCE has been extended for SR. Similarly, BGP is another protocol that is critical for SR-TE. But both PCE and BGP have not been considered much in existing green TE methods.

B. *Segment Routing*

SR has become popular in industry and academia [17] [20] [21]. SR simplifies the control plane used in MPLS by removing RSVP. But, similar to the MPLS data plane the source of the traffic uses predefined labels or segments for a particular flow along a path or part of a path. A segment can be either a 32-bit MPLS label or a 128-bit IPv6 addresses (SRv6). This paper emphasizes on SRv6, as it has a set of evolving features which offer a number of advantages, including [17] [20] [21]:

- SR has lower overhead than MPLS, as sophisticated RSVP-TE operations and maintenance are eliminated.
- It is difficult to support network-wide ECMP using MPLS which needs a large number of tunnels and results in complex and the costly operations. On the other hand, SR is IP-based and does not need to maintain per-flow state, which has much lower overhead.
- PCE and BGP are seamlessly integrated with SR [17] [21], as PCE is often used for SR path computations and BGP routes can be installed to SR policy

SR adopts two types of Segment Identifier (SID): global prefix SID (Prefix-SID) and local adjacency ID (Adj-SID). Prefix-SIDs are known to all nodes in a SR domain, so that each node will install the instruction in its forwarding table. An Adj-SID is only meaningful to a node that originates it [17]. In addition, there is binding SID (BSID) which is a local segment, but it is used for steering labeled packets to a pre-configured SR policy. BSID can reduce the number of segments and provide service independence and isolation from the churn of the network [21]. See section III.B for an example of BSID.

Next, SR supports resiliency. When a link fails, packets are rerouted from the node that detects the failure without complex pre- and post-failure operations that are needed for MPLS fast reroute (FRR). A full explanation of these procedures, the types of labels, as well as various applications is given in [17].

On the other hand, from the energy efficiency perspective, when a link fails and a new or backup route is selected, if ECMP is also deployed in SR, this may result in a complex rerouting of flows and balancing of the traffic in the network. Further, using ECMP can have negative impact on energy efficiency, especially for off-peak hours, as the traffic amount is divided into smaller pieces on multiple links, each with low utilization.

When SR is integrated with SDN, the deployment is even easier, as the controller has global information of the entire network and PCE has been incorporated into the controller. However, the current concerns for SR-TE are TE efficiency, load balancing, and resiliency. Energy efficiency is not explicitly included in SR-TE, though networks are often over-provisioned.

In this research, we advocate using SR-TE techniques for energy saving. SR is flexible compared to IGP-based TE and has considerably lower overhead compared to MPLS-TE.

C. *Energy-Aware TE*

Green TE is not a new concept. A number of approaches have been reported in the literature, as highlighted in Section I. Some key concepts and related techniques that have been adopted in green TE approaches are listed as follows:

- Modeling technique: Linear programming [2] [3] [11]
- Routing technology: IP, MPLS, SR (see Section I)
- Network topology: full topology, a subset of the full topology [3], or a minimum spanning tree [11] [12]
- Traffic demands: traffic patterns may be used [2] [3] [11]
- Management of links/nodes based on utilizations

The control plane is either SDN Controller or the IP/MPLS control plane. Almost all TE approaches describe a mechanism to switch on-off network elements, links and nodes, based on the monitored utilization levels and model used.

III. SYSTEM MODEL AND ENERGY-AWARE PARADIGMS

This section highlights the system model for the proposed energy aware SR-TE approach and describes some assumptions that are adopted, including evolving SR-specific features.

- Critical network devices, e.g., edge nodes and links, are excluded in the algorithms pruning. Nodes in the network will not be disconnected as a result of shutting down any link or node, which is assumed to be part of the algorithms.
- A port on a switch is unidirectional. Shutting down (or hibernating) a port means powering down its corresponding hardware components from one end to the other end only, i.e., data can still be transmitted from the opposite port. For simplicity, we use port and link interchangeably.
- Based on measurements, the aggregated network follows periodic patterns [2] [3] [11] [15] [16], e.g., on-peak and off-peak patterns, and even mid-peak patterns. During the peak hours, it is challenging to apply TE approaches frequently to reroute traffic due to possible disruptions. It is even more difficult to adopt a green TE approach during peak hours for robustness, stability, and QoS constraints. Hence, we advocate evolutionary approach to target the off-peak or mid-peak traffic to avoid drastic disruptions. The approach can be applied to on-peak demands after careful evaluation.
- Evolving SDN and SR features can be leveraged: First, Prefix-SIDs, Adj-SID, and BSID are basic building blocks that facilitate flexible data forwarding and management of the SR policy. In addition, a *dynamic candidate path* is used

to express an optimization objective and constraints. Conceptually, dealing with links for energy efficiency is related to dealing with Adj-SIDs and BSIDs to forward traffic rather than simply the *SP*. The dynamic candidate path is also suitable for expressing energy objective.

The second important SR-specific feature is backup path for failure protection. SR architecture supports FRR without the overhead of RSVP-based FRR [18]. Two SR-based techniques for FRR are Loop-Free Alternate (LFA) and Topology-Independent Loop-Free Alternate (TI-LFA). LFA is simple and mostly available, but TI-LFA improves LFA to have 100% coverage and optimality of the backup path [17]. However, TI-LFA may not be available for each destination node if some links or nodes are powered off.

- Different QoS levels are not explicitly considered.

A common objective for many TE approaches is to minimize MLU or *min(MLU)* [15]. The objective of *min(MLU)* is to accommodate traffic bursts or to support switchover for failures. From the energy efficiency perspective, *min(MLU)* can result in more energy waste, especially for light traffic demands, as more or many links could have very low link utilization.

There are three main paradigms using explicit routing or tunneling techniques for green TE. The first one makes use of an OSPF extension to carry energy consumption parameter of each node in the network [9]. The information could then be used in RSVP-TE path calculation, i.e., the links with smaller energy consumption are favored. The approach, however, requires an extension to both OSPF and RSVP-TE.

The second paradigm considers link or node utilizations. Intuitively, if the network traffic demands are low and the utilization is below a threshold, the flows traversing through a particular link can be rerouted to other links, so that the target link or port can be powered off [1] [2] [3] [10] [11] [13]. Most green TE approaches for IP, MPLS, and SR, center around the idea, though variations exist. If all the links of a node have low utilizations, the node could potentially be powered off. The complexity and impact of switching off a node increases or may not be feasible, e.g., for critical elements or topology reason.

The third paradigm is SR specific. SR data packets can carry a high number of segments, which results in higher energy consumption for packet processing and transmissions. The number of segments can be reduced for some paths, especially after switching on-off of devices. In addition to segment reduction, another approach is segment consolidation based on periodic optimization. Note that the techniques used for segment reduction and/or consolidation can be used together with the techniques described in the second category.

The following discusses more on last two paradigms.

A. *Green TE with Link/Node Manipulation*

To support green TE that reroutes traffic from links with low utilization to other links, this paper focuses on SR for flexible dynamic or explicit routing.

To realize *min(MLU),* a function to select a *widest shortest path (WSP) first* has been used. If multiple *SP*s exist, the one with the highest amount of available bandwidth is first selected to keep link utilization lower. In contrast, to support green TE, *min(MLU)* can be replaced with *max(MLU)*, i.e., maximize MLU to a configured level. The objective of *max(MLU)* is to select links with higher utilization to avoid some other links with low link utilization so that those links with low utilization may be put into the energy saving mode. Evidently, there is a need for an upper threshold for *max(MLU),* e.g., 60%, to accommodate traffic fluctuations. The upper or maximum utilization threshold for a link should be determined based on specific target network and demands, and can be configurable.

Therefore, an energy-aware TE model can adapt *WSP* that is often used in traditional TE to the *least shortest path (LSP) first* algorithm, i.e., the *SP* that has the least amount of available bandwidth and also meets the upper threshold is preferred in path calculation. Further, if energy efficiency has higher precedence than delay, the *shortest least path (SLP) first* algorithm could be further evaluated to identify paths that have the least available bandwidth first (above a pre-defined threshold), and then to choose the shortest one among those *least paths*. The idea is to reroute more traffic to those links, since they will have less remaining bandwidth and are less likely to be put into the saving mode. *SLP* could be applied with a delay or hop count constraint to avoid excessive long paths.

SDN Controller can periodically collect the information from each node and check if path re-calculation is needed. For energy-aware TE, a crucial piece of information is the bandwidth usage or utilization of each link. If a new path is calculated, SDN Controller can send new explicit routes to each of the affected ingress nodes. Ingress can then use SR to forward packets by pushing labels or segments to the stack.

For a node or an SDN switch utilization, in this paper, it is defined as the average utilization of all links going out from the node. To power off a node, all links on this node need to be switched off first.

For robustness, we assume that SR features for backup protection is included, e.g., either LFA or TI-LFA. With those features, sub-50 *msec* link or node protection can be provided [17], including changes needed for segments if a link is put into sleep, as it is equivalent to a link failure. However, the protection coverage could be reduced over time, as network elements are powered off for energy savings. This is a trade-off that needs to be factored in for the mechanism.

B. *Green TE with Segment Management*

Two SR features may incur negative impact on energy efficiency: ECMP and large segment stack. SR often uses ECMP to support load sharing and resilience. However, when the network demands are low, some links have low utilizations which are good candidates for energy saving. Using ECMP in such a scenario can reduce the chance to switch off lightly utilized devices. Hence, ECMP is not recommended.

SR supports explicit routing without the signalling protocol. However, a caveat of this source routing-based feature is the possibility of large number of segments carried in data packets. Fig. 1 illustrates an example of numerous segments for data packets sent from node A to node Z via an explicit path.

Fig. 1 shows an explicit path (A, B, C, O, P, D, Z) for traffic flow from A to Z. Each number on a link represents a SID. The segment stack for each packet sent from node A is shown on the right, i.e., 6 128-bit segments in each packet from node A, 5 segments from node B, and etc. This is the worst-case scenario for the number of segments used for this simple example. Larger segment size requires higher energy consumption for transmissions and processing, as the number of packets is huge.

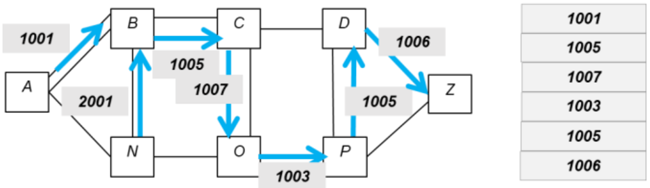

Fig. 1. SR Segment Stack: an Illustration

The number of segments is minimal if the *SP* and Prefix-SID are used, which is the default path calculation for SR. In this example, the minimal number of segments is only one if Prefix-SID 1006 is used together with the *SP*. In the event that the *SP* does not meet all constraints, e.g., bandwidth or link utilization for energy consideration, another path may be selected, as shown in Fig. 1. *Segment reduction* could be adopted to reduce the stack size in this case. For instance, the *SP* from A to C is (A, B, C), which is part of the explicit path. In this case, the first SID 1001 can be removed, as SID 1005 on top of the stack indicates the *SP* to node C, which is supported by SR already. Similarly, from C to P via O is also the *SP*; hence, only SID 1003 is enough and SID 1007 is not needed. Segment reduction can be evaluated after a non-*SP* is selected or periodically.

BSID is another mechanism that can reduce segments imposed by the source, and provide service independence [21]. For instance, if segment 1005 in Fig. 1 is configured as a BSID at node C, packets from node A destined to node Z only need to use the top two segments, i.e., <1001, 1005>. Next, node B pops <1001>, and node C removes BSID 1005 and binds or pushes four segments <1007, 1003, 1005, 1006> and forwards them to node O. Similarly, if node N sends packets to node Z following the same path and policy, node N only needs to push two segments <2001, 1005> to get node B which applies the same operation. BSID is primarily used to steer packets to a particular SR policy, e.g., low delay or energy-saving path, but it also can reduce the number of segments imposed by the source.

Another technique that can mitigate the segment processing overhead is called *segment consolidation*. Consider the example in Fig. 1. The minimal number of segments from A to Z is only 1, Prefix-SID 1006. The current explicit path has 6 segments from the ingress, which is much longer than the optimal value. The concept of *segment consolidation* can be borrowed from MPLS-TE tunnel reoptimization mechanisms in MPLS-TE: periodic, manual, event-driven, and lockdown [18]. Using those techniques, a number of segments that is much longer than that of the *SP* or a threshold can be flagged for reoptimization or consolidation, given the path's configured constraints.

In Fig. 1, assuming that reoptimization is triggered and finds path (A, N, O, P, D, Z) later that has a maximum of five local segments for traffic from A to Z, and also meets constraints. (Note that the actual number of segments needed for the new path could be as low as two or even one.) The flows from the original path can then be rerouted via the new path to reduce the segment processing/transmission overhead and shorten delay. Energy consumption could be reduced as a result. Note that rerouting in this case is triggered by pre-configured segment size, not link or node utilization.

IV. GREEN SR-TE ALGORITHMS

Based on the aforementioned assumptions and model, the following presents algorithms for a request of bandwidth *bw* from source node *s* to destination node *d*, with a hop count constraint *c* (optional). Bandwidth requests could be initiated by a node (an SDN switch) asking more bandwidth (auto-bandwidth) or rerouting flows to put a port into sleep.

The proposed green TE algorithm for path calculation (conducted at PCE) is presented in Algorithm 1, adapted from [16] for time-dependent TE. Some key parameters and symbols are defined in TABLE I. The first three parameters are desirable, not mandatory, based on an assumption that traffic patterns are known in advance, which can reduce the number of traffic rearrangement or disruption [16]. A reduced network topology could be used if desirable, but not required.

TABLE I. LIST OF NOTATIONS (ADAPTED FROM [7])

Parameter or Symbol	**Description**
expectedBw	expected bandwidth based on traffic patterns
plannedFreeBw	provisioned bandwidth
residueFreeBw	remaining bandwidth of the previous two parameters
$\mathbf{S_{uT\text{-}H}}$	Switch utilization threshold high
$\mathbf{S_{uT\text{-}Max}}$	Switch utilization threshold maximum
$\mathbf{S_{uT\text{-}Min}}$	Switch utilization threshold minimum
$\mathbf{S_{uT\text{-}L}}$	Switch utilization threshold low
$\mathbf{T_{das}}$	Time duration to activate a switch
$\mathbf{T_{dds}}$	Time duration to deactivate a switch
$\mathbf{T_{dal}}$	Time duration to activate a link
$\mathbf{T_{ddl}}$	Time duration to deactivate a link

Algorithm 1: Path calculation for rerouting flows from s to d

```
// Initialize expectedBw, plannedFreeBw, residueFreeBw, $S_{uT-Max}$
if bw < expectedBw (s,d)
    do LSP (s, d, bw, c, $S_{uT-Max}$) on plannedTree
    if success
        expectedBw (s,d) = expectedBw (s,d) - bw
        plannedFreeBw (s,d) = plannedFreeBw (s,d)- bw
    else
        do LSP (s, d, bw, c, $S_{uT-Max}$) on residueNet
        if success
            residueFreeBw (s,d) = residueFreeBw (s,d)- bw
        end if
    end if-else
else
    do LSP (s, d, bw, c, $S_{uT-Max}$) on residueNet
    if success
        residueFreeBw (s, d) = residueFreeBw (s, d) - bw
    end if
end if-else
if success
    inform ingress switch s the new path related information
else
    reject the request
end if-else
```

In Algorithm 1, *LSP* is used to find the *SP* that has the least amount of available bandwidth under the maximum threshold S_{uT-Max}. The source node then uses SR to forward the aggregated flows to the destination. SR mostly exploits *SPs*, though explicit routing over longer paths can be used returned by PCE. Hence, *LSP* is consistent with *SP* and is adopted in Algorithm 1.

Further, PCE has been used for SDN for path calculation, including *constraint-based SP first* (*CSPF*) algorithm using available bandwidth or link utilization information. LFA or TI-LFA can also be calculated using SR or the proposed *LSP* algorithm for FRR (sub-50 *msec*) purpose. Note that there is a trade-off between availability and energy efficiency. More network elements are turned off means more energy savings, but robustness may suffer due to reduced topology. Hence, TI-LAF may not be available. The complexity of *LSP* is equivalent to that of existing *SPF* or *CSPF* used in practice.

Take Fig. 2 as an example. Assuming all links have the same cost and capacity of 100 Mbps, and the configured maximum link utilization threshold is 60%. The number on each link is the available bandwidth for the link. There are two *SP*s from B to D, i.e., (B, A, D) and (B, C, D). Path (B, C, D) has lower available bandwidth 70 Mbps, or 30% link utilization, hence (B, C, D) will be favored to stay active using the *LSP* algorithm. Because path (B, A, D) has lower link utilization, only 20%, hence it has a higher chance to be deactivated later for energy efficiency (see Algorithm 2, adapted from [7]).

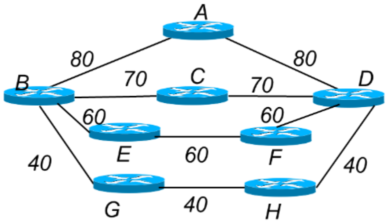

Fig. 2. A Simple Topology: an illustration for *LSP* and *SLP*

On the other hand, path (B, E, F, D) and (B, G, H, D) have higher link utilization at 40% (60 Mbps available) and 60% (40 Mbps available), respectively. The links along these two paths will be less likely to be selected for deactivation or energy saving. But both paths are not *SP*s.

There is a trade-off between energy saving and delay in a scenario like this. In this case, *LSP* could be replaced with the *SLP* algorithm in favor of potential energy savings. Some links that have low utilization may not identified by the *LSP* path calculation because they are not on the *SPs*. To select those links that have higher chance for the energy saving mode, links with higher utilization should be first considered for rerouting additional flows or staying active (assuming they are under the maximum upper threshold, e.g., 60% for this example), though the delay may be higher. In this example, (B, E, F, D) or (B, G, H, D) could be first selected for green rerouting purpose, provided that adding the new traffic will not exceed a threshold. (B, A, D) or (B, C, D) has lower utilization and higher chance to be deactivated, but two longer paths (B, E, F, D) and (B, G, H, D) are not selected by *LSP*, as they are not *SP*s.

To support the green TE objective, the *SLP* algorithm can be adopted instead of *LSP*. For this example, (B, G, H, D) has the least amount of available bandwidth, but it will not be selected, since its utilization is already 60%, i.e., available bandwidth is only 40 Mbps out of 100 Mbps. As a result, *SLP* will select the path wit the next least amount of bandwidth, i.e., (B, E, F, D), for additional traffic rerouting. In Algorithm 1, lines 2, 7, and 13 *LSP* could be replaced with *SLP* if energy consumption has a higher precedence than delay. Note that *SLP* does not find *SP*s first, which is inconsistent with SR native routing calculation. Hence *SLP* needs more modifications. More evaluation needs to be conducted for energy efficiency.

SR often enables ECMP for efficiency and resiliency purposes. However, ECMP is not included for Algorithm 1 for better control of the paths and links, i.e., it would have higher chance to put some links into sleep without ECMP.

Monitoring link utilization is essential for energy-aware TE. When a link utilization is below a low threshold (S_{uT-L}) for a certain period T_{ddl} or the minimum threshold (S_{uT-Min}), the node or SDN switch will notify the SDN Controller to invoke the path calculation algorithm via PCE.

Algorithm 2: Deactivating a link of an SDN switch

```
monitor each link utilization of a switch
for each link, if the link utilization of a link is < $S_{uT-L}$
    activate the timer T
    while T < $T_{ddl}$
        if link utilization > $S_{uT-L}$
            deactivate timer T   // and exit
        else
            invoke Algorithm 1 to reroute flow(s) to other path(s)
            deactivate the link
        end if
    end while
end for
```

Activating a link algorithm is depicted in Algorithm 3. Line 8 selects a link from the inactive list of a switch or when a switch is powered back on again.

Algorithm 3: Activating a link of an SDN switch

```
1.  monitor utilization for each active link,
    return if all links for a switch are active
2.  for each active switch, if utilization is > $S_{uT-H}$ & has inactive link(s)
3.      activate the timer T
4.      while T < $T_{dal}$
5.          if link utilization < $S_{uT-H}$
6.              deactivate timer T   // and exit
7.          else
8.              select and activate a link from inactive list
9.              add the link to the active list & recalculate path(s) if
                needed
10.         end if
11.     end while
12. end for
```

Algorithms 2 and 3 deal with links of a switch. If the SDN switch utilization is below a threshold for a certain period of time, the switch could be put to sleep. Algorithm 4 depicts the steps that is running at SDN/PCE for candidate switches.

Algorithm 4: Deactivating an SDN switch

```
monitor the utilization of all active switches
```

```
sort the switches based on utilization in increasing order
for each active switch  // start from the least utilization
    if the switch utilization is < $S_{uT\text{-}L}$
        activate the timer T
        while T > $T_{dds}$
            if target switch utilization > $S_{uT\text{-}L}$
                deactivate timer T   // and exit
            else
                invoke Algorithm 1 for each link for rerouting
                If all links can be deactivated,
                    deactivate the selected target switch
            end if-else
        end while
        update the forwarding table
    end If
end for
```

To activate a node when traffic demands increase and if a node has higher utilization than the high threshold and also has adjacent links that are inactive, the inactive node becomes a candidate. If high utilization persists, an inactive node that is adjacent to the target node will be activated along with the link between these two nodes. Some paths or segments may need to be adjusted. Other links could be powered on when needed.

Algorithm 5: Activating an SDN switch

```
monitor the utilization of all active switches
for each switch, if its utilization > $S_{uT\text{-}H}$ and it has inactive link(s)
    activate the timer T
    while T > $T_{das}$
        if switch utilization < $S_{uT\text{-}H}$
            deactivate timer T   // and exit
        else
            invoke Algorithm 3 to activate an adjacent switch
                and the link connected to the current switch
            add the switch/link to active list and compute paths
        end if-else
    end while
end for
```

Discussions. Using the SDN/PCE and SR features, computational complexity is no more than that of existing *SPF* or *CSPF* algorithms, perhaps even with fewer links or nodes. Similarly, path calculation for FRR is also supported with the SR techniques. Extra time needed for segment reduction or consolidation is also bounded by *SPF* or *CSPF* which is done at PCE. With segment manipulation, segment processing and transmission costs can be reduced for each packet at each hop, which happens much more frequently than path calculations. The proposed method also identifies candidate links earlier to put into sleep, as *max(MLU)* is used and ECMP is disabled.

V. CONCLUSIONS AND FUTURE RESEARCH

Energy consumption for networks is increasing rapidly, which raises concerns for global environment. Consequently, energy-aware TE becomes a vital topic. Various approaches have been proposed, but they incur extra complexity. This paper instead exploited existing and evolving SR and SDN/PCE features that can facilitate green TE. Green TE methods should be built on those valuable SR and SDN/PCE techniques and be applied seamlessly with them, instead of being developed separately. Other novel SR features could be further considered for devising green TE techniques. For instance: *BSID* can be used more to effectively manage segments in parts of the network for stable green TE policy. The *color* attribute is used for SR policy identification. *Color*, a number, can be considered for energy efficiency *intent* which is then expressed using the *dynamic candidate path* concept for an optimization objective, i.e., energy efficiency, and a set of constraints.

Experimental evaluation is needed to validate the proposed approach and evolving features. Trade-off analysis between energy efficiency and requirements, e.g., QoS, resiliency, and scalability, also needs to be evaluated for various use cases.

REFERENCES

[1] R. Maaloul, L. Chaari, B. Cousin, "Energy saving in carrier-grade networks: A survey," *Computer Standards & Interfaces*, Jan. 2018.

[2] Carpa R, Olivier G, and Lefevre L, "Segment routing based traffic engineering for energy efficient backbone networks," *Proc. of IEEE Int'l Conf. on Advanced Net. and Telecommuni. Systems*, Dec. 2014.

[3] O. Okonor, N. Wang, S. Georgoulas, and Z. Sun, "Dynamic link sleeping reconfigurations for green traffic engineering," *Int'l Journal of Communication Systems*, vol. 30, no. 9, p. e3224, Jun. 2017.

[4] J. A. Manjate, M. Hidell, and P. Sjodin, "Can energy-aware routing Improve the Energy Savings of Energy-Efficient Ethernet?," *IEEE Trans. on Green Communi. and Networking*, pp. 787–794, Sep. 2018.

[5] C. Thaenchaikun, et al., "Mitigate the load sharing of segment routing for SDN green traffic engineering," *Proc. of Int'l Symp. on Intelligent Signal Processing and Communication Systems*, Oct. 2016.

[6] X. Li, C.-H. Lung, S. Majumdar "Green spine switch management fordatacenter networks," *Journal of Cloud Computing: Advances, Systems and Applications*, vol. 5, no. 9, pp. 1–19, 2016.

[7] O. Osamudiamen, C.-H. Lung, "Segment routing green spine switch management systems for data center networks," *Proc. of IEEE Int'l Conf. on Dependable and Secure Computing*, pp. 1–8, Dec. 2018.

[8] K. S. Ghuman, A. Nayak, "Per-packet based energy aware segment routing approach for Data Center Networks with SDN," *Proc. of the 24th International Conference on Telecommunications*, Jul. 2017.

[9] Z. L. Gang Yan, Jianjun Yang, "OSPF extensions for MPLS green traffic engineering," *IETF draft-li-ospf-ext-green-te-01*, 2014.

[10] M. Zhang, C. Yi, B. Liu, B. Zhang, "GreenTE: power-aware traffic engineering," *Proc. of IEEE Int'l Conf. on Network Protocols*, 2010.

[11] R. Carpa, *Energy Efficient Traffic Engineering in Software Defined Networks*, Thèse de Doctorat, Université de Lyon 2017.

[12] A. B. Vieira, et al., "An SDN-based energy-aware traffic management mechanism," *Annals of Telecommunications*, Jun. 2021.

[13] X. Jia, et al., "Intelligent path control for energy-saving in hybrid SDN networks," *Computer Networks*, pp. 65–76, Feb. 2018.

[14] D. Awduche, et al., "RSVP-TE: Extensions to RSVP for LSP Tunnels," *IETF RFC 3209*, 2001.

[15] J. R. and M. T. B. Fortz, "Traffic engineering with traditional IP routing protocols," *IEEE Communi. Magazine*, 40(10), pp. 118–124, 2002.

[16] Y. Yang, C.-H. Lung, "The role of traffic forecasting in QoS routing - a case study of time-dependent routing," *Proc. of IEEE ICC*, May 2005.

[17] C. Filsfils, et al., *Segment Routing Part I*, 2017.

[18] E. Osborne and A. Simha, *Traffic engineering with MPLS*. Cisco, 2002.

[19] Z. Li, D. Dhody, Q. Zhao, K. Ke, and B. Khasanov, "The use cases for path computation element (PCE) as a central controller (PCECC).," *Internet-Draft:draft-ietf-teas-pcecc-use-cases-09*, 2022.

[20] P. L. Ventre *et al.*, "Segment routing: A comprehensive survey of research rctivities, standardization efforts, and implementation results," *IEEE Commu. Surveys and Tutorials*, pp. 182–221, Jan. 2021.

[21] C. Filsfils, et al., *Segment Routing Part II*, 2019.