A COMPREHENSIVE SURVEY OF ENERGY-EFFICIENCY APPROACHES IN WIRED NETWORKS

Rahil Gandotra[1] and Levi Perigo[2]

[1]Interdisciplinary Telecom Program, University of Colorado Boulder, USA
[2]Department of Computer Science, University of Colorado Boulder, USA

ABSTRACT

Energy consumption by the network infrastructure is growing expeditiously with the rise of the Internet. Critical research efforts have been pursued by academia, industry and governments to make networks, such as the Internet, operate more energy efficiently and reduce their power consumption. This work presents an in-depth survey of the approaches to reduce energy consumption in wired networks by first categorizing existing research into broad categories and then presenting the specific techniques, research challenges, and important conclusions. At abroad level, we present five categories of approaches for energy efficiency in wired networks – (i) sleeping of network elements, (ii) link rate adaptation, (iii) proxying, (iv) store and forward, and (v) network traffic aggregation. Additionally, this survey reviews work in energy modeling and measurement, energy-related standards and metrics, and enumerates discussion points for future work and motivations.

KEYWORDS

Energy efficiency, energy proportionality, energy-aware protocols, wired networks.

1. INTRODUCTION

The Internet has proven to be one of the most important technological innovations. It acts as the primary catalyst in the global digital revolution, and is considered a public utility, along with running water and electricity. Fig. 1 compares access to running water to the Internet in the USA. Although this is not a like-for-like comparison, it is still important to note the pace of Internet proliferation when compared to other public utilities. Recent statistics estimate the global Internet adoption rate to be around 65.6%, or 5.1 billion users [1], and these numbers are increasing rapidly.

In addition to the escalation of Internet adoption, the digitization of services including over the top (OTT) video streaming, ecommerce, voice over IP (VoIP), and the Internet of things (IoT), has established pressure on service and network providers to expand their network hardware infrastructure. This network hardware expansion results in an upward trend of the energy consumed by the Internet globally. While Koomey’s law states that post-2000 the energy efficiency of computing hardware has doubled every 2.6 years, it is still slower than the increase in data traffic, which follows Moore’s law, doubling every 18 months [2]. Although the energy consumption by the wired-networking infrastructure is a small fraction of the total consumption by the information and communications technologies, the absolute numbers indicate that efforts to reduce energy consumption in computer networks are warranted [3].

David C. Wyld et al. (Eds): MLDS, NECO, SEMIT, IBCOM, SPPR, SCAI, CSIA, ICCSEA - 2021
pp. 261-282, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.111822

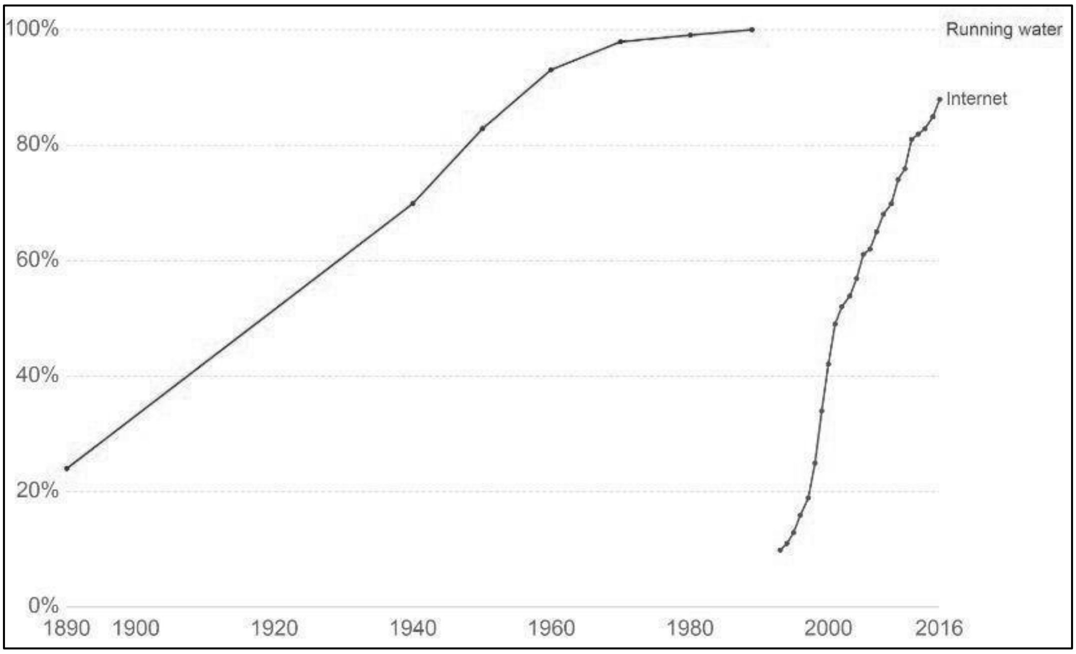

Figure 1. Access to running water vs the Internet in the USA

In the past two decades, there has been a surge of research dedicated to make the network infrastructure more energy efficient. The three major areas of focus have been, (i) system-oriented energy management, (ii) energy-aware network design, and (iii) energy-aware protocol design. System-oriented approaches strive to make the underlying hardware more energy-efficient using techniques like frequency scaling, dynamic voltage scaling (DVS), providing additional sleep states (like C-states in Intel) and performance states (like P-states in Intel). Energy-aware network design approaches seek to find the optimal network topologies that are most suited for energy savings while maintaining performance and reliability standards. Energy-aware protocol design approaches attempt to rethink and remodel existing protocols, and propose new protocols, to incorporate energy-awareness with the goal to provide opportunities to reduce network energy consumption. The system-oriented approaches are comparatively more mature than the other two approaches. This survey paper, therefore, excludes system-oriented approaches and focuses on network design and protocol-oriented approaches for wired networks. We study the work done in the last twenty years, while analyzing and discussing the research challenges, with the aim to present the state-of-the-art and realize important conclusions that can benefit future work in this field.

The remainder of the paper is organized as follows: Section II discusses the broad categories of the existing work, Section III presents a detailed survey of the specific approaches in each category, Section IV presents work that is closely related, but are not included in Section III, and Section V presents conclusions and discussion points for future research.

2. CATEGORIZATION OF EXISTING WORK

Computer networks are a critical part of the Internet infrastructure acting as highways connecting users to online services. They are conventionally designed to handle peak-traffic loads with sufficient redundancy and Quality-of-Service (QoS). Owing to the best-effort nature of the Internet Protocol (IP), network architectures and applications are developed to provide reliable transmission in case of failures. While such techniques have promoted the growth of the Internet, they are not typically energy-proportional, i.e. their energy consumption is not proportional to the traffic load. The practice of over-provisioning and observations of under-utilization and non-energy-proportional behavior during the majority of times prompted researchers to find methods to reduce the energy-consumption of wired networks. Since individual network hardware devices and their components have become increasingly reliable, and network protocols have become more robust, new techniques can be leveraged to determine better energy savings.

Different parts of the wired-network infrastructure – network topology, hardware devices, electrical/optical links, protocols, and network management techniques – exhibit different runtime behaviors and offer different opportunities to lower the energy consumption. We categorize the existing work into five distinct categories: **sleeping of network elements**, **link rate adaptation**, **proxying**, **store and forward**, and **network traffic aggregation**, and provide an overview of them in this section. Each category targets a specific area of wired-networking with its own research challenges, implementation-specific techniques, and scope for future work. Fig. 2 shows the taxonomy of existing research in different categories.The detailed survey of specific approaches in each category is presented in section III.

The concept of **sleeping of network elements** is based on the intuition that because computer networks are provisioned for peak load and are generally under-utilized, parts of the network or individual devices can be put to sleep during off-peak hours. This category of work was one of the earliest approaches (2003) to save network energy consumption and has seen a multitude of proposals. The work involved in this approach is two-fold: identifying/predicting low-utilization periods suitable for energy savings, and reconfiguring the network to put network elements to sleep. The target network element to be put to sleep could either be the physical device(s), or individual components of one device. Although networking devices traditionally provide limited support for sleeping, approaches in this category propose techniques which either make assumptions about the underlying hardware support, or find ways to work around this limitation. Waking up a sleeping element and understanding the impact of sleeping on network protocols were some of the research challenges in this category.

Link rate adaption gained popularity as a potential approach after research proved that high link rates (1/10 Gbps) consume more energy as compared to low link rates (10/100 Mbps), and that a desirable level of performance could be achieved by running links at lower speeds. The work involved in this approach is two-fold: identifying/predicting low-utilization periods wherein the links could be operated at slower rates, and reconfiguring the network to configure specific links at lower speeds. Both reactive approaches, based on matching the current utilization with link speeds, and proactive approaches, predicting current/future traffic patterns from historical utilization data, were proposed. Research challenges in this category include understanding the tradeoffs between frequent link-rate change and performance, and the impact of these changes on protocols, especially cost-based routing and switching protocols.

Proxying is based on the hypothesis that the periodic network protocol traffic does not allow end-hosts to sleep effectively. Although this approach does not directly target energy savings from the network, it involves the network elements to act as proxies to enable infrastructure-level savings. The techniques proposed in this category attempted to counter the requirements necessitating networked end-hosts to be connected and responsive even while sleeping, i.e. always-on should not mean always powered on. A survey of office and commercial equipment indicated that most of the computers were not put into sleep modes due to the above requirement of always maintaining network presence [27], and this prompted researchers to consider network proxying and find ways to offload some of the end-host responsibilities to a proxy. The placement of the proxy in the network and the tradeoff between energy savings by end-hosts sleeping and the additional energy consumption by the proxy were some of the research challenges in this area of work.

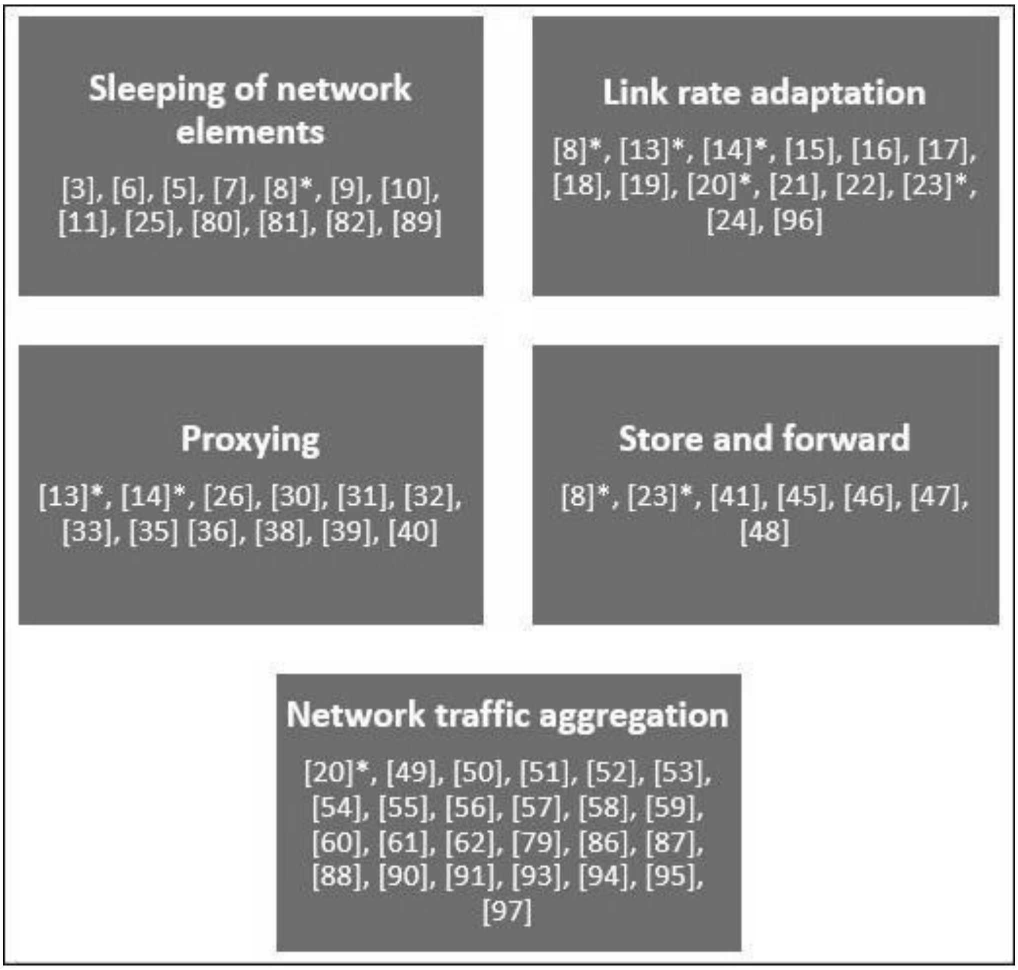

Figure 2. Categorization of existing work (* indicates work appears in multiple categories)

The observation that many web applications exhibit bursty traffic with high inter-arrival traffic times led to the development of **store and forward** techniques. Although this is not a mature area and does not have many research proposals, it is still an important area of work to study with insights for future works. While the traditional practice in network engineering is to avoid bursts using congestion avoidance and QoS queues, the insight behind the techniques in this area is to shape traffic in bursts in order to allow network elements to be put in low power/sleep mode in between these bursts. The difference between sleeping of network elements and store and forward is that techniques in this area use explicit proactive mechanisms to shape and engineer traffic with the intention to create sufficient bunching that would allow efficient sleeping, while techniques in the former aim to reactively put network elements to sleep based on utilizations. Store and forward techniques could be considered as a subset of the sleeping of network elements category, but we believe there are sufficient design differences in the two areas to warrant separate categories. Developing practical ways to implement this approach, synchronizing the bursts across the network to enable efficient sleeping, and understanding the tradeoffs between performance and energy savings were some of the research challenges in this area.

Network traffic aggregation techniques are based on the assumption that networks are over-provisioned and under-utilized, and that a subset of devices and links would suffice during off-peak hours. While we do not include server load aggregation techniques – clustering compute resources onto minimal physical servers to save energy – in this area, specific techniques that involve manipulating network traffic to enable server-load aggregation are included. The difference between network aggregation techniques and sleeping of network elements is that approaches in this area attempt to proactively find minimum-power subsets of the infrastructure that can provide the desired level of performance while enabling energy savings, whereas, techniques on sleeping of network elements attempt to put network elements to sleep reactively while considering the entire topology. These techniques could be included in the first category, but we believe that the insight and the way forward is different enough to justify separate categories. Research challenges in this area focus on traffic engineering problems such as NP-complete [37] and finding methods to reduce the computing time that allows for on-the-fly solutions to efficiently save energy.

3. DETAILED SURVEY

In this section we present a chronological survey of specific approaches in each category.

3.1. Sleeping of network elements

Gupta and Singh [3] were among the earliest researchers to examine the energy consumption of networking devices. A previous study which analyzed the energy consumption of office and telecommunications equipment in commercial buildings revealed that Internet devices consumed 6.05 TW-h of energy in the USA [4]. This realization prompted the researchers to find ways and suggest directions to save energy going forward. They proposed sleeping of devices/interfaces as a solution and explored coordinated and uncoordinated sleeping. In coordinated sleeping, the network devices collectively made decisions on which elements to put to sleep, while in uncoordinated sleeping, devices made such decisions in isolation, independent of others. Analyzing sample traces from their autonomous system (AS), they showed that sleeping was a reasonable solution, however, modifications to existing protocols and the Internet architecture would be needed to maximize the amount of energy conserved.

Soteriou and Peh [6] earmarked links interconnecting routers as a major source of energy consumption and proposed a dynamic power management policy employing on/off links. The work involved deriving a power-performance connectivity graph to identify candidate on/off links, developing a deadlock-free routing algorithm based on the information from the graph, and implementing an on/off decision mechanism. Input buffer utilization of a router was used to determine if a link could be turned off. Their results indicated a 37.5% reduction in energy consumption for an 8-ary 2-mesh topology with a moderate latency increase.

Gupta et al. extended their prior work in [5] to investigate the feasibility of sleeping in LAN devices. LAN switches are targeted as they comprise the bulk of LAN devices and therefore consume the largest amount of energy. Three different sleep models were proposed – (i) Simple sleep – wherein timers are employed to wake up a sleeping device/interface and all incoming packets during sleep time are lost, (ii) Hardware assisted sleep (HAS) – wherein the incoming packet is lost but it wakes up a sleeping device/interface, and (iii) Hardware assisted buffered sleep (HABS) – wherein the incoming packet is buffered and it wakes up a sleeping device/interface. Both HAS and HABS assumed hardware support which was not available at the time. A need to modify existing protocols (like STP and TCP) to allow for efficient sleeping was reiterated.

While the work in [5] was a preliminary study to evaluate the viability of sleeping as a solution to save energy, in [7] Gupta et al. presented practical approaches to achieve it, using technology available in the hardware of the time. The target area was Ethernet interfaces and their proposal leveraged smart timer-based Ethernet transceivers that automatically turned off when no power was detected on the other end. The algorithms to decide when to turn off a link included – (i) On/Off-1 – wherein the upstream interface of a link determines if the link can be put to sleep by estimating the number of packet arrivals in a time period t, and finding the maximum value of t for which the probability of packet arrivals being less than a buffer threshold is less than 10%, and (ii) On/Off-2 – which is a modification of the previous algorithm, wherein instead of both the upstream and downstream interfaces waking up after time period t and reevaluating if they can go back to sleep, only the upstream interface wakes up and runs the algorithm, to allow for higher energy savings. The simulation results on real-world traces indicated that 37% of the interfaces could be put to sleep anywhere from 40-98% of the time. Another conclusion from their work was that sleeping interfaces allowed portions of the internal switching fabric to be put to sleep as well.

Nedevschi et al. [8] studied opportunistic sleeping in which link interfaces sleep when no packet arrivals are observed for a period of time. This approach assumed hardware support of wake-on-arrival – the circuitry that sensed traffic on the interface and has the capability to wake up the device. The simulation results on Abilene showed that opportunistic sleeping was suitable for LANs with high idle times, however, it was not suitable for high-speed links with low idle times. The percentages of time links could sleep with either constant bit rate traffic or bursty traffic were minimal, proving that this approach was suitable only in the specific LAN scenarios.

Following similar intuitions as before, Ananthanarayanan and Katz [9] proposed two energy-saving schemes for switches and studied the tradeoffs between performance and energy consumption. Their schemes fall under the umbrella of uncoordinated sleeping in the way that they are run on each switch independently. The concept of shadow port is introduced wherein a shadow port is associated with a cluster of normal ports allowing packets to be buffered at the shadow port in case the associated normal ports are sleeping. The Time Window Prediction (TWP) scheme is similar to the On/Off-1 algorithm in [7], wherein predictions are made about the number of packets traversing a port in a time window, and if this number is below a set threshold then the port is powered off. However, the difference in this scheme was that the sleep window is made adaptive by setting a bound on the increase in latency – if the latency of the buffered packets increases above a set bound, the sleep window is reactively reduced. The Power Save Mode (PSM) scheme was a modification to TWP to make it more aggressive for energy-savings by not considering traffic flows while making sleep decisions. This enabled higher energy savings at the cost of increased latency. The simulation results employing an enterprise network's traffic patterns demonstrate a 18-21% potential for energy savings using the TWP scheme. Their tradeoff study indicated that the wake-on-arrival capability and large buffer sizes could allow for a significant increase in energy-savings while ensuring minimal performance degradation.

Chiaraviglio et al. [10] proposed an algorithm that creates an ordered list of nodes by decreasing power consumption, and for each element in the list, it first powers down the node and then checks if network-wide connectivity is maintained. If the network is disconnected, the node is powered back on and the algorithm proceeds to the next list element. Similar procedure is employed to power down links ensuring connectivity and maximum link load constraints are met. The simulation results on topologies similar to those of national ISPs confirmed the possibility to save more than 23% of energy.

The IEEE 802.3az – Energy Efficient Ethernet (EEE) standard was approved in September 2010 with the aim to reduce the energy consumption of computer networks [11]. While there was more than one scheme initially proposed for this standard, the approach of low-power idle, developed at Intel, was finally adopted [12]. Ethernet interfaces traditionally transmitted an auxiliary signal called IDLE when no data packets are transmitted to keep the transmitters and receivers synchronized. In 802.3az, in the case of no traffic, the link enters a low-power state and sends a Refresh signal over short durations allowing the link to consume less energy over large durations. Estimated savings by using EEE were projected to be $410 million/year in the USA, and over $1 billion/year globally.

Herrería et al. enhanced the techniques of opportunistic sleeping in [25] by proposing four interface states – active, idle, transition, and sleep. The interface transitioned to sleep as soon as its buffer was empty, to allow for larger sleep durations, and the wake up back to active state was triggered either by a timer or the buffer length crossing a certain threshold. The simulation results showed a potential to save 75% energy without noticeable impact to performance.

Distributed algorithms to make sleeping decisions are presented in [80-82]. In [80], the approach involved selecting a router as the power saving router (PSR) by random election, and the PSR checking if network connectivity would be maintained if itself is disconnected from the network for a certain time period. If connectivity could be maintained, the PSR recomputed the routing table which would then be broadcast through the network. The numerical results from this approach indicated that up to 18% of the power could be saved for the whole network. [81] utilized the periodic link state information (LSA) to decide which links to switch off in a distributed way. The proposed algorithm did not require the knowledge of the actual and past/future traffic matrices, and the results from realistic case studies indicated energy savings of up to 50%. In [82], Patota et al. proposed DAFNES – a distributed algorithm for network energy saving based on stress-centrality. The stress centrality of a network node n refers to the number of shortest paths between any two endpoints which pass through node n. The proposed approach falls under the category of coordinated sleeping wherein switches, one at a time, made the decision to power off linecards independently, but the selection of the switch required synchronization by the exchange of control messages. The algorithm involved the computing and exchange of the stress centrality values between all switches, and the switch with the lowest stress centrality switched off its linecards in each iteration. The testing results showed the potential to save up to 50% of network energy.

[89] aimed to reduce the power consumption in data center networks by optimizing the number of SDN controllers active to serve the OpenFlow switches. The proposed heuristic first sorted all switches by decreasing order of the number of aggregated flow arrival rates, and then assigned each switch to a partially filled controller with the smallest but sufficient capacity. If no active controller was found with sufficient capacity, then a new controller node would be turned on.

3.2. Link rate adaptation

The concept of link rate adaptation was first proposed by Christensen et al. [13] and was based on the realization that higher link speeds equated to higher energy consumption. The aim was to match link data rates with the traffic levels to make networks more energy proportional. In [14], Gunaratne et al. preliminarily explored rate adaptation for links between PCs and access-layer switches. The proposed algorithm made decisions to increase or decrease link rates based on the queue lengths on personal computer (PC)NICs and switch interfaces, and the simulation experiments on a university campus network confirmed that it was feasible to operate at lower data rates with no significant increase in delay.

The work to develop a buffer threshold policy was extended in [15], [16] and [17]. In [15], a Markov model is developed for a state-dependent service rate, single-server queue while making the assumptions of standard Poisson arrival rates and exponential service rates. The proposed model is augmented by ensuring that service rate transitions occur only at service completions to replicate traditional Ethernet behavior. Single threshold policy, wherein service rates are switched to higher or lower rates if the output buffer utilization crosses a set threshold, and dual threshold policy, wherein upper and lower thresholds are defined and the service rate is switched to a higher rate when the output buffer utilization cross the upper threshold, and switched to a lower rate when the output buffer utilization falls below the lower threshold, are presented. The simulation experiments performed on traces from two university campuses indicated that links could be run at lower rates 99% of the time with a 4 ms average increase in delay. To prevent frequent link rate changes adversely impacting performance in case of smooth and bursty traffic, current link utilization information is incorporated into the buffer threshold policy in [16]. An important insight in this work was that links should be operated at lower data rates for at most 50% utilization, which equates to 5% utilization at the higher rate, and thus, if the current link utilization exceeds 5% at the higher rate, the link is not switched to the lower rate in order to balance performance with energy savings. An additional policy – time-out threshold policy –was proposed in [17] with the aim to reduce complexity of a NIC having link rata adaptation capability. The notion behind this scheme was to hard-set the time a link would run at the higher data rate and once this timer expired, and if the buffer utilization is below the set threshold, the link would switch to the lower data rate, and if not, it would continue to operate at the high rate. Transition from low to high data rates were immediate, triggered by the buffer utilization crossing the set threshold. Callegari et al. extended this work in [24] by incorporating transition times into their Markov model for a dual threshold policy, but their simulation experiments concluded that link rate adaptation was unsuitable for the NIC buffer sizes of the time.

A mechanism to implement link rate adaptation was proposed in [18]. A MAC handshake protocol was presented wherein a MAC frame containing the desired line rate was exchanged between the two nodes comprising a link. A switch to a lower data rate initiated by one node was permitted only if the other node responded with an ACK; if the other node cannot switch to the lower data rate, it responded with a NACK, and the first node continued operation at its original rate. A switch to a higher data rate initiated by one node would always be responded with an ACK by the other node to ensure no performance degradation. This mechanism was formalized as Rapid PHY Selection (RPS) in [19] and was proposed as one of the candidate approaches for IEEE 802.3az. The emulation results indicated little to no impact, i.e. minimal increase in delay and no packet loss, on TCP and UDP file transfers using this scheme.

Nedevschi et al. studied rate adaptation in [8] with the motivation being that operating links at slower speeds have twofold benefits – operating at slower frequencies consumes less energy, and operating at slower frequencies allows the use of dynamic voltage scaling (DVS) – making energy consumption scale quadratically with operating frequency. A practical algorithm is proposed which uses exponentially weighted moving average (EWMA) to make estimations about packet arrival times and uses current link utilization and operating rate information to ensure a rate change does not violate delay constraints. Two important conclusions from the simulation experiments performed on traces from Abilene were - (i) the granularity and distribution of operating rates played an important factor in the number of transitions and therefore the amount of energy saved; uniformly distributed link rates performed significantly better than exponentially distributed rates available in most networking hardware, and (ii) the time to transition between data rates also impacts the amount of energy savings, with higher transition times leading to reduced savings and higher delay.

Mahadevan et al. [20] studied link state adaptation (LSA) and performed experiments by simulating a Web 2.0 workload in a production data center topology. The proposed strawman scheme reactively adapted link rates based on current utilizations, while the service level (SL) aware scheme added constraints to guarantee a minimum level of performance was maintained – ensuring the link utilization was below 70% at any given time. The schemes assumed the existence of an Oracle that had perfect knowledge about upcoming traffic. The results showed that 16% energy savings were possible employing the LSA scheme, while SL-aware LSA consumed slightly more energy but provided significantly better performance. The deployment considerations described two approaches – reactively making changes by collecting link utilization information using protocols such as SNMP, and proactively making changes by predicting future traffic patterns using simple models such as AR(1). However, the prediction-based approach led to over and under-estimations in their simulations.

Abts et al. [21] leveraged the capability of modern plesiochronous links to operate in a dynamic range to make data centers more energy proportional. The intuition is that high-speed communication channels generally comprise of multiple links operating plesiochronously independent of each other and of the core router logic rate, and this independence is exploited to match operating rates with estimated bandwidth requirements. The proposed mechanism involved the switch tracking the amount of traffic traversing through it in a given time period, and if it exceeded a threshold defined for each link, the operating rate was doubled, and if it was less than the threshold, the operating rate was halved. The simulation experiments performed on synthetic and real-world traces (Google's data center) and assuming perfectly energy-proportional channels indicated a potential to save 15-36% of energy depending on traffic patterns.

Ginis [22] described the various link rates and power states for ADSL2 and ADSL2+ with the aim to achieve energy savings when averaged over a long period of time. Staessens et al. [23] proposed leveraging the capabilities offered by software-defined networks (SDN) to perform link rate adaptation. Since an SDN controller has a global view of the network, collecting and analyzing traffic patterns is less complicated, and this allows for easier deployment of the developed link rate policies [62].[96] combined the strategies of sleeping of network elements and link rate adaptation into one scheme for SDN-based data centers. The proposed algorithm proactively checked for inactive links and switches to power them off, and also reduced the link speeds of ports with low utilization.

3.3. Proxying

The concept of proxying in Ethernet interfaces was introduced in [13]. The authors suggested adding some intelligence to the PC NICs allowing them to respond to non-critical packets and waking up the PC from sleep in case of packets requiring a response. [26] presented the initial design and development behind ideas for a proxying Ethernet adapter. A study conducted by the Lawrence Berkeley National Laboratory in 2004 revealed that existing energy management features such as sleep states were disabled in around 95% of all PCs [27]. The major reasons behind this included certain applications not allowing the PC to sleep due to open sessions by exchanging keepalive packets, or network administrators manually disabling the features to ensure seamless firmware upgrades. Industry standards for networked energy management such as Wake-on-LAN (WOL), which used a Magic Packet (MAC address of the receiving NIC 16 times) to wake up a sleeping PC [28], and Advanced Configuration and Power Interface (ACPI), which specified standard interfaces for communication between applications and the BIOS, and extended WOL by providing direct wake-up using IP or ARP packets [29], were proposed, but rarely used for the purpose of energy savings. The authors in [26] suggested a control logic, wherein incoming packets either were ignored, handled at the proxy, or triggered the PC wake-up. The energy savings using this approach were estimated to be around 1 TW-h/year, implying an $80 million savings assuming 8 cents/kWh.

Gunaratne et al. [14] proposed two approaches in this area – protocol proxying and split TCP connections. Protocol proxying involved categorizing all traffic as – (i) no response required, such as broadcast, bridging, and routing packets, (ii) minimal response required, such as ARP and ICMP packets, and (iii) wake-up required, such as TCP SYN and SNMP Get packets. Traces from a campus PC indicated that protocol proxying could discard or minimally respond to 91% of all incoming packets. Split TCP connections involved adding a shim layer between the application and transport socket interface, which allowed the application to be presented with a persistent connection while allowing the underlying client to sleep. The initial experiments confirmed the feasibility of this approach. To avoid hardware changes to PC NICs, placing the proxy functionality on the LAN switches was discussed in [30].

Allman et al. [31] discussed the various architectural components required to support selective connectivity of end systems – (i) assistant – the mechanism which filtered traffic and took the corresponding action, (ii) exposing selective connectivity –making the protocol stack and the neighbors aware about the end system's energy state, (iii) evolving soft state – maintaining a proxyable state or a limbo state which enabled the distinction between a sleeping host and a non-existing host, (iv) host-based control – which ensured that the end system had the capability to dictate its selective connectivity policy, which allowed different policies to coexist in one network, (v) application primitives – making applications proxy-aware by using less number of general primitives instead of more specifics, and (vi) security – understanding the various vulnerabilities of offloading end system functionality to an external proxy. Purushothaman et al. [38] analyzed proxying for the specific Peer-to-Peer application Gnutella. The experiments confirmed that end hosts could spend large amounts of time sleeping both while downloading or uploading files. A prototype of such a proxy was developed in [39].

Nedevschi et al. [32] performed an in-depth evaluation of the potential of energy savings and the effectiveness of using proxying as a solution. Network data collected from 250 enterprise end systems indicated that incoming traffic comprised of significant portions of unicast, multicast, and broadcast traffic, therefore suggesting a need to tackle all three traffic classes by the proxy. The authors identified both broadcast protocols, such as ARP, DHCP and NBNS, and multicast protocols, such as HSRP, PIM and IGMP, which were suitable candidates to be either discarded by the proxy or requiring a simple response without having to wake up the end system. A prototype proxy is implemented using the Click Modular Router and the experiments performed showed that additional delays incurred due to proxying were minimal and lower than TCP SYN timeouts. Sabhanatarajan and Gordon-Ross [40] presented a partitioned TCAM-based proxying technique for Smart-NICs (SNIC). The simulation results indicated 62% lower energy-delay as compared to the existing non-partitioned router approaches.

Agarwal et al. [36] developed a proxy prototype, Somniloquy, employing USB-based network interfaces with the capability to support BitTorrent, instant messaging and web download applications. [34] and [35] studied and presented the generic architecture of a network connectivity proxy and its responsibilities. The major research challenges identified for this approach include high memory and processing requirements of the proxy, determining its optimal location in the network, awareness about end systems' energy states, application independency and support for mobile hosts or over different subnets. The concept of thin clients is presented in [33] in which low-power client machines replace end user desktops with the applications being hosted on another dedicated machine serving multiple thin clients. Experimental results indicated energy savings of upto 66% as compared to that of a traditional desktop environment.

3.4. Store and forward

The techniques in this category aspire to predict, control, and make the most use of idle times [44]. To enhance the typical sleeping of network elements techniques, Nedevschi et al. [8] proposed shaping traffic into small bursts at the edge of the network allowing the edge devices to transmit packets in bursts throughout the network in order to increase the sleep times possible for all devices. The concept of a buffer interval is introduced wherein an ingress router buffers all incoming traffic for some time and periodically transmits all buffered traffic in a burst to its neighbors, allowing for alternating periods of sleep and transmissions. A practical algorithm is presented in which routers aggregate packets targeted for one destination together and after every buffer interval, the bursts are transmitted serially. This allowed for an ingress router to receive packets from multiple upstream routers as a single burst. Comparing with the typical wake-on-arrival sleeping techniques, the uncoordinated buffer and burst technique performed significantly better for constant bit rate traffic. The simulation results indicated lower transition times and hierarchical topologies would benefit this approach.

Intuitions in this area originated from delay-tolerant networking [43] which was a store and forward architecture for the interplanetary Internet to allow data to propagate opportunistically as connectivity subsequently allowed. Baldi and Ofek [41] proposed leveraging pipeline forwarding of IP packets in order to save network energy. The concept of pipeline forwarding was introduced in [42] wherein IP switches synchronized transmissions by either using a central time authority, like GPS, or a distributed network time protocol, in order to achieve deterministic QoS. The authors in [41] described a parallel network based on WDM-based fiber infrastructure, implementing time-based scheduling, in conjunction with the Internet. The objective was to shunt a large portion of the traffic from the Internet to this ‘super-highway’ to achieve dictated performance and energy savings.

The authors in [45] analyzed the opportunistic sleeping algorithm proposed in [7] and concluded that queuing packets for some time before transmitting allowed for fewer transitions and therefore reduced the energy costs related with frequent state changes. In [46], experiments were performed to compare frame and burst transmissions for Energy Efficient Ethernet (EEE), and the results indicated energy savings ranging from 5 to 70% for end users and of about 50% for data centers. [47] presented an analytical model for energy saving using burst transmissions in EEE, adding the elements of maximum allowed queue size and the maximum added queuing time to the 802.3az model. In [48], the authors provided an analytical comparison of the frame and burst modes to quantify the efficiency of each method. The comparison results indicated significantly higher energy savings using burst mode with a bounded increase in delay.

Staessens et al. [23] proposed employing burst mode operation in OpenFlow-based networks wherein packets are buffered at a node and then transmitted at the maximum rate. The authors mentioned that the approach worked on small scales allowing the number of elements to be switched off between bursts to be limited and emphasized the need for large packet buffers for efficient operation. An additional OpenFlow message, OFPC_BURST_MODE, was proposed which allowed switches to advertise their burst mode capability to the SDN controller.

3.5. Network traffic aggregation

Chabarek et al. were among the earliest researchers to benchmark and study the energy consumption of network devices [49]. The work involved creating a generic energy consumption model, optimizing the multicommodity network-flow problem by formulating it as a mixed-integer problem and solving it to find minimum-power system configurations. The concept was to leverage the relationship between energy consumption, network configuration and provisioning, and experimental results indicated that substantial amounts of energy could be saved by incorporating energy awareness into existing routing protocols. Mahadevan et al. [20] proposed Network Traffic Consolidation (NTC) wherein traffic is engineered to flow over fewer links allowing non-utilized links and switches to be turned off. A service level-aware NTC approach is also proposed which incorporates path-availability constraints into the scheme to sacrifice some energy efficiency at the cost of additional network redundancy. A web 2.0 workload is simulated in a tiered data center topology and the results showed that using the NTC scheme allows for a 58% reduction in the energy consumption, while the service level-aware scheme allows for a 16% reduction with minimal latency increase.

Cianfrani et al. [50] leveraged the link-state exchange behavior of routing protocols, such as OSPF, to design a network-wide strategy to save energy. The proposed Energy-Aware Routing (EAR) algorithm is three-phased – (i) electing some routers as exporters, typically the ones having the maximum number of neighbors, which are used to calculate minimum-power trees (MPT), (ii) the remaining routers, called importers, use the exporters as reference to run a modified version of Dijkstra's algorithm to detect powered down links, and (iii) computing the shortest-path trees (SPT) based on the modified topology. The objective of the algorithm was to consider a subset of routers' SPTs to select routing paths allowing some links to be powered off, and the experiment results assuming a real IP network topology showed that more than 50% of the links could be switched off. Vasic and Kostic [51] presented Energy-Aware Traffic engineering (EATe), a distributed online technique that leverages sleeping of links and routers to save energy by spreading the traffic load over multiple links. The technique employed is to shift all traffic from the links and routers with the minimum utilization to the remaining resources, allowing the network to become energy proportional to the traffic load.

Heller et al. [52] proposed ElasticTree which aimed to reduce the energy consumption of data center networks by turning off switches and links not required during times of low-utilization. The work involved collecting traffic statistics such as the topology, current traffic matrix, and the desired fault tolerance levels, modelling and optimizing the problem while satisfying all constraints, and re-provisioning the devices using the OpenFlow protocol to maintain a minimum power subset of the network. Three different models are presented – a formal model based on a standard multicommodity flow problem, a greedy-bin packing model, and a topology-aware heuristic model requiring only the port counters as the input. The experiments conducted on tree-based topologies like Fat-Tree indicated a potential to save up to 50% of network energy with the ability to handle traffic fluctuations. Zhang et al. [53] proposed a centralized energy-aware traffic engineering scheme, GreenTE, combining device, component and link-level solutions. The general TE problem is formulated as a multicommodity flow with the specific variables of link and line-card energy states modeled using mixed-integer programming (MIP) problem. Since MIP problems are generally NP-hard, practical heuristics such as constraints on the maximum link utilization and the use of candidate paths instead of searching in all paths are used to reduce the computation time. The simulation results using production topologies from Abilene and GEANT showed a reduction in the energy consumption of line-cards' by 27-42% while maintaining link utilization levels below 50%. The energy-aware routing algorithm was also formulated as an integer linear programming (ILP) problem in [54], and solved analytically for links in [55], and for both links and nodes in [56]. Puype et al. [57] proposed to leverage multilayer traffic engineering (MLTE) to traffic engineer around and shut down energy inefficient portions of the IP-over-optical networks. In [58], the authors presented the Responsive Energy-Proportional Networks (REsPoNse) framework that proactively identified energy-critical paths by analyzing the traffic matrices, installed the corresponding routing entries into three different route tables – always-on, on-demand, and failover – and reactively modified the state of network elements according to the demand. The objective was to overcome the high computation times of computing energy-aware routing tables by analyzing the tradeoff between optimality and scalability. Mohammadpour and Bakhshi [61] employed a realistic power model of network devices from [63] to develop a routing algorithm, RLA-ENAR, formulated as an ILP problem and the experiments using Abilene network showed 40% more energy savings as compared to OSPF-TE.

Wang et al. proposed CARPO [59], a correlation-aware power optimization algorithm that consolidated network flows onto a subset of switches and links in a data center network allowing for the remaining network elements to be switched off. The work involved understanding the correlations between flows for the purpose of traffic aggregation, developing a heuristic algorithm and using OpenFlow to modify flow entries accordingly. The experiments performed on Wikipedia traces showed the potential to save 46% of network energy in a data center. In [79], Chiaraviglio et al. formulated the problem of reducing the power consumption of backbone networks as an ILP formulation. To reduce the computation time of the proposed algorithm, additional constraints, reduced notations and simple heuristics are employed. The algorithms aimed to find the minimal set of routers and links to satisfy a given traffic demand under connectivity and quality-of-service constraints, while assuming that the traffic matrix at a given time and the power consumption of each link and router is known. The test results from both real and synthetic topologies indicated that up to 35% of power could be saved, especially during off-peak times, when traffic is low. While virtual-machine (VM) consolidation to save server energy in data centers had received much research focus, the authors in [60] leveraged this VM migration approach to power off unused switches and attempt to increase the number of inactive switches. The proposed approach involved adding bypass links between the physical machine, called a honey machine, and upper-tiered switches, aggregating all VM's on one rack connecting to the bypassed switch to the honey machine, and powering off the unused switch to enable energy savings. The simulation results showed that the energy savings using this approach were up to 7.8% in a fat tree network as compared to the conventional VM-migration schemes. Son et al. [90] developed a VM-consolidation scheme based on the historical monitoring data of the host and network utilization. The algorithm first groups VMs based on their connectivity, and then sorts VM groups according to their resource requirements. Connected VM groups are consolidated on a single host in order to minimize the number of active transit switches. Also, VMs are migrated from over-utilized hosts to partially utilized hosts in order to maintain SLA requirements.

In [86], Zhu et al. discussed the energy efficiencies of common routing and scheduling algorithms by analyzing their traffic aggregation capabilities. The simulation results using Fat Tree and 3-level BCube data center topologies indicated that using priority based shortest routing – selecting the highest priority, i.e. the least congested, path amongst all shortest paths – with exclusive flow scheduling – transferring flows one by one and allowing each flow exclusive use of the total link bandwidth – was the most energy efficient strategy. Wei et al. [87] studied the energy-efficient traffic engineering problem in hybrid SDN/IP networks. The proposed fast heuristic algorithm aimed to reduce the number of active links in the network and power them off by optimizing OSPF link weights and traffic-splitting ratios in SDN devices. OSPF link weights were optimized for energy efficiency by increasing the costs of congested or sleeping links, in order to concentrate traffic on the least amount of links. SDN traffic was optimized for energy efficiency by moving traffic flows from low-utilization links to high-utilization links. The simulation tests performed in NS2 on real topologies indicated an energy savings improvement of 13.2% on exiting energy aware OSPF algorithms. In [95], the authors considered both the data plane and in-band control plane traffic to minimize the number of links required to satisfy a given traffic deman. [88] presented an energy monitoring and management application (EMMA) for SDN-based 5G backhaul networks. The scheme aimed to limit the number of active links and nodes by trying to fit any new flow into the current active network while meeting the flow requirements. If no suitable path was found, additional links and/or nodes were turned on, while also checking for possible best paths for existing flows. The algorithm was implemented using Mininet and the ONOS SDN controller, and experimental results showed that EMMA performed very close to the optimum solution.

Maleki et al. [93] presented a method to reduce the number of active links in the network by leveraging SDN features considering the GEANT network. The proposed scheme – Shared Path First (SPF) – calculated the shortest path between a source and destination for the first traffic flow from a particular source. For the subsequent flows from that source, existing active parts were first checked to determine if the new capacity requirements can be catered to, if not, new shortest paths are considered to accommodate the new flow. The experiment results indicated that 41% of links could be saved as compared to the shortest path first approach. [91] presented an approximate algorithm to save energy in cloud-based content distribution systems (CDNs). The strategy involved processing historical traffic data to determine the maximum value of traffic for each hour, which in turn was used to ascertain the number of vCDN functions required to be deployed. Next, the autoregressive integrated moving average (ARIMA) static forecasting model is employed to predict the future traffic load. Based on that and the number of redundant functions required, the minimum number of vCDN functions was computed. In [94], the authors leveraged the IEEE 802.3az functionality in bundles of links to reduce the operation costs in data centers and wired access networks. They presented several algorithms to select output links in bundles in order to increase the effective time they could be out in low-power modes while satisfying the QoS requirements of different applications. The greedy algorithm (GA) filled links to their maximum capacities before allocating new links, the bounded greedy algorithm avoided filling links to their full capacities in order the bound the packet delay and losses in GA, the conservative algorithm evenly spread the load amongst the minimum number of links in a bundle determined in a time interval. Additional modifications are also discussed such as the spare port algorithm wherein all best-effort traffic is concentrated on the least number of links, while the low-latency flows are mapped on to the remaining unused links in the bundle, and the two queues algorithm, wherein two queues are created for the best-effort traffic (low-priority queue) and the low-latency traffic (high-priority traffic) for each physical link. The experiments were performed on the ONOS SDN controller using real network traffic obtained from CAIDA, and the results indicated that the proposed algorithms consumed at least 18% less link energy than the baseline uniform-distribution algorithm. [97] proposed a bi-level optimization problem for ISP networks where the upper level represented the energy management function, and the lower level represented a multi-path routing protocol. Then, it was reformulated as a one-level MILP replacing the second level problem by different sets of optimality conditions. Numerical evaluations on Abilene, Geant and Polska networks indicated that the iterative cutting plane and branch-and-cut algorithms were close in terms of CPU time.

4. COMPLEMENTARY WORK

This section presents three additional categories of work – modeling and measurement, standards work, and energy efficiency metrics – which are closely related to the work presented in Section III.

4.1. Modeling and measurement

Research efforts have been put into modeling the energy consumption of a network device and its components. The work in [64] involved estimating the energy consumption of packet switching fabrics using statistical modeling. The discrete-time batch Markovian arrival process is used to create a stochastic traffic model and the results emphasized the importance of moving the energy optimization process from the circuitry to the system level. [65] and [66] employed analytical models to estimate the energy consumption of on-chip switching interconnects. The approach in

[67] was an architectural-level estimation aimed to incorporate the realtime nature of dynamic contention between packets into the model. The simulation results led to important conclusions – storing a packet in the buffer consumed far more power than transmitting it on the interconnect, energy consumed by the buffers was a significant part of the total energy consumption, and buffer energy would increase as the packet flow throughput increases. The model developed in [63] incorporated per-packet processing and per-byte store and forward handling estimations.
Chabarek et al. [49] measured the power consumption of two widely-used routers to come up with a generic power consumption model. The authors simulated a range of different configurations and operating conditions to ensure the comprehensiveness of the model. The experiments revealed that the power consumption of a router was dependent on the underlying chassis, the installed line cards, and the configuration and traffic utilization of the device. Mahadevan et al. [68] presented a power measurement study of a variety of network devices such as hubs, edge switches, core switches, routers, and wireless access points. The experiment observations noted that the power consumed by a network device is determined by factors such as the number of active ports, the line speed configured for each port, and the firmware version on the device, while traffic utilization and packet size had minimal-to-no impact. In [78] they present a large power profile study conducted in an enterprise network, comprising of 90 live switches from various vendors. The work in [69] showed that the energy consumption depends on the volume of computed traffic and device reconfigurations, while queue management policies and BGP updates had no impact. In [83], Orgerie et al. propose an end-to-end cost model and simulator for evaluating power consumption in large-scale networks. Their model computes the energy consumption per equipment depending on the amount of bandwidth traversing, the length of a transfer, and the type of equipment. The simulator module is integrated with NS2 and takes user inputs such as a network topology, network traffic, and the energy consumption values for the network equipment used computed from their model. The simulator also supports advanced functionalities such as dynamic on/off links and adaptive link rate. The authors extend their work in [84] to redesign the simulator to be integrated with NS3 which allows running of native Linux code to provide more accurate consumption values.

Chabarek et al. extended their prior work in [70] to develop a network-wide power consumption framework. The work involved employing an application programming interface to interface with existing network management tools, comparing the device configurations with community database benchmark measurements, and auditing to infer any missing data to enable power consumption estimations. Hossain et al. [92] measured and modeled the energy consumption of Ethernet switches considering parameters such as link speeds, traffic, the number of connections. Additionally, full factorial and linear regression analysis is performed to identify the most influential parameters. Their results showed that link capacities and the number of connections have an impact on the power consumption while changing traffic had little impact. [85] modeled energy consumption of the software stacks of Ethernet and Infiniband NICs related to VM migration. The experiment results indicated that transferring the same quantity of data over Infiniband in connected mode was more energy efficient as compared to Infiniband in datagram mode or Gigabit Ethernet. However, for message centric traffic, wherein the volume of effective data is low and the number of transmitted packets is high, Gigabit Ethernet was more efficient, closely followed by Infiniband in datagram mode.In [86], Zhu et al. developed a network energy monitoring prototype using OpenNaaS to obtain energy usage information from OpenFlow switches. They used SNMP to fetch and control the power state of switches and ports by creating power meter drivers for SNMP access to different meter vendors.

4.2. Standards work

The European Telecommunications Standards Institute (ETSI) published a standard, ES-203 237 – the Green Abstraction Layer, that specified the Green Standard Interface (GSI) for a uniform

way for interactions between the energy-aware hardware and the control framework [71]. The proposed GSI intended to provide the functionalities of discovery – control plane retrieving information about the different energy states supported by the data plane, provisioning – control plane configuring different energy states, and monitoring – exchanging relevant device parameters. The goal of this standard was to represent an abstraction of the energy-aware capabilities of networking devices to higher-layer protocols.
The Internet Engineering Task Force (IETF) published the Energy Management Framework (eMAN) [72], which presented a physical reference model and an information model for devices and device components within, or connected to, networks. The framework modeled relationships and capabilities between energy objects such as power, power state, energy, demand, power attributes and battery. This was the first attempt to standardize monitoring and control for power and energy of networked devices using a Management Information Base (MIB) [73].

The Society of Cable Telecommunications Engineers' (SCTE) Energy Management Subcommittee (EMS) published a standard ANSI/SCTE 216 2015 – Adaptive Power System Interface Specification (APSIS) [74]. They proposed interfacing APSIS applications with the network elements over standard management protocols such as SNMP, NETCONF, IPDR, and HTTP. Other standards such as SCTE 184 – providing guidelines for balancing energy-efficient operations with essential business requirements, SCTE 211 – defining energy metrics, and ANSI/SCTE 212 – defining a framework to establish energy baselines, were also published.

4.3. Energy efficiency metrics

Metrics have been proposed in this field to measure and compare the impact of energy saving schemes; [75] surveyed and presented them. Equipment-level metrics included ECR – energy consumption in Watt/Gbps, EER – energy efficiency in Gbps/Watt, EPI – energy proportionality in percentage, and FLOPS per watt – peak power in terms of operations per second. Facility-level metrics included PUE – Power Usage Effectiveness is the ratio of total data center power to the power drawn by IT equipment, DCiE – Data Center infrastructure Efficiency in percentage, and DCP – Data Center Productivity measured the amount of useful work done by the data center. Country-level metrics included EPI – Environmental Performance Index, ESI – Environmental Sustainability Index, and EVI – Environmental Vulnerability Index.

5. CONCLUSIONS

This paper presented a detailed survey of energy-efficiency approaches in wired networks focusing on energy-aware protocols and network design. We categorized the existing work into different broad categories – sleeping of network elements, link rate adaptation, proxying, store and forward, and network traffic aggregation - and described the specific research efforts in each category. Additionally, work in modeling and measurement, standards, and metrics is also discussed. The research challenges, test results, and important conclusions are indicated to make this survey holistic. Below we examine discussion points for future work in this field.

5.1. Energy proportionality does not imply energy efficiency

While research efforts are being invested to make devices more energy proportional and energy efficient, it is to be noted that one does not imply the other. As pointed out in [68], while some devices may consume power more proportionally to their load, higher values of absolute consumption still make them inefficient. Contrarily, some devices may consume less power while transmitting the same amount of traffic as others, negligible variations with respect to traffic load make them non-energy proportional. The efforts concentrated on future networking hardware must take this into consideration.

5.2. Energy efficiency does not imply decreased energy consumption

The fact that per unit computational power and unit energy efficiencies are improving does not mean that the energy consumption of the Internet is decreasing. Absolute numbers suggested an upward trend in the energy consumed and it is projected to continue increasing [76]. Improving energy efficiency reduces the implicit cost to use the resource making it more affordable and thus leading to increased use of the resource – a phenomenon known as the rebound effect in energy economics [77]. Therefore, future efforts should be focused on decreasing the absolute energy consumption of the Internet to ensure its long-term sustenance.

5.3. Incorporate energy efficiency into network management

Network management has traditionally comprised of five major areas – fault, configuration, accounting/administration, performance, and security. However, the criticality of energy management warrants its inclusion in the traditional definition of network management to ensure that both academia and industry do not consider energy management an afterthought.

5.4. User awareness

While the supply side of the information and communications technology industry has started to understand the importance of energy efficiency, the demand side still lags. Since end users are aware and responsible for remunerating a small portion of the total energy costs, the urgency has not trickled down. We recommend educational and governmental efforts to enable everyone to make energy-conscious decisions.

REFERENCES

[1] Internetworldstats.com. (2019). World Internet Users Statistics and 2019 World Population Stats. [online] Available at: https://www.internetworldstats.com/stats.htm [Accessed 11 October 2021].

[2] J. Koomey and S. Naffziger, Efficiency's brief reprieve: Moore's Law slowdown hits performance more than energy efficiency, in IEEE Spectrum, pp. [http://spectrum. ieee. org/computing/hardware/moores-law-might-be-slowing-down-but-not-energyefficiency], Apr. 2015.

[3] M. Gupta and S. Singh, Greening of the Internet, in Proceedings of ACM SIGCOMM ’03, Karlsruhe, Germany, Aug. 2003.

[4] K. W. Roth, F. Goldstein, and J. Kleinman, Energy Consumption by Office and Telecommunications Equipment in Commercial Buildings Volume I: Energy Consumption Baseline, Tech. Rep. Vol I, National Technical Information Service (NTIS), US Department of Commerce, Jan. 2002.

[5] M. Gupta, S. Grover, and S. Singh, A Feasibility Study for Power Management in LAN Switches, in Proceedings of 12th IEEE International Conference on Network Protocols (ICNP), Berlin, Germany, pp. 361-371, Oct. 2004.

[6] V. Soteriou and L.-S. Peh, Dynamic power management for power optimization of interconnection networks using on/off links, in Proceedings of the 11th Symposium on High Performance Interconnects, pp. 15-20, 2003.

[7] M. Gupta and S. Singh, Using Low-Power Modes for Energy Conservation in Ethernet LANs, in Proceedings of the 26th Annual IEEE Conference on Computer Communications (INFOCOM 2007), (Anchorage, Alaska), pp. 2451 – 2455, May 2007.

[8] S. Nedevschi, L. Popa, G. Iannaccone, S. Ratnasamy, and D. Wetherall, Reducing Network Energy Consumption via Sleeping and RateAdaptation, in Proceedings of the 5th USENIX Symposium on Networked Systems Design and Implementation (NDSI2008), (San Francisco, California, USA), Apr. 2008.

[9] G. Ananthanarayanan and R. H. Katz, Greening the Switch, in Proceedings of the USENIX Workshop on Power Aware Computing and Systems (HotPower), held at the Symposium on Operating Systems Design and Implementation (OSDI 2008), (San Diego, California, USA), Dec. 2008.

[10] L. Chiaraviglio, M. Mellia, and F. Neri, Energy-aware Backbone Networks: a Case Study, in Proceedings of the 1st International Workshop on Green Communications (GreenComm) in conjunction with the IEEE International Conference on Communications, (Dresden, Germany), June 2009.

[11] K. Christensen, P. Reviriego, B. Nordman, M. Mostowfi, and J. A. Maestro, IEEE 802.3az: the road to energy efficient ethernet, IEEE Communications. Magazine, vol. 48, no. 11, pp. 50–56, Nov. 2010.

[12] H. Sedarat, O. Barkan, and W. Woodruff, Low-power idle mode for network transceiver, United States Patent 8156359, Sep.2009.

[13] K. Christensen, B. Nordman, and R. Brown, Power management in networked devices, IEEE Computer, vol. 37, pp. 91–93, Aug.2004.

[14] C. Gunaratne, K. Christensen, and B. Nordman, Managing energy consumption costs in desktop PCs and LAN switches with proxying, split TCP connections and scaling of link speed, International Journal of Network Management, vol. 15, pp. 297–310, Sep. 2005.

[15] C. Gunaratne, K. Christensen, and S. W. Suen, Ethernet Adaptive Link Rate (ALR): Analysis of a buffer threshold policy, in Proceedings of the IEEE Global Communications Conference (GLOBECOM 2006), (San Francisco, California, USA), Nov. 2006.

[16] C. Gunaratne and K. Christensen, Ethernet Adaptive Link Rate: System Design and Performance Evaluation,in Proceedings of the IEEE Conference on Local Computer Networks, pp. 28-35, Nov. 2006.

[17] C. Gunaratne, K. Christensen, B. Nordman, and S. Suen, Reducing the Energy Consumption of Ethernet with Adaptive Link Rate (ALR), IEEE Transactions on Computers, vol. 57, pp. 448–461, Apr. 2008.

[18] H. Anand, C. Reardon, R. Subramaniyan, and A. George, Ethernet Adaptive Link Rate (ALR): Analysis of a MAC handshake protocol, in Proceedings of the 31st IEEE Conference on Local Computer Networks, pp. 533–534, Nov. 2006.

[19] F. Blanquicet and K. Christensen, An Initial Performance Evaluation of Rapid PHY Selection (RPS) for Energy Efficient Ethernet, in Proceedings of the 32nd IEEE Conference on Local Computer Networks, pp. 223-225, Oct. 2007.

[20] P. Mahadevan, P. Sharma, S. Banerjee, and P. Ranganathan, Energy Aware Network Operations, in Proceedings of the IEEE Global Internet Symposium, (Rio de Janeiro, Brazil), Apr. 2009.

[21] D. Abts, M. R. Marty, P. M. Wells, P. Klausler, and H. Liu, Energy proportional datacenter networks, SIGARCH Computer Architecture News, vol. 38, no. 3, pp. 338–347, Jun. 2010.

[22] G. Ginis, Low-Power Modes for ADSL2 and ADSL2+, SPAA021, Broadband Communications Group, Texas Instruments, Jan. 2005.

[23] D. Staessens, S. Sharma, D. Colle, M. Pickavet, and P. Demeester, Software defined networking: Meeting carrier grade requirements, in Proceedings of the 18th IEEE Workshop LANMAN, pp. 1–6, 2011.

[24] C. Callegari, R. G. Garroppo, S. Giordano, and G. Nencioni, A new Markov model for evaluating the ALR Dual-Threshold Policy, 2nd IEEE Workshop on Green Communications, Hawaii, USA, Nov. 30 – Dec. 4, 2009.

[25] S. Herrería-Alonso, M. Rodríguez-Pérez, M. Fernández-Veiga, and C. López-García, Opportunistic power saving algorithms for Ethernet devices, Computer Networks, vol. 55, no. 9, pp. 2051–2064, Jun. 2011.

[26] K. J. Christensen, C. Gunaratne, B. Nordman, and A. D. George,The next frontier for communications networks: Power management, Computer Communications, vol. 27, pp. 1758–1770, 2004.

[27] J.A. Robertson et al., After-hours Power Status of Office Equipment and Energy Use of Miscellaneous Plug-Load Equipment, Lawrence Berkley Laboratory, May 2004.

[28] AMD, Magic Packet Technology, Publication# 20213, Rev. A, Nov. 1995.

[29] Advanced Configuration and Power Interface Specification, Rev. 5.0a, Nov. 2013.

[30] K. Christensen and F. Gulledge, Enabling Power Management for Network-Attached Computers, International Journal of Network Management, vol. 8, no. 2, pp. 120–30, Apr. 1998.

[31] M. Allman, K. Christensen, B. Nordman, and V. Paxson, Enabling and Energy-Efficient Future Internet Through Selectively Connected End Systems, Proc. ACM SIGCOMM HotNets Workshop (HotNets 07), Atlanta, GA, Nov. 2007.

[32] S. Nedevschi et al., Skilled in the Art of Being Idle: Reducing Energy Waste in Networked Systems, in Proceedings of the. 6th ACM/USENIX Symposium on Networked Systems Design and Implementation (NSDI 2009), Boston, MA, USA, Apr. 2009.

[33] W. Vereecken et al., Power Efficiency of Thin Clients, European Transactions on Telecommunications, vol.21, no. 6, pp. 479-490, 2010.

[34] R. Khan, R. Bolla, M. Repetto, R. Bruschi, and M. Giribaldi, Smart Proxying for Reducing Network Energy Consumption, in IEEE International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS), July 2012.

[35] R. Bolla, M. Giribaldi, R. Khan, and M. Repetto, Network Connectivity Proxy: An Optimal Strategy for Reducing Energy Waste in Network Edge Devices, in The 24th Tyrrhenian International Workshop on Digital Communications, 2013.

[36] Y. Agarwal et al., Somniloquy: augmenting network interfaces to reduce pc energy usage, in Proc. of the USENIX symposium on Networked systems design and implementation (NSDI'09), 2009.

[37] M. K. Girish, B. Zhou, and J. Q. Hu, Formulation of the traffic engineering problems in MPLS based IP networks, in Proceedings of the 5th IEEE Symposium on Computers and Communications (ISCC 2000), pp. 214-219, 2000.

[38] P. Purushothaman, M. Navada, R. Subramaniyan, C. Reardon, and A. D. George, Power-Proxying on the NIC: a Case Study with the Gnutella File-Sharing Protocol, in Proceedings of the 31st IEEE Conference on Local Computer Networks (LCN 2006), Nov. 2006.

[39] M. Jimeno and K. Christensen, A Prototype Power Management Proxy for Gnutella Peer-to-Peer File Sharing, in Proceedings of the 32nd IEEE Conference on Local Computer Networks (LCN 2007),Oct. 2007.

[40] K. Sabhanatarajan and A. Gordon-Ross, A Resource Efficient Content Inspection System for Next Generation Smart NICs, in Proceedings of the IEEE International Conference on Computer Design (ICCD 2008), pp. 156–163, Oct. 2008.

[41] M. Baldi and Y. Ofek, Time for a Greener Internet, in Proceedings of the 1st International Workshop on Green Communications (GreenComm) in conjunction with the IEEE International Conference on Communications,June 2009.

[42] C.-S. Li, Y. Ofek, and M. Yung, Time-driven priority flow control for real-time heterogeneous Internetworking, IEEE INFOCOM'96, Mar. 1996.

[43] V. Cerf et al., Delay-Tolerant Networking Architecture, IETF Request for Comments #4838, April 2007.

[44] A. Akella, R. Balan and S. Seshan.. Protocols for Low Power, SIGCOMM CCR, Jan. 2002.

[45] M. Rodríguez-Pérez, S. Herrería-Alonso, M. Fernández-Veiga, and C. López-García, Improved opportunistic sleeping algorithms for LAN switches, in Proceedings of the IEEE Globecom, Honolulu, HI, USA, Dec. 2009.

[46] P. Reviriego, J. A. Maestro, J. A. Hernandez, and D. Larrabeiti, Burst transmission for Energy-Efficient Ethernet, IEEE Internet Comput., vol. 14, no. 4, pp. 50–57, Jul. 2010.

[47] S. Herrería-Alonso, M. Rodríguez-Pérez, M. Fernández-Veiga, and C. López-García, A power saving model for burst transmission in energy-efficient Ethernet, IEEE Commun. Lett., vol. 15, no. 5, pp. 584– 586, May 2011.

[48] S. Herrería-Alonso, M. Rodríguez-Pérez, M. Fernández-Veiga, and C. López-García, How efficient is energy-efficient ethernet?, in 3rd International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), pp. 1-7. 2011.

[49] J. Chabarek et al., Power Awareness in Network Design and Routing, in Proc. IEEE 27th IEEE Conf. on Computer Communications (INFOCOM 2008), Phoenix, AZ,pp. 457- 465, Apr. 2008.

[50] Cianfrani, V. Eramo, M. Listanti, M. Marazza, and E. Vittorini, An Energy Saving Routing Algorithm for a Green OSPF Protocol, in INFOCOM IEEE Conference on Computer Communications Workshops, pp. 1-5, Mar. 2010.

[51] N. Vasic and D. Kostic, Energy-Aware Traffic Engineering, in Proceedings of the 1st International Conference on Energy-Efficient Computing and Networking, Apr. 2010.

[52] B. Heller et al., ElasticTree: Saving Energy in Data Center Networks, in Proc. of USENIX Symposium on Networked Systems Design and Implementation (NSDI), Apr. 2010.

[53] M. Zhang, C. Yi, B. Liu and B. Zhang, GreenTE: Power-Aware Traffic Engineering, in Proceedings of the 18th IEEE International Conference on Network Protocols, pp. 21-30, Oct. 2010.

[54] L. Chiaraviglio, M. Mellia, and F. Neri, Reducing Power Consumption in Backbone Networks, in Proceedings of the IEEE International Conference on Communications (ICC 2009), (Dresden, Germany), June 2009.

[55] W. Fisher, M. Suchara, and J. Rexford, Greening Backbone Networks: Reducing Energy Consumption by Shutting Off Cables in Bundled Links, in Proceedings of 1st ACM SIGCOMM workshop on green networking, (New Delhi, India), Aug. 2010.

[56] A. Bianzino, C. Chaudet, D. Rossi, and J.-L. Rougier, Energy-Awareness in Network Dimensioning: a Fixed Charge Network Flow Formulation, in 1st International Conference on Energy-Efficient Computing and Networking (e-Energy 2010), Extended Abstract, (Passau, Germany), Apr. 2010.

[57] A. Puype, W. Vereecken, D. Cole, M. Pickavet, and P. Demeester, Multilayer traffic engineering for energy efficiency, Photonic Network Communications, vol. 21, no. 2, pp. 127-140, Apr. 2011.

[58] N. Vasic et al., Identifying and Using Energy-Critical Paths, in ACM CoNEXT, Dec. 2011.

[59] X. Wang and X. Wang, CARPO: Correlation-Aware Power Optimization in Data Center Networks, in Proc. of IEEE INFOCOM, pp. 1125-1133, Mar. 2012.

[60] H. Shirayanagi, H. Yamada, and K. Kono, Honeyguide: A VM Migration-Aware Network Topology for Saving Energy Consumption in Data Center Networks, IEICE TRANSACTIONS on Information and Systems, pp. 2055-2064, Sept. 2013.

[61] E. Mohammadpour and B. Bakhshi, RLA-ENAR: A Realistic Near-Optimal Energy-Aware Routing, in 24th International Conference on Software, Telecommunications and Computer Networks (SoftCOM), pp. 1-5, Sep. 2016.

[62] R. Gandotra and L. Perigo, SDNMA: A Software-Defined, Dynamic Network Manipulation Application to Enhance BGP Functionality, IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS), pp. 1007-1014, 2018.

[63] A. Vishwanat, K. Hinton, R. W. A. Ayre, and R. S. Tucker, Modeling Energy Consumption in High-Capacity Routers and Switches, IEEE Journal on Selected Areas in Communications, vol. 32, no. 8, pp. 1524-1532, Aug. 2014.

[64] G. Wassal and M. A. Hasan, Low-Power System-Level Design of VLSI Packet Switching Fabrics, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 20, no. 6, pp. 723-728, 2001.

[65] S. Patel, S. M. Chai, S. Yalamanchili, and D. E. Schimmel, Power Constrained Design of Multiprocessor Interconnection Networks, in Proceedings of the International Conference on Computer Design VLSI in Computers and Processors, pp. 408-416, 1997.

[66] A. Langen, A. Brinkmann, and U. Ruckert, High level estimation of the area and power consumption of on-chip interconnects, in Proceedings of the 13th Annual IEEE International ASIC/SOC Conference, pp. 297-301, 2000.

[67] T. T. Ye, L. Benini, and G. D. Micheli, Analysis of Power Consumption on Switch Fabrics in Network Routers, in Proceedings of the Design Automation Conference, pp. 524-529, 2002.

[68] P. Mahadevan, P. Sharma, S. Banerjee, and P. Ranganathan, A Power Benchmarking Framework for Network Devices, in Proceedings of IFIP Networking 2009, May 2009.

[69] Adelin, P. Owezarski, and T. Gayraud, On the Impact of Monitoring Router Energy Consumption for Greening the Internet, in Proceedings of the 11th IEEE/ACM International Conference on Grid Computing, pp. 298-304, 2010.

[70] J. Chabarek and P. Barford, Energy Audit: Monitoring Power Consumption in Diverse Network Environments, in Proceedings of the International Green Computing Conference, pp. 1-10, Jun. 2013.

[71] European Telecommunications Standards Institute, Green Abstraction Layer (GAL), ES 203 237 (ETSI Standard), 2014. [Online]. Available at: https://www.etsi.org/deliver/etsi_es/203200_203299/203237/01.01.01_60/es_203237v010101p.pdf.

[72] J. Parello, B. Claise, B. Schoening, and J. Quittek, Energy Management Framework, RFC 7326 (Informational), Internet Engineering Task Force, September 2014. [Online]. Available at: https://tools.ietf.org/html/rfc7326.

[73] M. Chandramouli, B. Claise, B. Schoening, J. Quittek, and T. Dietz, Monitoring and Control MIB for Power and Energy, RFC 7460 (Standards Track), March 2015. [Online]. Available at: https://tools.ietf.org/html/rfc7460.

[74] Society of Cable Telecommunications Engineers (SCTE), Adaptive Power System Interface Specification (APSIS), American National Standard, 2015. [Online]. Available at: https://www.scte.org/SCTEDocs/Standards/ANSI_SCTE%20216%202015.pdf.
[75] A. P. Bianzino, A. K. Raju, and D. Rossi, Apple-to-Apple: A Framework Analysis for Energy-Efficiency in Networks, Proc. of SIGMETRICS, 2nd GreenMetrics workshop, 2010.
[76] A. Raghavan and J. Ma, The Energy and Emergy of the Internet, in Proceedings of the 10th ACM Workshop on Hot Topics in Networks, Nov. 2011.
[77] J. D. Khazzoom, Economic implications of mandated efficiency in standards for household appliances, Energy Journal, vol. 1, no. 4, pp. 21–40, 1980.
[78] P. Mahadevan, S. Banerjee, and P. Sharma, Energy proportionality of an enterprise network, in Proceedings of the first ACM SIGCOMM workshop on Green networking, pp. 53-60, Aug. 2010.
[79] L. Chiaraviglio, M. Mellia, and F.Neri, Minimizing ISP network energy cost: Formulation and solutions, IEEE/ACM Transactions on Networking, vol. 20, no. 2, pp. 463-476, Apr. 2012.
[80] K.-H. Ho and C.-C. Cheung, Green distributed routing protocol for sleep coordination in wired core networks, in Proceedings of 6th International Conference on Networked Computing, pp. 1-6, May 2010.
[81] A. P. Bianzino, L. Chiaraviglio, M. Mellia, and J. L. Rougier, GRiDA: Green distributed algorithm for energy-efficient IP backbone networks, Computer Networks, vol. 56, no. 14, pp. 3219-3232, Sep. 2012.
[82] F. Patota et al., DAFNES: A distributed algorithm for network energy saving based on stress-centrality, Computer Networks, vol. 94, pp. 263-284, Jan. 2016.
[83] A.-C. Orgerie, L. Lefevre, I. Guerin-Lassous, and D. M. L. Pacheco, ECOFEN: An end-to-end energy cost model and simulator for evaluatingpower consumption in large-scale networks, in Proceedings of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, pp. 1-6, Jun. 2011.
[84] A.-C. Orgerie et al., Simulation toolbox for studying energy consumption in wired networks, in Proceedings of 13th International Conference on Network and Service Management (CNSM), pp. 1-5, Nov. 2017.
[85] V. D. Maio, R. Prodan, S. Benedict, and G. Kecskemeti, Modelling energy consumption of network transfers and virtual machine migration, Future Generation Computer Systems, vol. 56, pp. 388-406, Mar. 2016.
[86] H. Zhu, X. Liao, C. d. Laat, and P. Grosso, Joint flow routing-scheduling for energy efficient software defined data center networks, Journal of Network and Computer Applications, vol. 63, pp. 110-124, Mar. 2016.
[87] Y. Wei, X. Zhang, L. Xie, and S. Leng, Energy-aware traffic engineering in hybrid SDN/IP backbone networks, Journal of Communications and Networks, vol. 18, no. 4, pp. 559-566, Sep. 2016.
[88] S. S. Tadesse, C. Casetti, and C. F. Chiasserini, Energy-efficient traffic allocation in SDN-based backhaul networks: Theory and implementation, in Proceedings of 14th IEEE Annual Consumer Communications & Networking Conference (CCNC), pp. 209-215, Jan. 2017.
[89] K. Xie et al., E3MC: Improving energy efficiency via elastic multi-controller SDN in data center networks, IEEE Access, vol. 4, pp. 6780-6791, Oct. 2016.
[90] J. Son, A. V. Dastjerdi, R. N. Calheiros, and R. Buyya, SLA-aware and energy-efficient dynamic overbooking in SDN-based cloud data centers, IEEE Transactions on Sustainable Computing, vol. 2, no. 2, May 2017.
[91] Liao, G. Sun, G. Yang, and V. Chang, Energy-efficient virtual content distribution network provisioning in cloud-based data centers, Future Generation Computer Systems, vol. 83, pp. 347-357, Jun. 2018.
[92] M.M. Hossain, E. Rondeau, J.-P. Georges, and T. Bastogne, Modeling the power consumption of Ethernet switch, inProceedings of International SEEDS Conference: Sustainable Ecological Engineering Design for Society, pp. 469-480, Sep. 2015.
[93] A. Maleki, M. Hossain, J.-P. Georges, E. Rondeau, and T. Divoux, An SDN perspective to mitigate the energy consumption of core networks –GÉANT2, in Proceedings of International Sustainable Ecological Engineering Design for Society (SEEDS) Conference, pp. 233-244, Sep. 2017.
[94] P. Fondo-Ferreiro, M. Rodríguez-Pérez, M. Fernández-Veiga, and S. Herrería-Alonso, Matching SDN and legacy networking hardware for energy efficiency and bounded delay, Sensors, vol. 18, no. 11, Nov. 2018.

[95] A. Fernandez-Fernandez, C. Cervello-Pastor, and L. Ochoa-Aday, Achieving energy efficiency: An energy-aware approach in SDN, in Proceedings of IEEE Global Communications Conference (GLOBECOM), pp. 1-7, Dec. 2016.

[96] M. d. S. Conterato, T. C. Ferreto, F. Rossi, W. d. S. Marques, and P. S. S. d. Souza, Reducing energy consumption in SDN-based data center networks through flow consolidation strategies, in Proceedings of 34th ACM/SIGAPP Symposium on Applied Computing, pp. 1384-1391, Apr. 2019.

[97] Bouras, R. Figueiredo, M. Poss, and F, Zhou, Minimizing energy and link utilization in ISP backbone networks with multi-path routing: A bi-level approach, Optimization Letters, pp. 1-19, Nov. 2019.