Abstract

Cloud computing has recently emerged as a dominant Internet service computing model due to its “pay-as-you-go” and “elastic service” features. Cloud computing systems are usually composed of distributed datacenters, which leverage virtualization technology to provide a scalable and reliable service. Optical networks are recognized as promising next-generation core networks for connecting these distributed datacenters due to their characteristics such as high bandwidth provisioning, low latency, low bit error rate, etc. However, concern about the ever-increasing energy consumption of cloud computing systems together with core networks has been raised due to high electricity bills as well as environmental pollution. In this paper, we study the Energy-aware Provisioning in Optical Cloud Networks (EPOCN) problem for both dynamic and static cases. When traffic requests arrive in an online fashion, we propose a polynomial-time energy-aware routing algorithm to solve the dynamic EPOCN problem. Simulations show that our energy-aware routing algorithm brings more energy savings in comparison to a shortest path-based routing algorithm and a traffic grooming algorithm. On the other hand, we show that the EPOCN problem in the static case (the traffic matrix is known in advance) is NP-hard. We further divide this problem into (1) the Energy-Aware Routing (EAR) problem in optical networks and (2) the Energy-efficient Server and Switch Allocation (ESSA) problem in datacenter networks. Considering these two (sub)problems are still NP-hard, we present an exact Integer Linear Program (ILP) and a heuristic to solve each problem. We also conduct simulations to compare the proposed ILPs and heuristics in terms of energy consumption and running time.

Keywords: Energy-efficiency; Routing; Cloud computing; Datacenters; IP-over-WDM optical networks.
1. Introduction

Cloud computing [1] is a distributed computing and storing paradigm, which can provide a scalable and reliable service over the Internet for on-demand data-intensive applications (e.g., online search, video streaming, etc.) and data-intensive computing (e.g., weather forecasting, stock transactions data analyzing, etc.). Cloud computing has emerged as a dominant future Internet service-providing model for both service providers and customers, due to its “pay-as-you-go” and “elastic service” features. Distributed cloud systems are usually composed of distributed inter-connected datacenters, which leverage virtualization technology to provide computing and storage service for each on-demand request. Due to the huge amount of traffic needs from current bandwidth-intensive applications such as IPTV, designing a high-capacity and adaptive network to interconnect datacenters is indispensable. The IP-over-WDM networks [2] are considered to be a promising network architecture for next-generation core networks for their flexibility and the adaptability offered in the IP control protocols and characteristics in optical networks such as high bandwidth provisioning, low latency, low bit error rate, etc. In Wavelength Division Multiplexing (WDM) technology, the capacity of a fiber is divided into several non-overlapping wavelength channels that can transport data independently. These wavelength channels make up lightpaths that are used to establish optical connections that may span several fiber links. With current commercial technology, each lightpath can be independently operated at data rate ranging up to 100 Gbps [3]. However, traffic between a pair of nodes may not be able to fill up the available bandwidth of a lightpath. In order to efficiently utilize the available bandwidth, several independent traffic streams can be aggregated to share the capacity of a lightpath. This is known as traffic grooming (e.g., see [4]). While traffic grooming has obvious potential to increase throughput, the grooming of traffic may also lead to energy efficiency, although this is not always the case. In this paper, we consider IP-over-WDM networks as the transportation backbone networks to connect distributed datacenters for cloud computing. A cloud network, therefore, implies an integrated cloud provisioning infrastructure, where datacenters are located on the core nodes of IP-over-WDM networks.

Energy management [5, 6] is a big challenge in cloud networks. Datacenter is the main functional IT facility for cloud computing concerning data processing, storage, etc., and hence, it is also the dominant energy-consuming part. According to [5], a typical 500 square meter datacenter can consume 38 megawatt hours of power per day, which exceeds the power consumption of more than 3500 EU households. The scale of datacenters is expected to increase with the increasing growth of the traffic demand, and consequently the energy consumption. Based on a Cisco study [7], the datacenter traffic in 2014 was 3.4 Zettabytes, and this value is predicted to triple in 2019. Meanwhile, the energy consumption of telecommunication networks can also not be neglected. According to [8], the electricity consumption in networks only for telecom operator networks increased from 150 TWh in 2007 to 250 TWh in 2012, which corresponds to an annual growth rate of 10.8%. In addition, 250 TWh energy
in 2012 was over 2.5% of the worldwide electricity consumed. It is also estimated in [9] that the overall electricity efficiency for fixed access wired networks (which consists of core networks, metro/edge networks and content distribution networks) is expected to improve on the expected scenario by 10% per year from 2012 to 2030. The huge amount of energy consumed by cloud computing infrastructures is causing not only high electricity bills but also incurring severe environmental pollution (e.g., carbon dioxide emissions.). Due to these concerns, energy-efficiency in cloud networks is becoming a very crucial issue to be tackled.

In [10], we proposed a polynomial-time dynamic path selection algorithm in IP-over-WDM networks. In this paper, we extend our work to study the energy-aware traffic provisioning problem in optical cloud networks for both dynamic and static cases. We especially target how traffic grooming can bring energy savings in optical cloud networks. Our key contributions are as follows:

- We propose, via auxiliary graphs, a polynomial-time energy-aware routing algorithm for when the traffic requests arrive in an online fashion with and without wavelength conversion.
- When all the traffic requests are known in advance, we propose an exact Integer Linear Program (ILP) and a heuristic for each problem to (1) design energy-aware paths in optical networks and (2) allocate servers and switches in datacenters for accommodating the whole requests in an energy-efficient way.
- We conduct simulations to verify: (1) the proposed dynamic energy-aware routing algorithm, (2) the exact ILP solutions and (3) the respective heuristics in terms of energy consumption and running time.

The outline of this paper is as follows. Section 2 presents related work. Section 3 describes our network model and analyzes the energy consumption with traffic grooming and when setting up a new lightpath. A dynamic energy-aware routing algorithm is proposed in Section 4. When traffic matrix is assumed to know in advance, the static EPON problem turns into NP-hard. In Section 5, we decompose the static EPON problem into two NP-hard (sub)problems and present an ILP and a heuristic to solve each problem. Section 6 provides our simulation results for the proposed dynamic routing algorithm and static provisioning algorithms (ILPs and heuristics) in terms of energy consumption and running time. Finally, we conclude in Section 7.

2. Related Work

Energy efficiency in both datacenter networks and optical core networks has been extensively investigated due to its importance. A large quantity of papers has been proposed trying to solve this issue. To classify them, for instance, a comprehensive survey about datacenter energy consumption model is shown in
It covers (1) the hardware-centric aspect from digital circuit level to hardware, server, and datacenter level, until system level; (2) the software-centric approach about energy models developed for operating systems, virtual machines and software applications. More papers about Energy-aware resource allocation, energy-efficient architectures and traffic management in datacenter networks can be found in [12–15]. [16–19] provide a survey about energy efficiency in optical (core) networks, which analyze and summarize different energy consumption models of network devices, as well as a taxonomy to classify the different energy-aware approaches in terms of turning off idle equipment, energy-efficient network design, green routing, etc. In the following, we will briefly introduce some representative and related literature.

2.1. Energy-Efficient Traffic Provisioning in Optical Cloud Networks

Lawey et al. [20] study how to choose the optimum number of datacenters (clouds) and their locations in IP-over-WDM non-bypass networks to accommodate all the traffic requests with minimum energy consumption. They establish an Integer Linear Programming (ILP) as well as a heuristic to solve this problem. They also consider the energy efficient storage and energy efficient Virtual Machine (VM) placement problems. However, the considered energy consumption model of the components in datacenter is not practical. For instance, the energy consumption of the cloud router is assumed to have a linear relation with the traffic (energy per bit), which is not the case in reality.

When the datacenter location is fixed, Buysse et al. [21] address how to route the (a set of given) traffic requests by using minimum energy in optical cloud networks. They consider the same energy consumption model for the servers as in this paper and non-bypass IP-over-WDM networks. They also incorporate the energy consumption of supporting hardwares such as coolers, water pumps and power supply system. These hardwares can be switched off if their served servers or racks are turned off. They subsequently propose two heuristics, Fully Anycast (FA), which tries to use the minimum network energy, and Assisted Anycast (AA), which tries to consume minimum datacenter energy.

Dong et al. [22] jointly consider how to route the traffic requests and select the datacenter locations to minimize the total energy consumption in IP-over-WDM networks with datacenters. They also study how to replicate content among datacenters by using the minimum energy. At last, they incorporate renewable energy source and study optimally allocate datacenters so that the total energy consumption is minimized.

Kantarci and Mouftah [23] assume the energy consumption of datacenter as: $E_{DC} = E_{idle} + (E_{busy} - E_{idle}) \cdot u_t$, where $u_t$ represents the datacenter utilization in terms of workload. An IP-over-WDM cloud network is adopted in their model, and they consider 3 kinds of traffic requests as we do in this paper. They subsequently propose both ILPs and heuristics to find energy-aware paths to accommodate the given traffic matrix.

Addis et al. [24] address the problem of minimizing energy consumption in both cloud systems and communication networks, i.e., cloud networks, which is similar to our paper. They propose a Mixed Integer Linear Programming
(MILP) optimization model to solve this problem. They apply Power Usage Effectiveness (PUE) which is a fractional number to model the energy consumption of datacenters. They do not consider optical networks as the transportation networks to connect datacenters. Instead, they assume that the energy consumption of a path is equal to the energy consumption of switched-on routers.

Nevertheless, the above papers only tackle the static energy-aware provisioning problem, where the traffic matrix is known in advance. Also, they do not consider the benefits of traffic grooming to the energy saving in optical cloud networks.

The following articles consider hybrid power networks, where selected datacenters are powered by renewable energy (named by green datacenter) while the rest of the datacenters consume conventional energy (called grown datacenter):

Suppose three link-disjoint paths are precalculated between each node pair, Borylo et al. [25] tackle energy-aware anycast routing problem in WDM networks with datacenters. They propose two straightforward strategies that always first choose a closest green datacenter from the client node in order to maximally reduce carbon footprint. Moreover, they also propose a penalty-based heuristic in making a balance between choosing green or grown datacenters in order to reduce carbon footprint and retaining network resource (e.g., bandwidth). On basis of [25], Borylo et al. [26] improve the proposed strategies when: (1) datacenters have finite computing resources and (2) three kinds of cloud service requests in terms of different energy requirements are assumed. Moreover, they propose a cooperation model between the network operator and cloud service provider, that strikes a balance between exchanging information/content of datacenters and resulted network performance (e.g., blocking probability).

Gattulli et al. [27] consider two different renewable energy source in various hours of a whole day: solar and wind. They propose two anycast routing heuristics to accommodate dynamic connection requests in order to minimize the carbon emissions. Three different IP-over-WDM network architectures are considered in their simulations. The simulation results reveal that the proposed algorithms can achieve up to around 30% carbon reduction compared to a shortest path-based algorithm, paying off a slight increase in terms of blocking probability.

Schondienst and Vokkarane [28] advocate choosing the green datacenters for traffic grooming in order to reduce carbon footprint since the O-E-O conversion at these green nodes generate very less emission. They propose two clustering approaches, namely (1) Distance Clustering: associates every node to their closest green node and designate the green node as a hub, (2) Label Forwarding: clusters the nodes according to the number of demands between the client nodes and datacenter nodes. Via simulations, they found these two clustering approaches can reduce more carbon emissions than no grooming approach (shortest path algorithm).
2.2. Energy-Efficiency in Optical Networks

2.2.1. Energy-Aware Provisioning

Cavdar [29] addresses the dynamic energy-aware traffic provisioning problem at the WDM layer by allocating weights to links and selecting a path with minimum weight. However, the author assumes that the capacity of each traffic request is equal to the maximum capacity of each wavelength, which means that traffic grooming cannot be applied.

Xia et al. [30] discuss the energy and traffic flow details of every operation in an IP-over-WDM network. They subsequently propose an energy-aware routing algorithm that uses an auxiliary graph to represent the consumption of each operation, both at the IP and WDM layers. However, the proposed routing method assume an infinite holding time. Chen and Jaekel [31] do take holding time into account and use an ILP to show that the holding time affects the energy consumption in traffic grooming. However, they do not propose a scalable energy-efficient traffic grooming algorithm for scheduled traffic, i.e., the arrival and holding time of the request are specified and the holding time is finite.

Zhang et al. [32] incorporate holding time in energy-aware traffic grooming for both the static and dynamic case. In the static traffic grooming case, they propose an ILP. For dynamic traffic grooming scenario, they find the shortest paths in an auxiliary graph with specific weights. Their algorithm is compared to two routing algorithms from [33], which are “minimum lightpaths” that tries to minimize the number of newly established lightpaths and “minimum hops” that tries to minimize the number of lightpath hops. Simulation results show that the algorithm in [32] performs best under low traffic, but performs worst under high traffic. The algorithm proposed in this paper uses a more refined energy model than in [32]. For instance, we distinguish between energy consumed in router ports at the IP layer and in the components at the optical layer that consume a fixed energy for each full wavelength connection.

Chen and Jaekel [34] consider three energy consuming components in optical networks, namely (1) IP router (2) OXC and (3) fiber link. They develop an exact ILP for energy-aware dynamic anycast routing problem. However, in their assumed energy consumption model, the network components (e.g., fiber links) can be switched on and off interchangeably (to save energy) based on whether the devices are busy or not, which is not practical today.

Beletsioti [35] investigate the energy-aware routing problem in IP-over-WDM networks for both dynamic and static traffic cases (with infinite holding time). They propose an exact ILP and three heuristics which try to route the requests by using existing lightpaths, since they advocate that traffic grooming can always bring energy saving compared to setting up new lightpaths. However, as we will elaborate in Section 3, traffic grooming is not always helpful regarding saving energy compared to establishing new lightpaths, and this depends on the required bandwidth and existing lightpaths in the network if we assume the power consumption model in Section 3.

In [10], we propose a polynomial-time path selection algorithm for dynamic scheduled lightpaths in IP-over-WDM networks via an auxiliary graph. Traffic
grooming and wavelength conversion are assumed and enabled in our algorithm. In this paper, we extend our work to solve both dynamic and static energy-aware traffic provisioning problem in optical cloud networks.

2.2.2. Energy-Efficient Network Design

Given the traffic matrix is known, the energy-efficient design refers to the case where the resources of a network (e.g., transponders and amplifiers) are unknown, and the aim is to reach the most energy or cost efficient design such that all the traffic requests are satisfied.

Shen and Tucker [36] first express energy consumption based on different components in IP-over-WDM networks, like Erbium Doped Fiber Amplifiers (EDFA), transponders, router ports, etc. Following that, under the constraints of flow conservation, wavelength capacity, and so on, an ILP is proposed for accommodating all the requests such that the total consumed energy is minimized. The authors also propose two heuristics. While this paper suggests that energy efficient design is often also cost effective, Palkopoulou et al. [37] claim that the most cost efficient architecture is not always the least energy consuming one.

Dey and Adhya [38] study how to jointly design an energy-efficient and cost-efficient IP-over-WDM networks. Given a physical topology and a traffic matrix, their considered problem refers to determine the minimum number of transponders, regenerators required at different nodes, amplifiers at different fiber links, routing and wavelength assignment (traffic grooming is enabled) for lightpaths to be set up, and the fiber links to be deployed between node pairs such that the total expenditure (network costs+energy consumption) is minimized. Two Exact ILPs and an auxiliary matrix-based heuristic are proposed to solve this problem in each case when the Maximum Transparent Reach (MTR) is considered as distance and hops.

Ebrahimzadeh et al. [39] consider amplifiers, switches, and transceivers as the main energy consuming devices in optical WDM networks. Suppose that the traffic matrix is known in advance, they propose both an exact ILP and five heuristics to accommodate the requests in an energy-efficient way. The first heuristic is to find a Minimum Spanning Tree (MST) to route the requests, and all the other links could be switched off for saving energy. When a MST cannot accommodate all the requests or the blocking probability is higher than a threshold, the other four proposed heuristics try to switch on minimum number of links (and hence minimum energy) to lower the blocking probability based on different link wake up policy, e.g., the minimum energy consumption link, the link which increases the connectivity of the graph, most congested link and the link which reduces the blocking probability largely.

3. Network Model and Energy Analysis

3.1. Optical Cloud Network Architecture

The overall considered optical cloud network architecture is shown in Fig. 1. We adopt the transparent IP-over-WDM network with optical bypass and grooming capability [40] to transport data. It is a two-layered network architecture,
with routers in the IP layer and optical cross connects (OXC)s in the WDM layer. The following components consume energy:

1. Transponders: Transponders have two functionalities: (1) O/E and E/O conversion between the OXC and IP router and (2) transmitting and receiving signals.
2. OXCs: The optical cross connects optically switch traffic when traffic is bypassed at a node.
3. Router ports: Router ports (residing in the line card) mainly deal with electronic processing.
4. EDFAs: Erbium-doped Fiber Amplifiers deal with amplifying the signal, and the distance between two neighboring EDFAs is at most 80 km.

On top of the IP-over-WDM networks, we assume that the datacenter is located on some core router nodes. The datacenter consists of a cluster of servers and switches, and it is usually structured in a 3-tier architecture. In datacenter networks, servers are hosted by different racks, and within the same rack, servers are connected with a Top of Rack (ToR) switch. These ToR switches are further inter-connected through aggregate switches in a tree-like topology. Similarly, these aggregate switches are further connected with several core switches in the upper layer. Fig. 2 depicts a datacenter network topology. We consider two energy consuming components in datacenter networks, namely, (1) servers and (2) switches.

1. Servers: the servers are composed of a CPU, memory, storage systems, etc. They are responsible for processing the calculating tasks, data storage, etc.
2. Switches: the switches deal with exchanging data among the servers, routing traffic, etc.
Similar to [22, 23], we consider three kinds of traffic requests in this paper, namely, (1) downstream datacenter request, which is from a datacenter to a core router, (2) upstream datacenter traffic, which is from a core router to a datacenter, and (3) regular Internet traffic, which is from one core router to another core router (in core networks).

3.2. Energy Analysis of Traffic Flow in Different Cases

We will discuss the energy model that we adopt in this paper and analyze the energy consumption in the cases of both traffic grooming and setting up a new lightpath.

Often, e.g. in [30, 43, 44], it is assumed that power consumption $P$ has the following linear relationship with traffic:

$$P = P_i + P_t \cdot b$$  \(1\)

where $P_i$ is the overhead which represents the idle power consumption in the unit of Watts, $P_t$ symbolizes the traffic-dependent power factor in the unit of Watts/Gbps, and $b$ represents the amount of traffic in Gbps. We consider different power calculation methods for different operations. Some operations are only wavelength (but not traffic) dependent, in which case $P_t = 0$. For instance, once transponders and OXCs are switched on they will consume a fixed amount of energy corresponding to a full wavelength capacity (and not to the fraction of traffic transported).

$$P = \begin{cases} P_{TR}^{TR} \frac{W}{\lambda} & \forall \text{transponders} \\ P_{OXC}^{OXC} \frac{W}{\lambda} & \forall \text{OXCs} \end{cases}$$  \(2\)
For electronic switching (processing) at the IP layer, the power consumption is traffic-dependent.

\[ P = P^E_S \cdot b \text{ W } \forall \text{ router ports} \quad (3) \]

Although in reality \( P_i \neq 0 \), we have discarded its contribution (i.e., set \( P^E_S = 0 \)), since the core router (currently) cannot be automatically switched on/off, and hence their energy consumption is always fixed\(^1\).

In datacenter networks, the power consumption model is slightly different from the one in IP-over-WDM networks. More specifically, a server’s power consumption is calculated as [21]:

\[ P_{sr} = P_{sr}^i + \frac{P_{sr}^f - P_{sr}^i}{C_{sr}} \cdot b \quad (4) \]

where \( P_{sr}^i \) and \( P_{sr}^f \) represent the power consumption of the server when it is idle and fully loaded, respectively, \( C_{sr} \) denotes its maximum processing capacity or workload and \( b \) stands for its current load \( (b \leq C_{sr}) \). Denote \( P_{sr}^t = \frac{P_{sr}^f - P_{sr}^i}{C_{sr}} \) as the traffic dependent power factor, according to [47]. \( C_{sr} = 1.8 \text{ Gbps}, P_i = 325 \text{ W} \) and \( P_f = 380 \text{ W} \). Therefore, the power consumption of a server is:

\[ P_{sr} = 325 + 30.6 \cdot b \quad (5) \]

where \( 0 \leq b \leq 1.8 \).

The switch consists of line cards together with associated ports. If we assume a single line card switch, then the power consumption of a switch (represented by \( P_{sw} \)) can be calculated as:

\[ P_{sw} = P_{sw}^i + u \cdot P_{sw}^p \quad (6) \]

where \( P_{sw}^i = 760 \text{ W} \) represents the power consumption of a switch when it is on and all its ports are off, \( P_{sw}^p = 5 \text{ W} \) denotes the power consumed by one port and \( u \) indicates the number of switched on ports, where \( 1 \leq u \leq 48 \) according to [42]. In all, Table 1 shows the parameters for calculating the devices’ power consumption that we use in this paper.

In this paper, we use \( E \) to represent the energy consumption of different devices (operations) in the unit of Wh (or kWh), and \( P \) to indicate the power consumption of different devices (operations) in the unit of W (or kW). As a result, for a considered network device \( x \) in this paper, we have \( E_x = P_x \cdot h \), where \( h > 0 \) denotes the time in the unit of hours\(^2\).

\(^1\)Switching off idle line cards/chassis can bring energy saving according to [45, 46] when traffic demand decreases. In this paper, we focus on provisioning optimal paths from the perspective of minimizing the energy consumption of needed switched-on line cards and transponders, so we therefore do not explicitly show their numbers.

\(^2\)When the device working/processing time is less than one hour, for example, 30 minutes, we can convert it to \( h = 0.5 \) hour for the ease of calculating the energy consumption in the unit of Wh (or kWh).
Table 1: Parameters for calculating the devices’ power consumption in optical cloud networks.

<table>
<thead>
<tr>
<th></th>
<th>( P_i ) (Watt)</th>
<th>( P_i ) (Watt/Gbps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transponders</td>
<td>126 per 40 Gbps wavelength [48]</td>
<td>0</td>
</tr>
<tr>
<td>OXCs</td>
<td>2 per 40 Gbps wavelength [49]</td>
<td>0</td>
</tr>
<tr>
<td>Router ports</td>
<td>0</td>
<td>14.5 [40]</td>
</tr>
<tr>
<td>EDFAs</td>
<td>8 [36]</td>
<td>0</td>
</tr>
<tr>
<td>Servers</td>
<td>325 [47]</td>
<td>30.6 (0 ≤ ( b ) ≤ 1.8) [47]</td>
</tr>
<tr>
<td>Switches</td>
<td>( 760 + 5 \cdot u ) (1 ≤ ( u ) ≤ 48) [42]</td>
<td></td>
</tr>
</tbody>
</table>

Figure 3: Example.

Let us consider an example of an optical cloud network in Fig. 3, where all the nodes are assumed to host a datacenter. Suppose two lightpaths (6 → 2 → 3) and (3 → 5) already exist. Lightpath 1 still remains for 5 hours and lightpath 2 remains for 6 hours. The new downstream request \( r(6, 5, b, 3 : 00, 4) \) asks for a lightpath from node 6 to node 5, needing \( b \) Gbps, starting at time 3 : 00 and lasting for 4 hours. We discuss the energy consumption of two routing strategies to route this request: (1) making use of existing lightpaths, i.e. grooming the new request into the existing lightpaths, and (2) setting up a new lightpath, for instance along the shortest-hop path from source to destination. \( b \) is first not assigned with any specific value, and we will show at the end of this section that whether grooming the new request or setting up a new lightpath is more energy-efficient depends on the value of \( b \).

For traffic grooming, the procedure is shown in Fig. 4 [30]. The traffic is first **processed** by the server and **routed** by the switch (PR) in the datacenter, and then it is delivered to the core router at node 6. After that, it is electronically switched (ES) at node 6 and the signal is converted to an optical signal (EO). The signal is then optically switched (OS) and transmitted (TX). After being amplified (AMP) in the fiber, it is optically switched (OS) and amplified to be received by node 3 (RX). After OS, the signal is converted to an electronic signal (OE), and multiplexed to the next connection by electronic switching (ES). There the signal is converted again to an optical signal (EO),
Figure 4: Traffic flow in the grooming case.

and transmitted (TX) to node 5. Node 5 follows the same receiving procedure with node 3. Summing all these energy consumptions of grooming this traffic to lightpath 1 and lightpath 2 will lead to:

\[
E_{groom} = E_{pr} + (E_{es} + E_{eo} + E_{os} + E_{tx} + E_{amp}) + E_{os} + (E_{amp} + E_{rx} + E_{os} + E_{oe} + E_{es} + E_{eo} + E_{os} + E_{tx} + E_{amp}) \\
+ (E_{rx} + E_{os} + E_{oe} + E_{es})
\]

Transponders are responsible for O/E and E/O conversions as well as transmitting/receiving signals. According to the traffic grooming flow shown in Fig. 4 (and also the traffic flow in the new lightpath case in Fig. 5):

- if the traffic experiences the E/O conversion by the transponder, then it will be transmitted by the same transponder;
- if the traffic is received by the transponder, then it will be O/E converted by the same transponder.

That is, when a transponder processes a traffic signal, its energy consumption consists of either (i) the E/O conversion operation and the transmitting operation, where the signal is from the IP layer to the WDM layer or (ii) the O/E conversion operation together with the receiving operation, where the signal is from the WDM layer to the IP layer. In either of these two cases, the transponder operates once to process the traffic signal, and therefore its energy consumption \[^{3}\] is

\[
E_{tr} = E_{eo} + E_{tx} = E_{rx} + E_{oe}.
\]

Considering that traffic grooming makes use of existing lightpaths, the energy consumption (overhead) \[^{4}\] Similar to \[^{40}\], we regard that the E/O (O/E) conversion operation together with the transmitting (receiving) operation consume transponder operating energy only once when the transponder processes a traffic signal from the IP (WDM) layer to the WDM (IP) layer, and we do not consider their individual energy costs in this paper.
of optical switching and transponders is already paid for by the existing lightpath (at least for the time it still remains, after which the cost corresponds to that of setting up a new lightpath for the remaining time) and amplifiers consume constant energy irrespective of traffic, which leads to

\[ E_{groom} = E_{pr} + 3E_{es} \]  

(8)

Figure 5: Traffic flow in the new lightpath case.

The case for directly setting up a new lightpath is shown in Fig. 5 and can be expressed as:

\[ E_{new} = E_{pr} + 2(E_{es} + E_{tr} + E_{os}) + E_{amp} \]  

(9)

For traffic grooming, the IP layer electronic processing will be equal to

\[ E_{es} = P_t^{ES} \cdot b \cdot h \]  

(10)

where \( h \) represents the request’s holding time. Eq. (10) indicates that electronic processing is always traffic dependent and has no energy savings compared to setting up a new lightpath. After the end of the previously allocated lightpaths, the energy costs of traffic grooming equal those of setting up a new lightpath (for the remaining time \( h' \)), where \( E_{es} = P_t^{ES} \cdot b \cdot h' \) and \( E_{os} \) and \( E_{tr} \) behave as \( P_t^{OXC} \cdot h' \) and \( P_t^{TR} \cdot h' \), respectively.

Let us look at some cases in which traffic grooming is not the best solution. For simplicity, let us assume link (6,5) contains one optical amplifier. As a result, the energy consumption of the amplifier on the fiber link (6,5) from 3:00 to 7:00 (the service time of request \( r \)) is \( 8 \cdot 4 = 32 \) Wh. In this sense, \( E_{groom} - E_{new} = E_{es} - 2(E_{tr} + E_{os}) - E_{amp} = P_{es} \cdot 4 - 2 \cdot (126 + 2) \cdot 4 - 32 = (P_{es} - 264) \cdot 4. \) Unlike \( P_{tr} \) and \( P_{os}, P_{es} \) depends on the requested bandwidth \( b \). Hence, if \( P_{es} = 14.5b > 264, \) i.e. \( b > \frac{264}{14.5} \) Gbps, then it is more energy efficient.
to set up a new lightpath than to do grooming. Another example would be where a request would need to be groomed onto many lightpaths, in which case the amount of electronic switching could become too costly. Finally, in some cases it may not be possible to groom traffic all the way from the source to the destination, which in some cases may render a new lightpath a more efficient solution.

4. An Energy-Aware Auxiliary Graph for the Dynamic EPOCN Problem

In this section, we propose an energy-aware routing algorithm for the dynamic EPOCN problem. First, we introduce an auxiliary graph to represent the original topology and then each link in the auxiliary graph will be assigned energy weights, as specified in Section 3. Finally, we apply Dijkstra’s shortest path algorithm on each layer of the auxiliary graph to find the minimum energy consumption path to route the request. We present three types of auxiliary graphs, namely for when the network can offer (1) no wavelength conversion, (2) selective wavelength conversion, and (3) full wavelength conversion.

4.1. Problem Definition

For a certain device $x$, its operational energy consumption \([30]\) refers to the energy consumption of its specific operation that depends on traffic or wavelength (e.g., transponder), without considering its overhead (idle) energy consumption. For example, for a server $sr$ whose energy consumption is $P_{sr} = 325 + 30.6 \cdot b$ according to Eq. (5), the operational energy consumption is equal to $30.6 \cdot b$, where $b$ is the requested rate. Accordingly, the operational energy consumption of switches and amplifiers will be regarded as zero, since they consume a fixed or constant amount of energy irrespective of the presence of the traffic or wavelength.

Formally, the dynamic Energy-aware Provisioning in Optical Cloud Networks (EPOCN) problem is defined as follows:

**Definition 1.** Given a network represented by $G(N, L)$ where $N$ represents the set of $N$ router nodes, $L$ denotes the set of $L$ fiber links with $W$ wavelengths. $N_{DC} \subseteq N$ stands for the set of nodes equipped with a datacenter. Traffic requests $R$ arrive in an online fashion, and for each $r(s, d, b, a, h) \in R$, $s$ and $d$ denote the source and destination, respectively, $b$ indicates the requested bandwidth, and $a$ and $h$ represent the arrival time and holding time, respectively. The dynamic Energy-aware Provisioning in Optical Cloud Networks (EPOCN) problem is to find a path from $s$ to $d$ for each request $r \in R$ such that the total operational energy consumption is minimized.

Considering that turning on servers and switches in datacenters incurs a non-negligible delay when traffic requests arrive in an online fashion, we need to use already switched-on devices in datacenters to accommodate these requests, otherwise the requested services will be degraded. Moreover, the amplifiers in
optical networks cannot be switched on and off interchangeably in reality, so once
they are switched on, they will consume a constant value of energy regardless
of the presence of traffic. The calculation of the number of switched-on/needed
devices is a network design/planning problem, which is out of the scope of this
paper. This is because this paper focuses on provisioning the traffic requests by
allocating the existing devices in an energy-efficient manner. Therefore, when
accommodating traffic requests arriving online, the overhead (idle) energy con-
sumption of servers and switches in datacenters as well as energy consumption
of amplifiers cannot be saved, and we are only interested in the energy con-
sumption involved in accommodating a new request in the dynamic EPOCN
problem. In Section 5 we will address the static EPOCN problem, where all
the traffic requests are known in advance. In this case, these static energy costs
would have to be taken into account, since for instance switching off a subset
of links (with their associated amplifiers) will bring more energy saving. In the
following, via an auxiliary graph, we will present a polynomial-time algorithm
to solve the dynamic EPOCN problem with and without wavelength conversion.

4.2. Auxiliary Graph without Wavelength Conversion

We use $W$ to denote the number of wavelengths that each fiber link contains.
We first assume that there is no wavelength conversion, such that the network
can be regarded as $W$ separate sub-layered graphs, one for each wavelength.
Later, we also include wavelength conversion. There are (at most) three kinds
of nodes in the auxiliary graph, namely (1) a datacenter node, (2) an IP node
and (3) a WDM node. The IP node and WDM node together represent a
network node, and the datacenter node is built on the network node (if any).
As a result, in the auxiliary graph, there are three types of nodes and these
nodes are connected via four types of links:

- Datacenter node: the source and destination node for a downstream and
  upstream traffic request, respectively.
- IP node: the IP source and destination node of a regular traffic request,
  the source node for an upstream traffic request and the destination node
  for a downstream traffic request.
- WDM node: the source and destination for a newly established lightpath
  in the optical layer.
- Datacenter link ($dt_link$): connects the datacenter and the IP nodes.
- Conversion link ($conv_link$): connects the IP and WDM nodes, i.e. O/E
  ($conv_link_{rx}$) or E/O ($conv_link_{tx}$) conversion.
- Optical link ($opt_link$): connects two WDM nodes and could be used to
  establish a new lightpath.
- Lightpath link ($light_link$): connects two IP nodes if they are the start
  and end of an existing lightpath.
For example, in the 3-node network (each node is equipped with one datacenter) of Fig. 6(a) with two existing lightpaths (using the same wavelength $w_x$), the corresponding auxiliary graph on the $w_x$ layer will be represented as in Fig. 6(b). In this way $W$ auxiliary graphs will be created to represent each wavelength of the original graph.

Figure 6: Auxiliary graph for energy-aware path selection with no conversion.

4.3. Weights Allocation

We will discuss how to allocate weights to the 4 different types of links.
4.3.1. **Datacenter Link**

This relates to the case when a signal originates from the datacenter (downstream traffic) to the IP router or terminates at the datacenter (upstream traffic) from the IP router. Recall that $P_{sr}^f$ denotes the power consumption of a fully loaded working server under its maximum workload $C_{sr}$, $P_{sr}^i$ represents the server’s idle power consumption, and $P_{sr}^t = P_{sr}^f - P_{sr}^i$ in Eq. (4). Moreover, let $\Delta$ represent the remaining free workload for servers that have already switched on. Therefore, the energy weight allocation can be calculated as:

$$E_{DC\_link} = \begin{cases} 
(P_{sr}^i \cdot \lceil \frac{b - \Delta}{C_{sr}} \rceil + P_{sr}^t \cdot b) \cdot h & \text{if } b > \Delta \\
P_{sr}^t \cdot b \cdot h & \text{otherwise}
\end{cases} \quad (11)$$

where $\lceil \frac{b - \Delta}{C_{sr}} \rceil$ indicates the number of needed switched-on servers if $b > \Delta$, otherwise there is no need to switch on new servers and the existing servers can serve the request via their remaining workload $\Delta$. As a result, $P_{sr}^i \cdot \lceil \frac{b - \Delta}{C_{sr}} \rceil$ stands for the total idle power consumption for newly switched-on servers (if any), and $P_{sr}^t \cdot b$ represents the total traffic dependent power consumption.

4.3.2. **Conversion and Optical Links**

This case corresponds to the weight allocation when setting up a new lightpath. Note that the link between an IP node and a WDM node represents the energy consumption at the IP layer. Hence, the energy consumption of the conversion link can be calculated as

$$E_{conv\_link\_rx} = E_{tr} + E_{os} \quad (12)$$

$$E_{conv\_link\_tx} = E_{es} + E_{tr} \quad (13)$$

where $E_{tr}$ and $E_{es}$ represent the energy consumption in the case of setting up a new lightpath. The conversion link which originates at a WDM node will be allocated according to Eq. (12), while the conversion link originating at an IP node will be allocated according to Eq. (13). The conversion link originating at the WDM node does not contain $E_{es}$ when its corresponding IP node is not the destination node, because this is included in $E_{conv\_link\_tx}$ when the traffic leaves the IP node. Since traffic does not leave the destination node, it should be added there (as indicated in Section 3). However, from an algorithmic perspective, the $E_{es}$ contributions at the destination nodes are the same for traffic grooming and setting up a new lightpath, and they therefore do not affect the solution. To make the equations simpler, they have therefore been disregarded in the auxiliary graph. The same applies to the lightpath link weight. The weights of the optical link will represent the energy of optical switching, namely $E_{os}$.

4.3.3. **Lightpath Link**

The energy weight for a lightpath link will be set according to two cases, depending on whether the request’s ending time is earlier than that (end) of an
existing lightpath. We use $E_{\text{gnew}}$ to represent setting up a new lightpath for the remaining time.

\[
E_{\text{light,link}} = \begin{cases} 
E_{\text{es}} & \text{if } \text{arrival} + h \leq \text{end} \\
E_{\text{es}} + E_{\text{gnew}} & \text{if } \text{arrival} + h > \text{end}
\end{cases}
\] (14)

where

\[
E_{\text{gnew}} = 2E_{\text{tr}} + E_{\text{es}} + nE_{\text{os}} \text{ during time } (\text{arrival} + h, \text{end})
\] (15)

and arrival denotes the arrival (starting) time of the request, and $n$ denotes the number of nodes (including source and destination) that a lightpath traverses.

4.4. Auxiliary Graph with Wavelength Conversion

Although wavelength conversion consumes additional energy, it may reduce the blocking probability. Wavelength conversion can be easily incorporated into our auxiliary graph by connecting the $W$ independent sub-layers. To do so, we make use of an extra virtual layer on which the path search will start and connect all layers to that virtual layer. Besides using the virtual layer as a starting point, it has the additional advantage that the existing lightpaths with weight $E_{\text{light,link}}$ that were previously (in Section 4.2) present at each sub-layer need now only be present at the virtual layer, which reduces the total number of lightpath links. The IP nodes are connecting to their IP counterpart in the virtual layer by a virtual link ($\text{virt} \text{link}$) of cost $E_{\text{virt,link}} = 0$. If, at some WDM node, one can convert from that wavelength to another, then we add a wavelength link ($\text{wave} \text{link}$) between those two WDM nodes at the two corresponding sub-layers. The cost of that link equals the energy cost of wavelength conversion, which in our case is set to $E_{\text{wave,link}} = E_{\text{tr}}$ since wavelength conversion relies on an O/E/O operation. For example, in Fig. 6(a) there are already two existing lightpaths present in the network. To obtain the auxiliary graph, we first follow the procedure described in Section 4.2, thereby excluding the lightpath links. Subsequently, we add the virtual layer, place the existing lightpath links, connect the IP nodes with their corresponding IP node in the virtual layer, and also add the wavelength links to obtain Figure 7. In the example, nodes $A$ and $B$ are assumed to be able to convert to the other wavelength, and hence a link is drawn between the respective WDM nodes.

The above-described way of connecting the wavelength conversion links is most versatile, since it can also capture the cases where wavelength conversion is only possible from a wavelength to a restricted range of other wavelengths (e.g., see [50]). If this restriction is not there and a node with wavelength conversion capabilities can convert to any other wavelength, we can use the virtual layer to reduce the number of wavelength conversion links. This time we would also need to represent the WDM nodes in the virtual layer. If a node (say $B$) has wavelength conversion ability, then for all the sub-layers a link connecting node $B$ to its virtual companion is added. Instead of $\frac{W(W-1)}{2}$ links per node with wavelength conversion capabilities, $W$ links now suffice (each with half the energy cost of wavelength conversion).
4.5. Algorithm and Complexity

For the arrival and termination of each request we should update the auxiliary graph to reflect the proper weights and available capacity. If there is no wavelength conversion, after allocating weights to the links on each sub-layer of the auxiliary graph, running Dijkstra’s shortest path algorithm on each sub-layer will allow us to choose the most energy-efficient route. In this case, the complexity of the algorithm per request is dominated by running Dijkstra’s algorithm $W$ times, which leads to an overall complexity of $O(WN \log(N) + WN^2)$ where $N$ denotes the number of nodes in the original topology, and $W$ represents the number of wavelengths in a fiber. $N$ is used instead of $L$ (the number of links in the original topology), because in the worst case $N(N-1)$ lightpaths (reflected in links) could be present.

In the general case with wavelength conversion, Dijkstra’s algorithm is run only once - starting from the source node at the virtual layer - but now on a larger graph. This larger (auxiliary) graph consists of $O(WN)$ nodes, and $O(W^2) + O(WL)$ links, where $O(W^2)$ indicates the number of wavelength links and $O(WL)$ denotes the number of all the other links. As a result, this leads to a complexity of $O(WN \log(WN) + WL + W^2 + N^2)$ for each request, since in the worst case $N(N - 1)$ lightpaths (reflected in links) could be present in the virtual layer. The term $W^2$ could be dropped if wavelength conversion goes through the virtual layer.

Figure 7: Auxiliary graph for energy-aware path selection with wavelength conversion.
5. The Static EPOCN Problem

In Section 4, we study the dynamic EPOCN problem, where the traffic requests arrive at the network in an online fashion. The servers and (some) link fibers (with associated amplifiers) cannot be switched off and their energy consumption cannot be saved in the dynamic EPOCN problem, because of lack of knowledge about future requests. In this section, we assume that in the static EPOCN problem the whole traffic request matrix is known in advance, and the goal is to accommodate all the requests by using minimum energy consumption. Formally, the static EPOCN problem is defined as follows:

Definition 2. Given a network represented by $G(N, L)$ where $N$ represents the set of $N$ router nodes, and $L$ denotes the set of $L$ fiber links with $W$ wavelengths. $N_{DC} \subseteq N$ stands for the set of nodes equipped with a datacenter. A set of $|R|$ requests $R$ is assumed to know. For each request $r(s, d, b, a, h) \in R$, $s$ and $d$ denote the source and destination, respectively, $b$ indicates the requested bandwidth, and $a$ and $h$ represent the arrival time and holding time, respectively. The static Energy-aware Provisioning in Optical Cloud Networks (EPOCN) problem is to (i) provision a path from $s$ to $d$ and (ii) allocate server(s) and switch(es) for each traffic request such that the total energy consumption is minimized.

Unlike the dynamic EPOCN problem, the energy consumption in the static EPOCN problem consists of the idle energy consumption of all the devices during the whole network lifetime, and traffic-dependent energy consumption. More specifically, for regular traffic requests which demand a connection from a router node to another router node, the traffic is only transferred within core networks, i.e., the traffic will not be processed in datacenter networks. On the other hand, with respect to upstream or downstream traffic, the signal is not only delivered within core networks, but also processed in datacenter networks, since either the source or the destination is a datacenter node.

Without loss of generality, since the traffic can be transferred at most in both core networks and datacenter networks, we decompose the static EPOCN problem into two subproblems, namely, (1) the Energy-Aware Routing in optical networks problem (EAR), which corresponds to item (i) in definition 2 and (2) the Energy-efficient Server and Switch Allocation (ESSA) problem in datacenter networks, which corresponds to item (ii) in definition 2. The reason to solve the static EPOCN problem separately is two-fold: (1) either the static traffic grooming problem [51] in WDM networks or the server allocation problem in datacenter networks (can be reduced to the bin-packing problem [52]) is NP-hard, which makes the static EPOCN problem ever harder to solve, and (2) although the EAR problem and the ESSA problem are dependent, the energy-consuming devices are different and independent, which means that we can calculate energy consumption in core and datacenter networks separately and add them together as the total energy consumption. Consequently, by jointly solving the EAR problem and the ESSA problem, we can solve the static EPOCN problem.
5.1. Energy-Aware Routing in Optical Networks

According to [51], the static traffic grooming problem is NP-hard, even for the ring networks, and the EAR problem is therefore NP-hard. In the following, we present an Integer Linear Program (ILP) and a modified energy-aware routing algorithm to solve the EAR problem. Although Zhang et al. [32] have proposed an ILP to deal with the energy-efficient time-aware traffic grooming problem, our proposed ILP is different from [32] in terms of: (1) the constraints in our proposed ILP are different from [32], which can more clearly present and distinguish flow conservation in both physical topology and virtual topology, and (2) we use a more refined energy model and incorporate different kinds of network devices (e.g., amplifiers, transponders, OXCs, router ports) to minimize the total energy consumption. We first present our ILP and start with some necessary notation:

\( R(s, d, b, a, h) \): the set of \(|R|\) traffic requests. \( s \) and \( d \) are the source and destination, respectively, \( b \) is the requested bandwidth, and \( a \) and \( h \) denote the arrival time and holding time, respectively.

\( G(N, L) \): physical topology, where \( N \) is the set of \( N \) nodes and \( L \) is the set of \( L \) links.

\( C \): the maximum capacity of each wavelength.

\( W \): the set of \( W \) wavelengths in each fiber.

\( P_{tr} \): power consumption of a transponder per wavelength.

\( P_{os} \): power consumption of a MEMS optical switching per wavelength.

\( P_{es} \): power consumption of electronic switching per Gbps in the IP layer.

\( P_{amp}(m, n) \): the amplifiers’ power consumption of link \((m, n)\).

\( i \) and \( j \) represent the node pairs in the network, i.e., \( i, j \in N : i \neq j \).

\( A_{mn} \): boolean variable indicating whether a fiber link \((m, n)\) is switched on.

\( T \): We decompose the whole network lifetime \( T \) into a set of \(|T|\) discrete time slots. Each time slot \( t \in T \) is the basic time unit (interval) and can span \( \theta \) hour(s), minute(s), or second(s), which depends on the specific application scenario. Each time slot should be present in the unit of hours for ease of energy consumption calculation. For example, if \( \theta \) is equal to one minute, it should be converted into \( \frac{1}{60} \) hour. Consequently, the whole network lifetime consists of in total \(|T|\) slots: \((0, \theta), (\theta, 2\theta), \ldots, ((|T| - 1) \cdot \theta, |T| \cdot \theta)\). As a result, \( T = |T| \cdot \theta \) is in the unit of hours.

Physical topology:

\( P_{mn}^{i,j,w,t} \): number of lightpaths between node \( i \) and \( j \) which traverse link \((m, n)\) on wavelength \( w \) in time slot \( t \).

Virtual topology:

\( V_{ij}^t \): number of lightpaths between node \( i \) and \( j \) in time slot \( t \) in the virtual topology.

\( V_{ij,w}^t \): number of lightpaths between node \( i \) and \( j \) in the virtual topology on wavelength \( w \) in time slot \( t \).

\( \lambda_{ij,r}^t \): boolean variable whether request \( r \) traverses lightpath \((i, j)\) in time slot \( t \).
\( \delta_{ij,t}^{w,h} \): number of lightpaths between \( i \) and \( j \) which traverse through node \( h \) \((h \neq i, j)\) on wavelength \( w \) in time slot \( t \).

Objective:

Minimize

\[
\sum_{t \in T} \sum_{ij} 2 \cdot (P_{tr} + P_{os}) \cdot V_{ij}^t \cdot \theta
\]

\[
+ \sum_{t \in T} \sum_{ij} \sum_{r \in R} P_{es} \cdot r \cdot b \cdot \lambda_{ij}^{r,t} \cdot \theta + \sum_{r \in R} P_{es} \cdot r \cdot b \cdot r \cdot h
\]

\[
+ \sum_{t \in T} \sum_{w \in W} \sum_{ij} \sum_{h \in N \setminus \{i,j\}} \delta_{ij,t}^{w,h} \cdot P_{os} \cdot \theta
\]

\[
+ \sum_{(m,n) \in L} A_{mn} \cdot P_{amp}(m,n) \cdot T
\]  

(16)

Constraints:

Physical topology flow conservation:

\[
\sum_{m \in N} P_{mih}^{ij,w} = \sum_{n \in N} P_{hjn}^{ij,w} \quad \forall i, j \in N: i \neq j, w \in W
\]

\[
h \in N \setminus \{m, n\}, t \in T
\]  

(17)

\[
\sum_{m \in N} P_{mij,w}^{ij,w} = 0 \quad \forall i, j \in N: i \neq j, w \in W, t \in T
\]  

(18)

\[
\sum_{n \in N} P_{nij,w}^{ij,w} = 0 \quad \forall i, j \in N: i \neq j, w \in W, t \in T
\]  

(19)

Equations between the physical topology and virtual topology

\[
\sum_{n \in N} P_{nij,w}^{ij,w} = V_{wij}^w \quad \forall i, j \in N: i \neq j, w \in W, t \in T
\]  

(20)

\[
\sum_{m \in N} P_{mij,w}^{ij,w} = V_{wij}^w \quad \forall i, j \in N: i \neq j, w \in W, t \in T
\]  

(21)

Equations for the virtual topology wavelength

\[
\sum_{w \in W} V_{wij}^w = V_{wij}^t \quad \forall i, j \in N: i \neq j, t \in T
\]  

(22)

Intermediate node flow conservation

\[
\delta_{ij,t}^{w,h} \leq V_{wij}^w \quad \forall i, j, h \in N: i \neq j \neq h,
\]

\[
w \in W, t \in T
\]  

(23)
\[
\sum_{m} P_{m,h,t}^{ij,w} + \sum_{n} P_{h,n,t}^{ij,w} = 2 \cdot \delta_{w,t}^{w,h} \forall w \in W, t \in T, \tag{24}
\]

\[i, j, h \in \mathcal{N} : i \neq j \neq h\]

Virtual topology flow constraints:

\[
\sum_{i \in \mathcal{N}} \lambda_{i,t}^{r} \cdot \delta_{i}^{r} = \sum_{j \in \mathcal{N}} \lambda_{j,t}^{r} \forall r \in R, w \in W \tag{25}
\]

\[
h \in \mathcal{N} \setminus \{i, j\}, t \in T
\]

\[
\sum_{n \in \mathcal{N} \setminus \{r,s\}} \lambda_{r,s,t}^{r} = 0 \forall r \in R, t \in T \tag{26}
\]

\[
\sum_{m \in \mathcal{N} \setminus \{r,d\}} \lambda_{r,d,m,t}^{r} = 0 \forall r \in R, t \in T \tag{27}
\]

Virtual topology capacity constraint:

\[
\sum_{r \in \mathcal{R}} \lambda_{ij,t}^{r} \cdot r_{b} \leq C \cdot V_{ij}^{t} \forall i, j \in \mathcal{N}, t \in T \tag{28}
\]

Calculating whether link \((m, n) \in \mathcal{L}\) is switched on:

\[
A_{mn} = \begin{cases} 
1 & \text{if } \sum_{t \in T} \sum_{r \in \mathcal{R}} \lambda_{mn,t}^{r} > 0 \\
0 & \text{otherwise} 
\end{cases} \forall (m, n) \in \mathcal{L} \tag{29}
\]

Eq. (16) is the goal of the ILP which is to minimize the total energy consumption for each network equipment (operation). More specifically, \(\sum_{t \in T} \sum_{ij} 2 \cdot (P_{tr} + P_{os}) \cdot V_{ij}^{t} \cdot \theta\) calculates the energy consumption of the transponder and OXC in the end nodes, since there are always two end nodes. \(\sum_{t \in T} \sum_{ij} \sum_{r \in \mathcal{R}} P_{es} \cdot r_{b} \cdot \lambda_{ij,t}^{r} \cdot \theta\) calculates the energy consumption of electric switching in the starting nodes and the intermediate nodes, and \(\sum_{r \in \mathcal{R}} P_{es} \cdot r_{b} \cdot r_{h} \cdot \lambda_{ij,t}^{r} \cdot \theta\) calculates the energy consumption of electric switching in the end nodes. \(\sum_{t \in T} \sum_{w \in \mathcal{W}} \sum_{ij} \sum_{h \in \mathcal{N} \setminus \{i,j\}} \delta_{ij,t}^{w,h} \cdot P_{os} \cdot \theta\) calculates the energy consumption of optical switching in intermediate nodes. \(\sum_{(m,n) \in \mathcal{L}} A_{mn} \cdot P_{amp}(m, n) \cdot T\) calculates the switched-on link fibers’ energy consumption.

Eq. (17)-Eq. (19) are the multicommodity equations that account for the routing of a lightpath from its source to its destination [53]. Eq. (17) ensures that for an intermediate node \(h\) of lightpath \((i, j)\) on wavelength \(w\) the number of incoming lightpath streams is equal to the number of outgoing lightpaths. Eq. (18) ensures that for the origin node \(i\) of lightpath \((i, j)\) on wavelength \(w\),
the number of incoming lightpaths is 0. Eq. (19) ensures that for the termination node $j$ of lightpath $(i, j)$ on wavelength $w$, the number of outgoing lightpaths is 0.

Eq. (20) ensures that for the original node $i$ of lightpath $(i, j)$ on wavelength $w$, the number of outgoing lightpaths is equal to the total number of lightpaths between node pair $(i, j)$ on wavelength $w$. Eq. (21) ensures for the termination node $j$ of lightpath $(i, j)$ on wavelength $w$ the number of incoming lightpaths is equal to the total number of lightpaths between node pair $(i, j)$ on wavelength $w$. Eq. (22) ensures that the number of lightpaths between node pair $(i, j)$ should be equal to the number of lightpaths between node pair $(i, j)$ using all the wavelengths.

Eq. (23) and Eq. (24) are the intermediate node flow conservation. Eq. (23) ensures that the number of lightpaths going through an intermediate node using a particular wavelength is equal to or less than the total number of lightpaths between the two ends using that wavelength. Eq. (24) ensures that if a lightpath goes through an intermediate node, it will also go through two neighboring fiber links of the intermediate node.

Eq. (25)-Eq. (27) are the virtual topology network flow conservation, which make sure that the traffic grooming can happen. Eq. (28) represents the virtual topology capacity constraint which makes sure that the aggregate traffic flowing through lightpaths cannot exceed the total wavelength capacity. Eq. (29) calculates which fiber link should be switched on.

We could also slightly modify the proposed energy-aware routing algorithm in Section 4 as a heuristic to solve the EAR problem. More specifically: for each link $l$ in the original network, the weight allocation for its respective optical link in the auxiliary graph is $E_{os} + \alpha_l \cdot E_{amp}$, where $E_{os}$ and $E_{amp}$ denote the energy consumption of optical switching and amplifiers, and $\alpha_l$ is an indicator. Initially, $\alpha$ is set to 1 for all the links $l \in L$. Whenever one link $l$ is switched on for accommodating a request, we let $\alpha_l = 0$. By doing this, we try to route the requests by using a fraction of links in the network, so that the energy consumption of the amplifiers of the switched-off links can be saved.

The modified energy-aware routing algorithm treats the static EAR problem as a dynamic EPOCN problem, but it does not consider the data processing procedure in datacenter networks. It accommodates given traffic requests one by one based on their arrival time. In case more than one request has the same arrival time, the requests are served based on their default sequence given in the input.

5.2. Energy-Efficient Server and Switch Allocation in Datacenters

The Energy-efficient Server and Switch Allocation (ESSA) problem happens for upstream and/or downstream traffic, where data needs to be processed by the server and routed by the switch in the datacenter. Recall that in Section 3 according to Eq. (5), each server should work under its maximum workload $C_{sr}$, otherwise the performance (e.g., speed) of the overloaded server is degraded. When a data request with requested rate $b$ needs to be processed, we need to allocate just enough servers such that the requested rate $b$ is satisfied and each
allocated server does not exceed its maximum workload. For example, if \( b = 6 \) and \( C_{sr} = 1.8 \), then we need to allocate \( \lceil \frac{6}{1.8} \rceil = 4 \) servers to accommodate it. Clearly, allocating more than 4 servers will bring unnecessary idle server energy consumption in this example. Moreover, once a server is switched on, it can be switched off when it is idle (its accommodated request has departed) for saving more energy. But (re)switching on a server will incur a fixed delay of \( \gamma \). In this sense, when a traffic request arrives at time \( t \), the server should be triggered on at no later than time \( t - \gamma \), otherwise it cannot be switched off.

Similarly, we also need to allocate just enough ToR switches to connect those allocated servers. Since we are not confined to any specified datacenter topology (e.g., BCube, Fat-Tree, etc.), we only take into account the energy consumption of needed ToR switches and assume that all the aggregate and core switches are kept on. On the other hand, it is assumed that the ToR switch cannot be switched off once it is switched on.

**Theorem 1.** The ESSA problem is NP-hard.

**Proof 1.** Let’s first introduce the NP-hard Bin-Packing problem [52]: Given \( n \) items with sizes \( e_1, e_2, \ldots, e_n \), and a set of \( m \) bins with capacity \( c_1, c_2, \ldots, c_m \), the Bin-Packing problem is to pack all the items into a minimized number of bins without violating the bin capacity size. For simplicity, assume there is one datacenter and we only consider the server allocation problem. Besides, all the requests are assumed to have the same arrival and holding time. Now, if we regard items as requests and item size as the requested rate, then the ESSA problem is equivalent to the Bin-Packing problem, which is also NP-hard.

In the following, we propose an ILP and a heuristic to solve the ESSA problem. We first propose an ILP and start with some necessary notation.

- \( P_{sr} \): the server’s idle power consumption and traffic dependent power consumption factor, respectively.
- \( P_{sw} \): the switch’s idle power consumption and power consumption of one port, respectively.
- \( U \): the maximum number of ports in a single line card switch.
- \( C_{sr,e} \): the maximum workload of server \( e \) in the datacenter \( dc \).
- \( E \): the set of \( |E| \) servers.
- \( Z_{r,t}^{e} \): float variable representing how much rate server \( e \) in the datacenter \( dc \) has assigned for request \( r \) on time slot \( t \).
- \( X_{dc,e}^{t} \): boolean variable whether the server \( e \) in the datacenter \( dc \) is opened on time slot \( t \).
- \( Y_{dc} \): the number of allocated (opened) servers in the datacenter \( dc \).

**Objective:**

\[ \text{minimize} \sum_{e \in E} \sum_{dc} Y_{dc} \]

\[ \text{subject to} \]

\[ \sum_{e \in E} Z_{r,t}^{e} X_{dc,e}^{t} \leq Y_{dc} \]

\[ \sum_{t} X_{dc,e}^{t} \leq Y_{dc} \]

\[ X_{dc,e}^{t} \in \{0,1\} \]

\[ Y_{dc} \geq 0 \]

\[ Y_{dc} \in \mathbb{Z} \]

We only consider the requests in cases when \( t \geq \gamma \) in this paper.
\[
\min \sum_{dc \in DC, e \in E, t \in T} P_{sr}^i \cdot X_{dc,e}^t \cdot \theta + \sum_{r \in R} r \cdot P_{sr}^i \cdot r \cdot \gamma
\]
\[
+ \sum_{dc \in DC} \left[ \frac{Y_{dc}}{U} \right] \cdot P_{sw}^i \cdot T + \sum_{dc \in DC} Y_{dc} \cdot P_{sw}^p \cdot T
\]

Constraints:

\[
\sum_{r \in R} Z_{dc,e}^{r,t} \geq r \cdot b, \forall r \in R, t \in [r \cdot a, r \cdot a + r \cdot \gamma],
\]
\[
dc \in DC : dc = r \cdot s || dc = r \cdot d, e \in E
\]
\[
\sum_{r \in R} Z_{dc,e}^{r,t} \leq C_{sr}^{dc,e}, \forall t \in T, dc \in DC, e \in E
\]
\[
X_{dc,e}^t = \begin{cases} 
1 & \sum_{r \in R : r \cdot s = dc | r \cdot d = dc} Z_{dc,e}^{r,t} > 0 \| \sum_{r \in R : r \cdot s = dc | r \cdot d = dc} Z_{dc,e}^{r,t+\gamma} > 0 \\
0 & \text{otherwise}
\end{cases}
\]
\[
\forall dc \in DC, e \in E, t \in T
\]
\[
Z_{dc,e}^{r,t} \geq 0 \forall dc \in DC, e \in E, r \in R, t \in T
\]
\[
Y_{dc} = \sum_{e \in E} \max_{t \in T} X_{dc,e}^t \forall dc \in DC
\]

In the objective function Eq. (30), \( \sum_{dc \in DC, e \in E, t \in T} P_{sr}^i \cdot X_{dc,e}^t \cdot \theta \) denotes the overhead (idle) energy consumption of all the active servers, \( \sum_{r \in R} r \cdot b \cdot P_{sr}^i \cdot r \cdot \gamma \) represents the traffic dependent energy consumption for serving all the requests, \( \sum_{dc \in DC} \left[ \frac{Y_{dc}}{U} \right] \cdot P_{sw}^i \cdot T \) calculates the idle energy consumption of the needed switches, and \( \sum_{dc \in DC} Y_{dc} \cdot P_{sw}^p \cdot T \) calculates the energy consumption of the switch ports. Eq. (31) makes sure that each request’s requested bandwidth is accommodated by allocating sufficient servers. Eq. (32) makes sure that the actual workload of each switched-on server does not exceed its maximum workload. Eqs. (33)-(35) calculate the allocated (switched-on) servers and switches for each datacenter. More specifically, Eq. (33) sets \( X_{dc,e}^t \) to be 1 if on time slot \( t \) there exists request \( r \) that is being processed by server \( e \) in the datacenter \( dc \), or on time slot \( t+\gamma \) server \( e \) in \( dc \) is working for some request \( r \). The latter case ensures that we can only switch off a server at time slot \( t \) when it needs to be switched on after \( t+\gamma \), considering that switching on a server induces a delay of \( \gamma \). Eq. (35) calculates the number of allocated servers in each datacenter.
Next, we propose a heuristic, called First Fit (FF) to solve the ESSA problem. The general idea of FF is the following: we first sort all the upstream and downstream requests in $R$ based on their arrival time in an increasing order into the list $R'$. When multiple requests arrive simultaneously, a request with a shorter holding time is put in the front of $R'$. In case that the traffic requests have the same arrival time and holding time, we order them in $R'$ according to their sequence in $R$.

Initially, all the servers are free and switched off (not allocated) in each datacenter $dc \in DC$. For each request $r(s, d, b, a, h) \in R'$ (from the front to the back), we first search the servers (denoted by $S_w$) in the datacenter $d$ (if $r$ is an upstream request) or $s$ (if $r$ is a downstream request) that are currently working but whose maximum workload has not been reached. Denote $\Delta$ as the total remaining free workload of the servers in $S_w$.

- If $\Delta \geq b$, then we allocate a subset of servers in $S_w$ whose total free workload is no less than $b$. To that end, we first set $\beta = b$. Foreach server $s$ in $S_w$, if $\beta > 0$, then we use $s$ to provision $r$ via its free workload (denoted by $\Delta_s$) and let $\beta = \beta - \Delta_s$. This procedure stops when $\beta < 0$, which means that we allocate enough servers to issue $r$.

- If $\Delta < b$, we first serve $r$ with $\Delta$ by using all the servers in $S_w$. Moreover, we additionally allocate and switch on $\lceil \frac{b-\Delta}{C_{sr}} \rceil$ servers to accommodate $r$, such that the requested rate $b$ can be satisfied.

Whenever an allocated server is idle, i.e., all its served requests have departed, we regard it as “switched off” immediately in order to save energy. In order to reduce the number of needed ToR switches, when $\Delta < b$, apart from using all the servers in $S_w$ to offer $\Delta$ rates, we always first choose to allocate ever turned-on (working) but now “switched off” servers to satisfy the remaining $b-\Delta$ requested rates. However, if a server needs to serve a request $r$ at time $t$, then it should be kept switched on within time slots $[t-\gamma, t]$, considering that (re)switching on a server incurs a delay of $\gamma$. In this context, if an allocated server is idle within time slots $[t-\gamma, t]$ but it is necessary to serve request(s) on time $t$, then the server’s idle energy consumption (i.e., $P_{sr} \cdot \gamma$) during these time slots in $[t-\gamma, t]$ should be counted (and this server cannot be switched off). Consequently, this procedure continues for each request in $R'$ until all the requests in $R'$ have been served. In the end, if $S_{dc}$ different servers has been allocated in datacenter $dc$, then the number of needed ToR switches in datacenter $dc$ is $\lceil \frac{S_{dc}}{U} \rceil$, where $U$ is the maximum number of ports in a single card switch.

6. Simulations

6.1. Simulation Setup

We simulate two realistic carrier backbone networks [30], namely the NSFNet of 14 nodes and 20 links and the USANet of 24 nodes and 43 links (see Figures 8 and 9). We assume that the nodes with the red circle are core nodes.
with a datacenter, i.e., they are the source or the destination for downstream or upstream traffic request. We vary the amount of traffic requests from 100 to 500, where $s$ and $d$ are randomly generated according to the uniform distribution. For each set of traffic requests, the arrival time always begins with time 0 and it increases by 1 hour for every 50 traffic requests. The holding time is uniformly distributed in $[1, 5]$ in the unit of hours, leading to a total 9 (latest arrival time) + 5 (longest possible holding time) = 14 time slots (with each time slot spanning one hour) for maximum 500 traffic requests. The regular traffic, upstream traffic and downstream traffic each takes up one third of the total traffic requests. The requested capacity in these three kinds of traffic requests varies in $[5, 10]$, $[1, 5]$ and $[15, 20]$ in Gbps, respectively. In order to have a better observation of the results during the different time slots, for a set of traffic requests $R$, we always allow the arrival time of its regular traffic to be earlier than for the upstream traffic, and the upstream traffic always has a arrival time earlier than the downstream traffic. That is, the regular traffic always arrives first, followed by the upstream traffic, and the downstream traffic arrives last within the same traffic matrix. The capacity of each wavelength is 40 Gbps, and the number of wavelengths per link is chosen as 80. We use Cplex to implement...
the ILPs and C# to implement the routing algorithms and FF heuristic. The gap of ILPs is set to 0.02.

6.2. Dynamic Energy-Aware Provisioning in Optical Cloud Networks

![Graph](image)

(a) NSFNet (W=80)
(b) USANet (W=80)

Figure 10: Energy consumption in the dynamic EPOCN problem.

We compare our energy-aware routing algorithm proposed in Section 4, in terms of energy consumption, to both the method of directly setting up new lightpaths (Direct New Lightpath) based on the shortest path and to a traffic grooming algorithm. The traffic grooming algorithm first selects paths on which the request can be completely groomed. That is, the existing lightpath(s) can form a path to route the request from the source to the destination with sufficient wavelength capacity. If such multiple paths exist, the one using the lowest number of lightpaths is chosen. If no such path(s) exists, then a new (shortest-hop) lightpath will be set up. If these two options fail, the traffic grooming algorithm selects a path that partly grooms onto the existing lightpath and partly sets up one (or more) new lightpath(s). For example, suppose there is an existing lightpath \( a - b \), and we want to find a path from \( a \) to \( c \). Suppose that it is not possible to set up a new lightpath from \( a \) to \( c \) directly (because of for instance the lack of wavelength resource), then the traffic grooming algorithm first grooms this request on lightpath \( a - b \) and then sets up a new lightpath from \( b \) to \( c \). Otherwise, the request is blocked.

Since all the downstream or upstream traffic requests need to be processed in the datacenter, those energy costs caused by the datacenter are the same for aforementioned 3 algorithms. We therefore eliminate the datacenter’s energy consumption (i.e., \( E_{pr} \) in Section 3) for aforementioned 3 algorithms both from an algorithmic aspect and due to comparison reasons. Figs. 10(a) and 10(b) give the returned energy consumption value by employing the three algorithms for the dynamic EPOCN problem in the NSFNet and the USANet, respectively. From the figures we can see that the energy consumption grows almost linearly with the amount of traffic, but the slope is smallest for energy-aware routing. It is worthwhile to mention that, due to the above simulation
setup, no request is blocked for any routing algorithm. In this sense, the energy consumption comparison among those three algorithms is fair. In order not to clutter the figure, we omitted the results for wavelength conversion, since, with ample wavelengths, wavelength conversion was not needed and thus never used (giving the same results as for without wavelength conversion).

6.3. Static Energy-Aware Provisioning in Optical Cloud Networks

For each downstream or upstream traffic request \( r(s, d, b, a, h) \in R \), we divide it into (regard it as) two kinds of requests, namely (1) routing requests: \( r_o(s_o, d_o, b, a, h) \in R_o \), where \( s_o \) or \( d_o \) indicates the core router node in the network layer if there is a datacenter built on it, otherwise we let \( s_o = s \) or \( d_o = d \); (2) datacenter processing requests: \( r_p(x, b, a, h) \in R_p \), where \( x = s \) for downstream traffic and \( x = d \) for upstream traffic. The regular traffic request \( R_r \subset R \) keeps the same. Therefore, \( R_o \cup R_r \) serve the inputs of the EAR problem, and \( R_o \) is used for the ESSA problem. Moreover, we set \( \gamma = 1 \) for the ESSA problem, which is the incurred delay due to (re)switching on a server. Consequently, taking the sum of the results of the EAR problem and the ESSA problem returns the result of the static EPOCN problem.

6.3.1. EAR problem

We compare our proposed ILP with the modified energy-aware routing algorithm (in Section 5.1), and traffic grooming and Direct New Lightpath (in Section 6.2). Similar to the modified energy-aware routing algorithm, traffic grooming and Direct New Lightpath algorithms do not consider the data processing procedure in datacenter networks. They accommodate given traffic requests one by one based on their arrival time. If more than one request has the same arrival time, the requests are accommodated by these algorithms based on their default sequence given in the input.

Figs. 11(a) and 11(b) present the energy consumption value returned by the four algorithms to solve the EAR problem for two networks, respectively. Similar to Section 6.2, the energy consumption values in Fig. 11 grow almost linearly with the number of traffic requests. We see that the ILP has the smallest slope which indicates that it is the most energy-efficient, followed by energy-aware routing and traffic grooming, and the direct new lightpath heuristic results in the highest energy consumption value. Moreover, because of the contribution of the amplifiers’ energy consumption, we observe that the energy consumption in Fig. 11 is larger than the energy consumption in Fig. 10 returned by the same routing algorithm in the same network.

We have generated five sets of traffic requests with the amount of requests: 100, 200, 300, 400 and 500. In order to compare the total energy consumption, we consider all of the 1500 requests from all of the 5 sets. Although in reality the delay of switching on a server is less than one hour, since one hour is the basic time slot in the simulation, we round this delay value up to one hour.

Fig. 12 depicts the
total energy consumption for all these 1500 traffic requests over different time slots in two networks. There are more traffic requests from time slots 2 to 8, leading to a higher energy consumption during these time slots. It can be seen that more energy is saved by the ILP during these time slots compared to other time slots. Finally, Fig. 13 illustrates the total energy consumption for all of the 1500 traffic requests for different devices. We see that the router (electrical switching) consumes the highest energy consumption, followed by the transponder and amplifier, and OXC consumes very little energy compared to them. Nevertheless, the ILP can always return the minimum energy consumption value for different devices.

---

7This is caused by the traffic generation in Section 6.1 since the requests in each traffic set start to arrive at time 0.
6.3.2. ESSA problem

Regarding the ESSA problem, we compare the ILP with the First Fit (FF) allocation heuristic. Figs. 14(a) and 14(b) present the energy consumption value returned by the ILP and FF for the ESSA problem in two networks, respectively. The ILP is more energy-efficient in comparison to FF, since it is the exact solution. FF, on the other hand, can achieve a close performance.

Similar to Section 6.3.1, in the following we show the energy consumption over different time slots and per device for a total of 1500 requests. Figs. 15(a) and 15(b) show the total energy consumption for all of the 1500 requests over different time slots. We see that the energy consumption around the middle time slots (time slot 8) is the highest, and it decreases to (close to) 0 at time slots 1 and 14. This is because, as we mentioned in the beginning, the upstream and downstream traffic arrive later than the regular traffic in $R$, so at time slot
Figure 15: Total energy consumption over the time slots for all of the 1500 traffic requests in the ESSA problem.

Figure 16: Total energy consumption of servers and switches for all of the 1500 traffic requests in the ESSA problem.

1, there is no upstream or downstream traffic. The upstream or downstream traffic after time slot 12 can only happen when $|R| \geq 400$, which also leads to a smaller energy consumption. Fig. 16 reveals the total energy consumption of the servers and switches for all of the 1500 requests. It can be seen that the servers consume more energy than switches. Nevertheless, we see that the ILP always yields a better performance in terms of energy consumption than FF either in each time slot in Fig. 15 or for different devices (servers and switches) in Fig. 16. Finally, Fig. 17 plots the total energy consumption for all of the 1500 requests in the static EPOCN problem, i.e., the sum of energy values of Fig. 11 and Fig. 14. We observe that the ILPs are more energy-efficient compared to the heuristics.
6.3.3. Running Time

Finally, we present the running times of the ILPs and their counterparts in Fig. 18 and Tables 2 and 3 for the EAR and ESSA problems. In both cases, the running time of the ILP is significantly higher than the running time of the heuristic(s) and grows exponentially with the number of traffic requests. In particular, we see that the running time of 3 heuristics in Fig. 18 grows almost linearly for the EAR problem and that it remains very low for FF as shown in Tables 2 and 3 for the ESSA problem. Nevertheless, considering that the exact ILPs can bring more energy saving compared to the heuristics in Figs. 11 and 14, ILPs are promising to use when the running time is not a big concern.

Figure 17: Total energy consumption for all of the 1500 traffic requests in the static EPOCN problem.

Figure 18: Running times of algorithms for the EAR problem in two networks.
Table 2: Running times of the ILP and FF for the ESSA problem in NSFNet (ms).

<table>
<thead>
<tr>
<th>Algo.</th>
<th>Traffic</th>
<th>100</th>
<th>200</th>
<th>300</th>
<th>400</th>
<th>500</th>
</tr>
</thead>
<tbody>
<tr>
<td>ILP</td>
<td></td>
<td>343000</td>
<td>1075340</td>
<td>1548270</td>
<td>2641980</td>
<td>7980411</td>
</tr>
<tr>
<td>FF</td>
<td>≤10</td>
<td>≤10</td>
<td>≤10</td>
<td>≤10</td>
<td>≤10</td>
<td>≤10</td>
</tr>
</tbody>
</table>

Table 3: Running times of the ILP and FF for the ESSA problem in USANet (ms).

<table>
<thead>
<tr>
<th>Algo.</th>
<th>Traffic</th>
<th>100</th>
<th>200</th>
<th>300</th>
<th>400</th>
<th>500</th>
</tr>
</thead>
<tbody>
<tr>
<td>ILP</td>
<td></td>
<td>853231</td>
<td>1186170</td>
<td>2503352</td>
<td>5355971</td>
<td>9064365</td>
</tr>
<tr>
<td>FF</td>
<td>≤10</td>
<td>≤10</td>
<td>≤10</td>
<td>≤10</td>
<td>≤10</td>
<td>≤10</td>
</tr>
</tbody>
</table>

7. Conclusion

In this paper, we have studied the Energy-aware Provisioning in Optical Cloud Networks (EPOCN) problem for both dynamic and static cases. When traffic requests arrive in an online fashion, by applying an energy model to compute the energy consumption at the datacenter, IP and optical layer, the proposed energy-aware routing algorithm can attain practical energy consumption weights. Our model also allows the energy costs of full or sparse wavelength conversion to be taken into account. Simulation results show that the proposed energy-aware routing algorithm can always result in a better performance in terms of energy consumption compared to directly setting up new lightpaths or traffic grooming in an energy-oblivious way. Because wavelength conversion costs energy, it is only used when it is really needed to prevent a request from being blocked and otherwise it prefers wavelength continuous routes. When the whole traffic matrix is known in advance, the static EPOCN problem turns into NP-hard. We further divide this problem into two NP-hard (sub)problems, namely, (1) the Energy-Aware Routing (EAR) problem in optical networks and (2) the Energy-efficient Server and Switch Allocation (ESSA) problem. We present an ILP and a heuristic for each of them. The simulation results reveal that the ILPs can achieve more energy saving in comparison to heuristics, but this comes at the expense of a higher running time.

Acknowledgement

This work has been partially sponsored by EU FP7 ITN CleanSky (No. 607584). Part of this work has been done by Song Yang and Fernando Kuipers with the title “energy-aware path selection for scheduled lightpaths in IP-over-WDM networks”, and it appears in the 18th IEEE Symposium on Communications and Vehicular Technology in the Benelux (SCVT), pp.1-6, 2011. We would like to acknowledge the support and contribution of Fernando Kuipers on that part of this work.


[53] B. Mukherjee, Optical WDM Networks (Optical Networks), Springer Heidelberg, 2006.