Euclides Carlos Pinto Neto

A STATE-AWARE METRIC-BASED VSDN ALLOCATION METHOD IN MULTI-TENANT SDN SCENARIOS TO OPTIMISE POWER CONSUMPTION

Recife

2016

Euclides Carlos Pinto Neto

A STATE-AWARE METRIC-BASED VSDN ALLOCATION METHOD IN MULTI-TENANT SDN SCENARIOS TO OPTIMISE POWER CONSUMPTION

Undergraduate monograph presented to the program of Bachelor's of Computer Science of the Department of Statistics and Informatics of Federal Rural University of Pernambuco as partial requirement to obtaining of Bachelor's of Computer Science degree.

Federal Rural University of Pernambuco – UFRPE

Department of Statistics and Informatics – DEINFO

Bachelor's of Computer Science

Supervisor: Gustavo Rau de Almeida Callou

Recife

2016

Euclides Carlos Pinto Neto
A STATE-AWARE METRIC-BASED VSDN ALLOCATION METHOD IN MULTI-TENANT SDN SCENARIOS TO OPTIMISE POWER CONSUMPTION/ Euclides Carlos Pinto Neto. – Recife, 2016-
68 p. : il. (algumas color.) ; 30 cm.

Supervisor: Gustavo Rau de Almeida Callou

Undergraduate Monograph – Federal Rural University of Pernambuco – UFRPE
Department of Statistics and Informatics – DEINFO
Bachelor's of Computer Science, 2016.

1. Power Consumption. 2. Optimisation. 3. Software-defined Networking. I. Gustavo Rau de Almeida Callou II. Universidade Federal Rural de Pernambuco (UFRPE). III. Departamento de Estatística e Informática (DEINFO). IV. A State-aware Metric-based vSDN Allocation Method in Multi-tenant SDN scenarios to Optimise Power Consumption

Euclides Carlos Pinto Neto

A STATE-AWARE METRIC-BASED VSDN ALLOCATION METHOD IN MULTI-TENANT SDN SCENARIOS TO OPTIMISE POWER CONSUMPTION

Undergraduate monograph presented to the program of Bachelor's of Computer Science of the Department of Statistics and Informatics of Federal Rural University of Pernambuco as partial requirement to obtaining of Bachelor's of Computer Science degree.

Trabalho aprovado. Recife, 28 de dezembro de 2016:

Gustavo Rau de Almeida Callou
Orientador

Fernando Antônio Aires Lins
Convidado 1

Bruno Costa e Silva Nogueira
Convidado 2

Recife
2016

This work is dedicated to my father and hero Elizer Carlos Pinto, in memoriam, and my family, a God's gift.

Acknowledgements

The main thanks are directed:

- to God, our father and creator, who does wonders in my life;
- to Jesus Christ, who is the way, the truth and the life and no one comes to the Father except through him;
- to the Holy Spirit, who brings us peace, joy, and liberation;
- to Mary, who prays for us sinners now and at the hour of our death;
- to my mother (Odete), my brother (Eli), my stepfather (Sérgio) and my girlfriend (Aline).
- to my family and friends, who are always with me;
- To Professor Gustavo Callou, who has helped and motivated me throughout the work and to whom I have a great admiration and respect.
- To Federal Rural University of Pernambuco (UFRPE) for all support since the beginning of my course;
- To Department of Statistics and Informatics (DEINFO) for the infrastructure provided;
- To Athlone Institute of Technology (AIT), Software Research Institute (SRI) and Dr. Brian Lee for the opportunities that was offered to me;
- To professors Rafael Mello, Pablo Sampaio, Obionor Nóbrega, George Cabral, Julian Menezes, Jeísa Domingues, Fernando Aires, Marcelo Marinho, Filipe Cordeiro and Douglas Véras, who have contributed directly or indirectly to the production of this work;
- To all professors of DEINFO, who have helped me to learn and to be a better student;
- To all friends from UFRPE.

Abstract

In the past few decades, Information and Communication Technology (ICT) and the Internet had a rapid development. New services and technologies have appeared to afford new demands. In this context, software-defined networking (SDN) is introduced as a technology to offer a better network control and programmability to tenants, enabling innovation in business applications. Furthermore, software-defined networks can be divided in many different and independent virtual networks, with different controllers, called virtual software-defined networks (vSDNs). Each subnetwork can be given to different tenants independently, who has software-defined environment and does not need to worry about the complexity of the topology management and operation optimisation, in a process called vSDN allocation. However, this process presents issues such resilience, i.e., recovering when specific devices stop to work, and power consumption optimisation of topologies with idle devices.

In this work a state-aware metric-based vSDN allocation method to optimise power consumption by turning off idle devices is proposed. This method considers multiple tenants and allocates vSDNs based on metrics of the network topology (e.g. power consumption, power usage effectiveness (PUE), delay, packet loss, availability and reliability). Additionally, the Network Model Flow (NMF) is proposed to represent the data flow and the network. In order to optimise the power consumption, sleep modes are adopted in idle devices. In the experiments, the proposed strategy were able to reduce the power consumption upper than 50%.

Keywords: Power Consumption. Optimisation. Software-defined Networking.

Resumo

Nas últimas décadas, a Tecnologia da Informação e Comunicação (ICT) e a Internet tiveram um rápido desenvolvimento com novos serviços e tecnologias para atender uma crescente demanda. Nesse contexto, as redes definidas por software (SDN) surgiram como uma tecnologia que oferece maior controle e programabilidade das redes para os clientes, tornando possível inovação em aplicações de negócio. Além disso, uma rede definida por software pode ser dividida em diferentes e independentes redes virtuais, com controladores distintos, chamadas redes virtuais definidas por software (vSDNs). Essas redes podem ser entregues a diferentes clientes, que possuem um ambiente definido por *software* e não precisam se preocupar com a complexidade do gerenciamento da rede e otimização em sua operação, em um processo chamado de alocação de vSDN. Entretanto, esse apresenta alguns problemas como a resiliência (recuperação) e otimização de consumo energético.

Neste trabalho um método de alocação de vSDNs baseado em métricas e consciente do estado da rede para otimização de consumo energético pela desativação de dispositivos ociosos é proposto. Este método considera múltiplos clientes, e aloca as vSDNs baseando-se em métricas da rede (como consumo de energia, eficácia do uso de energia (PUE), *delay*, perda de pacotes, disponibilidade e confiabilidade). Além disso, o Modelo de Fluxo de Rede (NFM) é proposto para representação do fluxo de dados e da rede. Para a otimização do consumo energético, dispositivos ociosos são desativados. Nos nossos experimentos, nossa abordagem teve um bom desempenho chegando a atingir uma redução maior que 50% no consumo energético.

Keywords: Consumo energético. Otimização. Redes definidas por software

List of Figures

List of Tables

List of abbreviations and acronyms

A	Availability
DCN	Datacenter Network
GUI	Graphical User Interface
IaaS	Infrastructure as a Service
ICT	Information and Communication Technology
IoT	Internet of Things
ISP	Internet service provider
IT	Information Technology
NFM	NFM Network Flow Model
PaaS	Platform as a Service
PUE	Power Usage Effectiveness
QoS	Quality of service
SaaS	Software as a Service
SDN	Software-defined Networking
TTF	Time to Failure
VoIP	Voice over IP
VPNs	Virtual Private Networks
vSDN	Virtual Software-defined Networking

Contents

1 Introduction

In the past few decades, Information and Communication Technology (ICT) had a rapid development (HE et al., 2015). Nowadays, ICT is one pillar of the society and provides impacts in professional and personal live as well as drives the economic growth (SHRUTHI, 2014). With the advent of the Internet, the whole world became connected and the communication became much more efficient. ICT have been largely employed by companies worldwide with applications in many sectors, such as business sector, political, organisations (private and public) and education (POHRADSKý; LONDáK; ČAčíKOVá, 2010) (NWIZEGE et al., 2011). ICT can be considered as the study of technology used to handle information that aims to improve communication (NWIZEGE et al., 2011).

In the last years an enormous growth in Internet technology and its insertion in services and applications were observed as well as new technologies such as Mobile Computing, Internet of Things (IoT), Big Data were developed (PAUL; ZHONG; BOSE, 2015). These applications, services and technologies exploit the communications capabilities of the Internet in order to address problems. For instance, distributed processing exploits communication capabilities in order to share data among remote nodes. All these technologies rely on communication and networking, and the connections demand is growing.

By 2020, the annual global IP traffic is expected to reach 2.3 zettabytes, which corresponds to around 194 exabytes per month; in rush hours the Internet traffic will be 4.6 times larger in comparison to 2015(CISCO, 2014). In addition, the number of devices connected to IP networks worldwide will be three times higher than the global population in 2020. Finally, the global IP traffic will be 100 times larger from 2005 to 2020. This amount of traffic demands reliable and redundant backbones, which are increasingly robust and available. Thus, this drives to a higher power consumption and operational cost.

Energy consumption has been plenty increased thanks to globalisation and industrialisation (KHOSRAVANI et al., 2016), so that the worldwide reduction of energy consumption and greenhouse gas emissions are urgent challenges (CHOI; LEE, 2016). Green economy is attracting a lot of investments and is considered an important agent to reduce energy consumption (BONETTO; MELLIA; MEO, 2012). Moreover, ICT importance may also be increased specially with the development of new technologies such as cloud computing and Internet of Things (IoT) (CISCO, 2014).

Datacenters represent a quarter or more of total IT costs for large enterprises (MCKINSEYCOMPANY, 2008). As networking plays an important role in this scenarios, many expenses comes from it. The same report presents that the amount of energy consumed by datacenters around the world in 2008 is twice as many as the energy consumed between 2000 and 2006. In 2008, the average datacenter used to consume as much energy as 25.000 households. Note that a cost is related to all of this consumption, and an increase on the energy consumed also rises the associated cost.

A reduction on the energy consumption is a key aspect for many companies due to economical, environmental and marketing reasons. In addition, datacenters and networking infrastructure demands high-performance machines and involves unnecessary energy consumption (BIANZINO et al., 2012a). Many companies such as Amazon, IBM, Google, Yahoo! and Microsoft have their own Cloud platforms, which exploits the computer networking capabilities, consuming enormous amounts of power. Thus, this may lead to environmental drawbacks such as CO_2 emissions and global warming (HASAN et al., 2014). To couple with these drawbacks, solutions may exploit software/hardware implementations, green algorithms, modelling and virtualization alternatives.

An important factor for the following years is the usage of renewable energy. As gas emissions have become worldwide problem in different sectors, business professionals are looking for different energy sources and using renewable energy sources to reduce environmental impacts (JIN; SHENG; GHOSH, 2013). Due to the fact that non-renewable energy source have a greater environmental impact, strategies such as renewable energy allocation, in which decisions on mechanism of storing, selling or using renewable energy are made based on different parameters such as price variation, grid-load and carbon release (HASSAN et al., 2015), may be adopted.

In these scenarios, wired access networks represent critical scenario for reducing the carbon footprint of telecommunications infrastructures. They are employed considering a large set of network energy requirements, and network redundancy with green support may contribute to the energy consumption reduction compared to traditional network redundancy (BOLLA et al., 2011b). Note that this green support strategy may exploit different techniques, such as sleep-mode states.

1.1 Motivation

Many research institutes and projects are focused on sustainability and energy consumption reduction as well as new networking models. One of the main problems is to address the current demands without compromising the future in terms of energy sources and global warming. Furthermore, all the considered demands play an important role in network infrastructures that support services considering the Internet of Things and cloud computing and, in the next years, the number of data traffic will increase dramatically.

The entire network infrastructure demands a considerable amount of energy. As the global data traffic will increase dramatically by 2020 (CISCO, 2014), the energetic management is critical. Due to the fact that increasing the data sent also rises energy consumption and cost, companies are concerned about the energetic and cost requirements of new services. The growth of new technologies such as Cloud computing, Fog computing and Internet of Things drives this energy consumption to a much more expensive infrastructure.

This data growth is a result of a process in which many devices get connected to Internet and start sharing information with each other (DIXIT; KUMAR, 2015) and, in order to support this growth, a large distributed system is required. An increasing number of system developers use cloud technologies to provide IoT services (ZENG; WATSON, 2015). The advance of parallel computing, distributed computing and grid computing enabled the constructions of a new computing model called cloud computing, which aims to share data, processing, and services transparently among many users (WANG; HUANG, 2015). Thus, this large distributed system as well as the communication demands solutions to reduce the operational cost and energy consumption.

Furthermore, the new Internet models, that are being developed by researcher, need to address issues that the current architecture cannot deal with, such as power consumption optimisation. In this context, this work offers a method to optimise the energy usage in a software-defined networking exploiting virtual topologies and multiple tenants.

1.2 Objectives

The main goal of this research is to propose a method to reduce power consumption of physical network scenarios in software-defined networks that employs virtual topologies and multi-tenancy considering performance metrics and exploiting sleep-modes. As many networks hypervisors work with virtual topologies (which are built on physical topologies) and offer these topologies to multiple tenants, this work aims to reduce the network energy consumption without compromising the virtual networks' performance. To be more specific, the goals of this work are:

1. To propose an interface to collect the tenants requirements. In order to build a suitable model to represent the tenants requirements, quantity of tenants and their specific needs must to be considered.
2. To propose a workflow model that represents networks. In order to accomplish this, an abstract representation of networks using graphs will be proposed considering the studied metrics, i.e., availability, performance and energetic metrics.
3. To propose an algorithm to consider a set of metrics (e.g. performance, power consumption and availability) into graph's nodes. The graph's nodes represent each peace of equipment used in the IT infrastructure adopted for supporting the Cloud

and/or Fog computing system as well as network topologies. This algorithm is able to change during the execution time the metrics values.

4. To propose an algorithm that will be able to select the best path for that data to flow through it. This selection will enable the maximisation of the tenants' desired metrics (e.g. availability) and, further, the reduction of power consumption of the network by turning off the nodes that are not used. This work assumes a path as a set of nodes and edges that connects a specific node, called source, to another specific node, called target.
5. To implement an optimisation algorithm to reduce power consumption. This algorithm will be able to selectively switch nodes to cold standby mode considering the best paths in terms of selected metrics (e.g., performance, availability and power consumption).
6. To integrate the proposed model with the algorithms to reduce the desired metrics.

1.3 Contribution

The expected contribution is a formal model to represent energy consumption, operational cost, mean delay, mean packet loss, availability and reliability of software-defined network infrastructures. Furthermore, an optimisation process is also proposed in order to optimise the power consumption and cost considering the presented metrics (e.g. power consumption, operational cost, mean delay, mean packet loss, availability and reliability) requirements.

Along with the expected results of this work, the proposed goals are important for academic and industry communities. In academia, we have published the results in conferences. In IT companies, by reducing the electrical operation cost of networks infrastructure, we expect to contribute in three different ways:

1. Powering up the companies' profit
2. Reducing the environmental impact
3. Turning the companies greener

This benefits can be reached by employing the proposed approach to the network infrastructure operation planning.

1.4 Outline

This work is organised as follows:

In Chapter 2, we discuss related works and the intersections among them. In Chapter 3, we present basic concepts on computer network infrastructure, green IT and green networking and metrics. Then, we present the proposed models for energy consumption evaluation in Chapter 4. Chapter 5 shows the case studies considering the proposed model and optimisation process. Finally, Chapter 6 presents conclusions as well as future directions of this research.

2 Related works

Many contributions in networks power saving area have been developed worldwide. Some focus on virtualization, whereas others focus on full sleep modes for devices and links. In this section a list of solutions that are related to the proposed strategy is presented.

In (DUAN; ZHAN; HOHNERLEIN, 2015) a strategy for predicting idle intervals of processing components is presented in order to estimate the most effective way to reduce costs and energy consumption in real-time considering sleep states. The authors focus on datacenters, and show that CPUs are the most energy consumer components in terms of operational cost and energy consumption. Therefore, this paper focus only on CPUs. The experiments showed that, compared to Dynamic Voltage and Frequency Scaling (DVFS), approach based on low consumption states, an economy of 10%-50% of energy consumption is achieved in some cases. However the authors do not apply it in different environments, and do not consider parameters such as availability.

Feizi et al. (FEIZI; ZHANG; MEDARD, 2013) employ flow networks to address many challenges in cloud computing, such as to consider communication and computation requirements in the network. Two computation costs are considered, the linear computation cost model and the maximum computation cost model. The authors also show that their strategy can be used as a load balancer over different nodes, which can also be used for cloud networks design. Although the authors use similar abstraction tools that is adopted in this research, i.e., those based on flow networks, and consider different metrics in the several issues presented, the solutions are only applied to cloud computing and does not consider specific energetic constraints.

In (YANG; LEE; ZOMAYA, 2016) the authors propose an approach to datacenter network planning based on energy efficiency. The main goal is to put related VMs into the same server in order to reduce energy consumption in the transmissions. This solution also balance the load and avoid congestion by putting the VMs near to each other. The experiments, made on NS-2, considered the end-to-end delay of data packets, throughput, ratio of dropped data packets and energy consumption as metrics. The proposed solution is also compared to two different datacenter Network (DCN) management algorithms: the Global First Fit and the ElasticTree. Finally, it presented a lower delay and packet dropping ratio, because it reduces the distance between VMs in the network. A higher transmission capacity is also reached, because the entire network is less used in the DCN. Consequently, a lower energy consumption is reached. However, this paper does not exploit the cold standby modes in non-used node. Thus, metrics such as nodes availability and nodes energetic performance are not considered in the DCN planning.

In (FERREIRA; CALLOU; MACIEL, 2013), a power load distribution algorithm (PLDA) is presented to optimise energy distribution of datacenter power infrastructures. This approach optimises the flow distribution of energy flow models (EFM) (CALLOU et al., 2012). EFMs estimate sustainability and cost of datacenter infrastructures respecting the power capacity each device can provide (power system or source) or extract (cooling or sink) and the PLDA aims to improve energy distribution allocating optimised values to edge weights of the EFMs. In the experiments many metrics were considered, such as availability, and operation costs. A reduction on power consumption up to 15.5% was achieved. However, this paper does not consider the sleep or cold standby modes. Furthermore, the approach is based on Ford-Fulkerson algorithm (FORD; FULKERSON, 1962) and is focused on flow capacity, but does not consider metrics such as devices performance in the distribution.

In (ATTIAH et al., 2016), the authors propose an approach to support dynamic traffic load considering low energy consumption requirements. This approach is applied in sensor nodes and it is based on sleep and wake up modes. It aims to balance the conflicting goals of power saving and latency. This adaptive energy saving approach wireless sensors can effectively extend the network lifetime with low latency. The process of adjusting sleep time for each node dynamically plays an important role. The results showed that this approach achieves a significant gain in energy saving and energy efficiency, as well as good performance for latency over different scenarios. However, this work does not consider metrics of path in the graphs, such as best path in terms of energy consumption.

In (BOLLA et al., 2011a) an approach to implement and support standby modes in network infrastructure is developed. The authors develop a strategy to reconfigure nodes and links based on incoming traffic considering operational requirements, such as reliability and stability. This approach also exploits the network protocol stack and link virtualization. As it depends on the traffic, the nodes and links are reconfigured through virtualization. The routers line cards are considered the minimum block to be put to sleep, and this process is made in real-time when these line cards are idle or the load they have is small and can be sent to other line cards. The solution relies on finding best paths considering specific metrics in the network. This approach best performs in backbones with many links, considering redundancy. However, this solution is developed only for network backbones and the solution is not applied to different environments, such as cloud computing, without specific adaptations. Furthermore, energetic metrics such as energetic efficiency are not taken into account in the algorithm. Finally, the metrics cannot be specified in the processes of selecting the nodes to be put in sleep mode.

In (CHIARAVIGLIO; MELLIA; NERI, 2009) a real backbone is used as well as real traffic profile. An evaluation of the energy cost is made as well as a study about turning off idle nodes to achieve a more economic environment. The authors show that it is possible to achieve more than 23% of energy saving per year using their simple algorithm, which is based on flow networks. However, this paper does not focus in specific metrics such as availability and energy efficiency. Thus, these constraints are not used to manage the sleeping process.

The authors in (CHIARAVIGLIO; MELLIA; NERI, 2012) face the problem of minimising power consumption for Internet service provider (ISP) networks. Evaluation strategies are provided to keep network traffic on few networks resources and the paper's main goal is to turn off network devices (and links) respecting the connectivity requirements as well as link utilisation. The authors present a heuristic approach, in which many constraints are used to turn devices off, such as power consumption and flow. The algorithms are tested on real and not-real environments. By using this approach, the power savings can be larger than 35%. However, some important criteria are not used, such as availability.

Corigliano et al. (CORIGLIANO; TRUNFIO, 2014) present an analysis on how the sleep-and-wake energy-saving approach can reduce energy consumption. This study is made on Gnutella peer-to-peer file sharing network. The strategy is based on putting nodes (lead-peers) in the sleep state, and the device decides its own sleep duration. Five strategies are presented, and each one uses different metrics. Firstly, VAR_HR computes the sleep duration using the hit rate (number of query hits in a period of time). Secondly, VAR_FS sets the sleep interval based on the number of files shared. Then, VAR_QR considers the query rate to conduct the analysis. Finally, FIX_1WD and FIX_3WD are based on the wake periods. The results showed that the total energy consumption (TEC) is reduced. In some cases, this reduction is greater than half of TEC, for example adopting the VAR_QR the TEC is 42%. Although this paper considers specific metrics of the system, such as hit rate, some important metrics related to each node are not considered, such as energetic efficiency metrics. Furthermore, this approach delegates the decision of waking up to the nodes instead of making this decision in a centralised sleeping control system.

In (CHIARAVIGLIO et al., 2015) the authors show that, in an IP backbone using sleeping strategies, the sleep mode tends to increase the links lifetime. On the other hand, many transitions between sleep and wake modes can decrease its lifetime. The authors present a solution based on time slots to perform the decisions and compare it with two energy-aware sorting criteria algorithms: the least-flow (LF), which sorts the nodes set by the number of active links, and the most-power (MP), which sorts the nodes set by the power each node consumes (CHIARAVIGLIO; MELLIA; NERI, 2012). This solution allows the devices to be put in the sleep mode, which results in a higher lifetime performance than the other two approaches. The paper considers lifetime performance as the goal and showed that many transitions between states (sleep-wake) can be harmful to the devices lifetime. However, this approach can lead to higher energy consumption. Furthermore, metrics such as availability are not considered and the study is not applied to different environments, such as vSDN.

The authors in (OKONOR et al., 2014) propose an approach to achieve energy efficiency in ISP backbones based on traffic conditions, in which the main goal is to put the maximum number of links in sleep mode in off-peak time but respecting the system demand. As long as the traffic increases, the links are put in a wake state, so that they can handle the traffic. The results showed that this approach reached up to 47% and 44% of energy gains without any performance loss. Although the solution presents a good energy save ratio, it is based on the links. Furthermore, the selection of links to be put in sleep mode does not consider metrics such as availability and the solution is only designed for ISP networks.

Francois et al. (FRANCOIS et al., 2012) proposes an approach to reduce network topology by turning off links during network operations. The authors employ a greedy algorithm to smartly remove network links respecting the network demand and not causing congestion. An algorithm is also applied to determine the time that the network topology is reduced. The heuristic proposed in this paper is composed of three stages which basically computes the Traffic Matrix (TM), chooses the links to be removed and executes the off-peak window determination. The simulation results showed that a saving of 18.6% in energy consumption is achieved without significant performance impacts. Furthermore, this approach can also be deployed on legacy routing platforms. However, this paper only focuses on turning off links, and does not consider cold standby states for devices. Although the impacts in the network in terms of performance (delay) are evaluated, important network metrics such as availability are not considered.

In (AMOROSI et al., 2015) the authors consider the problem of devices lifetime and relate it to the energy consumption considering sleep modes. According to the authors, the sleep mode tends to increase the device's lifetime, but the transition between wake and sleep modes tends to decrease it. A realistic case study was conducted and the results showed that the devices lifetime can be increased by 43% compared to an approach without sleep modes. However, this work is developed to address reliability, i.e., lifetime problems. Furthermore, it does not consider different metrics such as network performance. Finally, it does not consider cold standby modes.

Abdul-Qawy et al. (ABDUL-QAWY; POTLURI, 2015) propose a hybrid green networking heuristic that exploits the parameters considered in topology-oriented (ESTOP) and traffic-oriented (ESTA) solutions for energy-efficiency. The "weighted Betweenness, Flow, Links, Power" (w-BFLP) and the "weighted Betweenness, Flow, Power" (w-BFP) heuristics are presented. The w-BLFP sorts the nodes and the w-BFP sorts the edges, both in terms of node/edge importance in the topology. The goal is to switch nodes and edges off respecting the bandwidth demand constraints. The authors also compare the proposed solution to ESTOP and ESTA. The experiments showed that the proposed approach presents a better performance than ESTOP and ESTA in terms of specific metrics such as power gain, mean utilisation of edge, percentage pf sleeping edges. However, although the algorithm to sort nodes is based on many metrics, such as the betweenness, total traffic flowing and degree of nodes, other important metrics such as power consumption and availability are not considered.

The authors in (CHIARAVIGLIO et al., 2013a) aim to study the impact of sleep modes on the backbones devices lifetime. A model that integrates sleep modes and lifetime is developed, as well as a model to compute the average network lifetime. The results showed that sleep modes increase the average network lifetime whereas constant changes between sleep and wake modes may be harmful to the network lifetime. Furthermore, topology plays an important role in lifetime, considering that a highly connected topology tends to increase the network lifetime. The authors focus on studying the impacts of sleep modes in devices lifetime, but does not consider metrics such as devices availability in the process of turning devices off.

In (CHIARAVIGLIO et al., 2013b), the authors deal with the issue of energy saving techniques application. The work aims to propose a strategy to minimise the impact of these applications on the performance of Internet Service Provider networks by limiting the number of network configurations. A study of putting in sleep mode links of a backbone network is made, which limits the number of times that each device changes its power state. A model to compute to compute energy saving considers parameters such as QoS requirements is presented. The results showed that the energy savings, considering limited configurations, are similar the energy savings of strategy that do not use configuration limits, i.e., in a practical implementation a limited number of configurations can be applied with similar energetic performance. However, although this solution considers QoS, combinations and priorities of metrics such as availability and power consumption are not considered.

In (MINERAUD et al., 2016), the authors deals with the problem of minimising fossil fuel consumption in Internet Service Provider (ISP) networks. It uses a gradient-based routing protocol, which dive a higher priority in forwarding packets to routers powered by the highest quantity of renewable energies. Also, the protocol can switch routers to sleep mode in order to optimise energy consumption. The performance evaluation using real data showed that the solution presents a reduction of fossil fuel usage by the network (>70%), keeping the routing performance similar as when no energy optimisation is used. However, this paper does not consider metrics such as availability and performance in the devices selection.

In (BOLLA et al., 2015), the authors face the problem of energy waste of consumer electronics. They employ the Network Connectivity Proxy (NCP), which exploits the usage of low-power states in idle devices and execute networking background routines to Home Gateways (HG). The HG plays the role of Control Point (CP) whereas the devices play the role of Controlled Device (CD). The paper describe the software architecture needed in CP and CD, which is made of logical elements. Finally, this work reports performance evaluation, made on a prototype, which has shown that a single HG can perform well with few dozen devices concurrently. It leads to substantial saves in the energy consumption and lower operational costs. However, this paper does not consider paths in the network topology and metrics such as availability and energy in order to sort devices priority.

Ansari et al. (ANSARI; HAN; TAHERI, 2016) present an overview on the optimisation of energy-efficient broadband access networks. The design of these networks is also exploit, i.e., the energy-efficient design of passive optical networks. It also presents a discussion on enabling and emerging technologies for enhancing the energy efficiency, such as Massive MIMO, CoMP, SCNs, and D2D communications. The authors discuss about classifications of green wireless access networks based on power supplies, such as off-grid green wireless access networks, on-grid green wireless access networks, and hybrid green wireless access networks. Finally, open research issues are presented. Although this work overviews open issues and technologies, it does not deals with the optimisation of energy consumption. Furthermore, strategies that considers combinations of metrics are not exploit.

In (WANG et al., 2016), the authors present a Software-defined Networking (SDN) based application to manage sleep modes in wireless sensors networks (WSNs). This application corresponds to a sleep scheduling algorithm that manages the energy of the network and it aims to employ SDN architecture instead of traditional WSN architecture. The computation is started in the sensors but finished in the controller, in order to achieve a better network management. Basically, the nodes send data based on the controller decisions. This controller is usually connected to power supplies. Finally, the controller only transmits decision packet to nodes that change the sleep status or next-hop nodes. The results showed that this algorithm presents advantage in energy management, especially in network lifetime, number of alive nodes and number of solo nodes. However, this project is applied to WSNs and does not consider metrics such as availability and energy. Furthermore, priority is also not considered.

In (GOMES et al., 2016a), a strategy to allocate vSDN considering resilience issues is presented aiming to improve the system operation by improving aspects such as resource usage and service delivery. An algorithm for VSDN allocation that considers resilience factors is presented. Thus, it considers QoS parameters defined in the SLA to deploy the VSDN. The experiments, which are conducted in a real network topology, outperform the existing algorithms, including higher connectivity. However, this work does not focus on energy aspects of vSDN allocation. Thus, combined metrics, such as availability and power consumption together, are not considered.

In (GOMES et al., 2016b), two algorithms to allocate vSDN are proposed. The first algorithm, called the Bandwidth and Reliability According to Redundancy (BRAR), aims to achieve resilience. The second one, called Bandwidth and Energy Efficiency Focus (BEE-Focus), aims to find path considering bandwidth availability and energy efficiency. The experiments showed that the BEE-Focus algorithm outperforms existing algorithm by increasing the energy efficiency and enabling more vSDN deployments and that BRAR algorithm is suitable to construct reliable VSDNs providing a certain level of connectivity. However, this work does not focus on reduce power consumption of the whole network infrastructure. Thus, this work does not consider sleep modes.

Gomes et al. (GOMES et al., 2016c) proposes two algorithms to allocate vSDNx. These algorithms combine reliability, bandwitch and energy efficiency. Also, this approach considers the current state of the network. The first algorith,called Relative Disjoint Paths (RDP), focus on achieving reliability whereas the second one, called State-Aware of Bandwidth and Energy Efficiency (SA-BEE), defines paths that connect desired nodes reducing power consumption. The experiments show that the proposed algorithms achieve better results comparing to existing approaches by increasing energy efficiency and respecting the SLA when facing failures. However, this work does not employ sleep modes to reduce power consumption.

Table 1 shows a summary of the related works previously presented in relation to the adopted metrics. These works addresses the power consumption problem in many ways, considering different metrics and different technologies. However, none of those relates availability, performance and energetic metrics in the optimisation of power consumption, operational costs and device lifetime. In order to address the energetic problem in networks, this work proposes a standby-mode flow-based approach to reduce operation costs and energy consumption in networks. This efforts aims to produce a model of Network Standby Unit (NSU), a centralised component of networks that decides which devices will be put in standby modes, based on flow networks. This decision is made considering the best path. In our definition, the best path is the most suitable path, in the network topology graph, considering specific metrics. Different from the previous works, this research proposes an integrated approach that combines energy consumption, performance and dependability metrics. Thus, these metrics can also have different weights, which means that one metric can be considered more important than others in the best path calculation. Note that this model can be applied in different scenarios, such as Internet Service Providers (ISPs) backbones and Datacenter Networks (DCN). Then, we sort the paths using the metrics and considering the connection constraints. Finally, devices with less importance to the topology are switched to cold standby mode and remains in this state until the network

Related Work	Focussed on SDN	Ener.	Perf.	Dep.
(DUAN; ZHAN; HOHNERLEIN, 2015)		X	X	
(FEIZI; ZHANG; MEDARD, 2013)			X	X
(YANG; LEE; ZOMAYA, 2016)		X		
(FERREIRA; CALLOU; MACIEL, 2013)		X		X
(ATTIAH et al., 2016)		X	X	
(BOLLA et al., 2011a)			X	X
(CHIARAVIGLIO; MELLIA; NERI, 2009)		X		
(CHIARAVIGLIO; MELLIA; NERI, 2012)			X	
(CORIGLIANO; TRUNFIO, 2014)			X	
(CHIARAVIGLIO et al., 2015)		X		
(OKONOR et al., 2014)			X	
(FRANCOIS et al., 2012)		X	X	
(AMOROSI et al., 2015)		X		X
(ABDUL-QAWY; POTLURI, 2015)			X	
(CHIARAVIGLIO et al., 2013a)			X	
(CHIARAVIGLIO et al., 2013b)			X	
(MINERAUD et al., 2016)		X		
(BOLLA et al., 2015)			X	
(ANSARI; HAN; TAHERI, 2016)		X		
(WANG et al., 2016)	X		X	
(GOMES et al., 2016a)	X		X	
(GOMES et al., 2016b)	X	X		X
(GOMES et al., 2016c)	X	X		X

Table 1 – Characteristics of the related works in terms of application (SDN) and energetic (Ener.), performance (Perf.) and dependability (Dep.) metrics.

demand increases and a need for a higher network capacity.

3 Background

This chapter presents the background information needed for a better understanding about this work. Firstly, Computers Network Infrastructure is presented. Secondly, a discussion on Green IT and Green Networking is made. Finally, important metrics in computers networking are shown, such as energy consumption, renewable energy usage, delay, packet loss, availability and reliability.

3.1 Computer Network Infrastructure

In the past years computers and computers networks have been extensively employed. New products and services have been offered by vendors and new demands have appeared. With the advent of the Internet, these new technologies could see a quick popularisation of its capabilities and reach a broad public. A computer network, in which Internet is based on, is “a collection of autonomous computers interconnected by a single technology “, and are used in applications such as (TANENBAUM; WETHERALL, 2011):

1. Business Applications: These applications are very diverse and are focused on resource and information sharing. Technologies such as VPNs (Virtual Private Networks), Voice over IP (VoIP) and powerful computers called servers may also be employed to address employees and companies’ needs.
2. Home Applications: These applications are focused on connectivity. Many services that may be employed such as instant messaging, social network and IPTV (IP TeleVision) are based on computer networks. Furthermore, new technologies such as smart home, smart cities and ubiquitous computing demands a much more abundant communication among devices.

An important application of computers networks are Cloud and Fog computing. Cloud Computing is a widely used computing paradigm, which can be defined as:

- "A Cloud is a type of parallel and distributed system consisting of a collection of interconnected and virtualised computers that are dynamically provisioned and presented as one or more unified computing resources based on service-level agreements established through negotiation between the service provider and consumers" (BUYYA; VENUGOPAL, 2008).

Many of the current clouds are built on top of modern data centers. Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) are provided such as utilities, so the end users pay for usage (TSAI; BALASOORIYA, 2010b). However, Cloud Computing faces many issues such as data fragmentation and utilising SaaS applications in conjunction with multiples applications (TSAI; BALASOORIYA, 2010a)

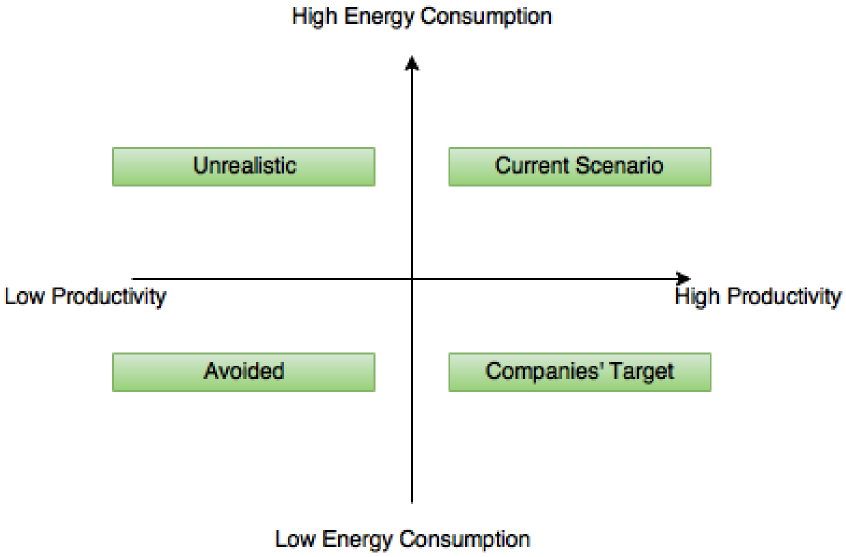
Fog Computing (BONOMI F., 2012) is a highly virtualized platform that provides Cloud Services between end devices and traditional Clouds, generally located at the edge of network. It provides several benefits (BONOMI F., 2012), such as edge location, location awareness, and low latency, which makes it suitable for IoT environments; geographical distribution; very large number of nodes; real-time interactions; and support for mobility. The Fog aims to bring the cloud to the edge, with the same services but in a smaller environment.

3.2 Green Networking

Nowadays, the success of the Internet is huge but environmental impacts of it, which had been overlooked until recently, comes from the significant amount of energy used and the greenhouse gas emission, which represents 2% of the total world's emission (BECKMANN; JAUCO; KOO, 2014).

Companies worldwide are always looking for energy reduction, which is important for economical, environmental and marketing matters (BIANZINO et al., 2012a). The biggest problem is that energy consumption may lead the company to a lower production level, which means lower profitability as presented in Fig. 1.

Figure 1 – Energy consumption and productivity relationship.



Source: The author

In companies that energy consumption is highly related to profitability and more specifically production, although there is a seek for a sustainable setting, managers may not prioritise sustainability in order to keep high productivity. However, IT infrastructures demand high-performance machines and involves unnecessary energy consumption (BIANZINO et al., 2012a) that can be optimised, i.e., the energy consumption can be reduced without impacting the profits.

The environmental problems related to Greenhouse gases has increased in recent years considering their implications on climate changes and considering that ICT represents an important agent in this reduction. In addiction, there are innovation opportunities in making network devices and protocols aware of the energy they consume (BIANZINO et al., 2012b)

Other point is that Green IT in general is an ideal way for companies for being in the green direction because IT is continually being refreshed and virtual server and virtual data storage technology are methods that allow companies to reduce equipment (LAMB, 2011). However, although the same principles are followed, networks present specific and general techniques to improve the sustainability and keep the operation at the same level.

3.2.1 Green Networking Techniques

The techniques to turn networks greener can be divided based on what level of the network they occur (BECKMANN; JAUCO; KOO, 2014). The authors consider two different approaches: Software- and Middleware-level and Hardware-level approaches.

The Software- and Middleware-level approaches present solutions related to a higher level of abstraction, i.e., software-based solutions, which are:

1. Virtualization: "is the common process of dividing up the resources of a server, operating system, network connection, etc. in order to give the illusion to network applications that there are multiple, independently working version of the divided object" (BECKMANN; JAUCO; KOO, 2014).
2. Resource Consolidation and Middleware: "resource consolidation is the efficient usage of computer server resources in order to reduce the total number of active machines required" (BECKMANN; JAUCO; KOO, 2014), i.e., is a technique to manage the infrastructure to deliver what is needed without wasting resources.
3. Green TCP: "energy-aware applications are applications that can detect their own behaviours in order to predict periods when they will be idle for long periods of time" (BECKMANN; JAUCO; KOO, 2014). This technique may employ, for example, software-defined networking (AHMAD et al., 2015).

On the other hand, Hardware-level approaches present solutions in a lower level of abstraction, i.e., hardware-based solutions where implementations in devices components may be required. They are:

1. Interface Proxying: is similar to Green-TCP because it sets machines to sleep mode, but Green-TCP only reduces energy consumption in cases where an entire client machine is permitted to sleep. In many applications a particular process or user on a machine can be idle, and that machine may still need to keep awake to receive background network traffic. Basically, it redistributes traffic in the network to reduce energy consumption. For instance, in (BOLLA et al., 2015) the authors employ a Network connectivity Proxy (NCP) to exploit the usage of low-power states in idle devices and execute networking background routines to Home Gateways (HG).
2. Adaptive Link Rate: is the ability to adapt the links to work in lower states, i.e., with a lower data rate. This helps to save energy and do not present problems that changing link to sleep mode may present, such as power spike overhead.
3. Power over Ethernet Improvements: Power over Ethernet (PoE) devices are very energy inefficient, they allow connected machines to be powered via a twisted pair in the Ethernet cable. This method is appropriate and has led to the widespread use of PoE switches, but the downside is that they use more power than regular devices and it is a result of the losses that occur in Ethernet cables. Improvements in this technology may lead the infrastructure to a substantial reduction in its energy consumption.

3.3 SDN

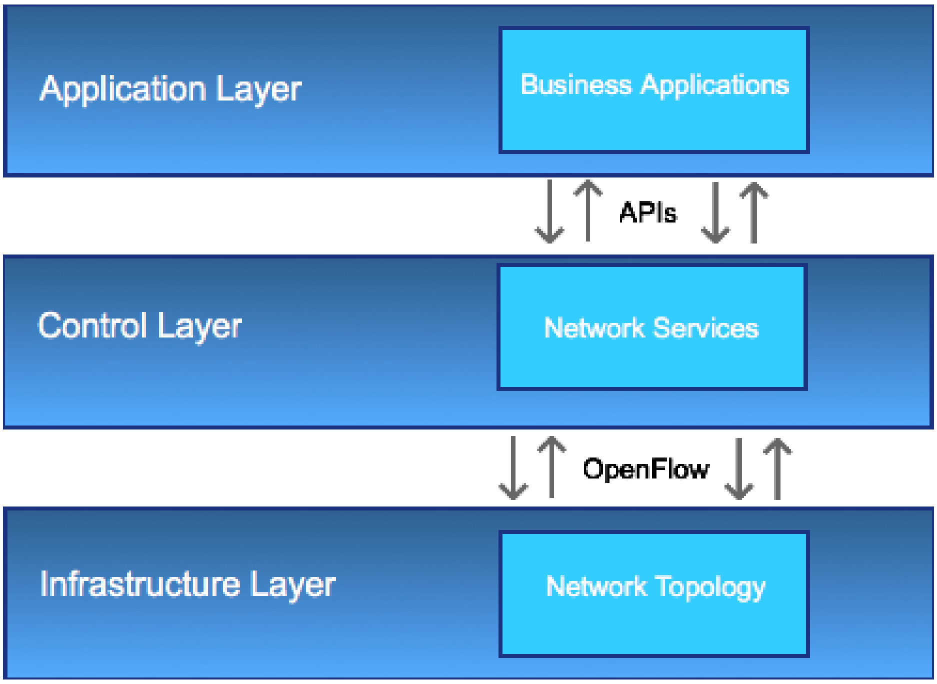
Software-defined networking (SDN) is a "networking architecture in which the control plane and traffic forwarding plane are decoupled" (CAMMARATA et al., 2016). In current networks, the decision as well as forwarding process is made on the devices. This new architecture employs an entity called controller to orchestrate the network decisions. The controller is logically centralised and can be used by developers to create new applications and optimise companies performance. Thus, the controller communicate to the devices in order to set the forwarding rules and, so that, decision can be made considering a global view of the network and abstracting network infrastructure (CAMMARATA et al., 2016). The SDN architecture is (ONF, 2016b):

1. Directly programmable: Network control is directly programmable considering that it is decoupled from forwarding functions, which is made by devices in the forwarding plane.

2. Agile: Separating control from forwarding lets IT designers dynamically adjust network traffic flow to meet changing needs.
3. Centrally managed: Network intelligence is logically centralised, and applications and policy engines can see it as a single component.
4. Programmatically configured: SDN lets network managers configure and optimise network resources dynamically using automated SDN programs.
5. Open standards-based and vendor-neutral: SDN simplifies network design because instructions are provided by SDN controllers, i.e., it does not use vendor-specific devices or protocols.

The communication between controller and device can be established employing different protocols. The most used protocol in this communication is the OpenFlow, which is a standard interface defined between the control and forwarding planes of an SDN (ONF, 2016a). Fig. 2 shows an overview of the SDN architecture: the application, control and infrastructure layers.

Figure 2 – Overview of the SDN architecture.

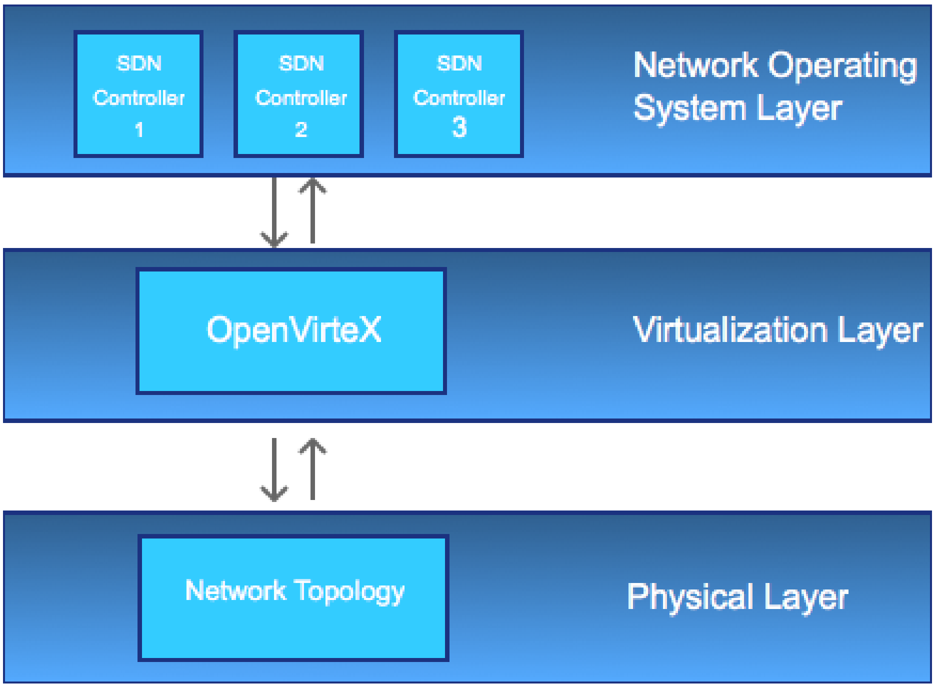


Source: (ONF, 2016b)

In the application layer the business rules are set and developers can use well defined APIs to innovate. The control layer presents a set of services and apply the business rules to the infrastructure devices. Finally, the infrastructure layer corresponds to the physical devices present in the network topology.

Although SDN provides an abstraction of the network topology, it can also be virtualised. An example of virtulisation on SDN is the OpenVirteX project, which is a platform of network virtualisation that is focused on virtual Software-Defined Networks (vSDNs) and can be customised in terms of topology and addressing scheme (AL-SHABIBI et al., 2014a). By using this virtualisation platform, a unique network can be controlled by different controllers and tenants. This control empowers the different tenants to orchestrate a virtual topology built on the top of the physical topology. Fig. 3 shows the system architecture.

Figure 3 – OpenVirteX architecture.



Source: (ON.LAB, 2015)

In Figure 3, the physical layer contains the physical devices from the network topology. The virtualisation layer provides an interface between the physical and the network operating system layer, i.e., this layer creates the vSDNs as well as apply the rules that come from the network operating system layer to the physical layer. Finally, the network operating system layer is composed of different controllers, i.e., it can support different tenants. Note that the application layer presented in Figure 2 is on the top of these three layers.

OpenVirteX presents specific characteristics from which components can be mapped to physical devices as (AL-SHABIBI et al., 2014b):

1. Topology customisation: virtual topologies can be the exactly the same or even subgraphs. The virtual links can represent one or more hops in the physical topology.

2. Resiliency: redundancy can be provided by mapping virtual links or switch onto multiple physical components.
3. Dynamic vSDN reconfiguration: Reconfigurations can be made at runtime in the virtual topology.

3.4 Metrics

In computers networking many metrics are considered in order to measure the quality of the connection. Some metrics are related to time, and others are related to speed. However, in recent years, IT design started to be worried about the energy consumption of network infrastructures and it became an important metric. In this subsection we discuss some important metrics for network infrastructures.

3.4.1 Power Consumption

Power consumption has plenty increased in the last years and it faces reduction as a challenge nowadays. As the green economy is attracting a lot of investments, computer networks plays an important role in energy consumption and cost reduction. Thus, power consumption is an important metric for IT designers in order to monitor the electrical behaviour of the networks and develop improvements to minimise cost and environmental impacts.

3.4.2 PUE

Power Usage Effectiveness (PUE) represents the amount the elapsed power usage in a specific device operation (KOOMEY, 2011). This elapsed power usage considers the device consumption as well as facilities consumption, such as cooling equipments. This metric is usually employed in datacenters but also can be extended to network devices. The PUE of datacenters (KOOMEY, 2011) and network devices can be calculated, respectively, by the equations 3.1 and 3.2:

$$PUE(datacenter) = \frac{Total\ datacenter\ power\ use}{Information\ technology\ equipment\ power\ use} \tag{3.1}$$

$$PUE(networkDevice) = \frac{Total\ operation\ network\ device\ power\ usage}{network\ device\ power\ usage} \tag{3.2}$$

Note that the ideal value PUE is 1.0, which means that all the power used represents the actual network device operation. Any value grater than that shows that other equipment may be consuming power for a proper device operation. For instance, if PUE is 2.0 it shows that the elapsed power consumption is twice as many as the network device consumption.

3.4.3 Delay

Delay is a metric that can be used to measure the quality of service (QoS) provided by network infrastructures (TANENBAUM; WETHERALL, 2011). This metric represents the time that each device takes to receive, process and send the reply packet. Depending on the device, this metric may vary considerably. IT designers plan the network infrastructure concerned about the delay to deliver the resources assuming a high QoS. Monitoring and minimising this metric (associated with other metrics) can improve the network operation.

3.4.4 Packet Loss

Basically, networking QoS can be quantified by the delay and packet loss values. and packet loss requirements. Suppose a scenario where the delay is very low, which may be suitable for real-time application, if the packet loss rate is high the response time may also be compromised. Ensuring a considerable rate of packet loss represents delivering data at good QoS. In order to characterise the packet loss accurately, short-term and long-term packet loss need to be considered (NANANUKUL; KOODLI; DIXIT, 2000). Thus, we can consider the percentage of recently (short-term) and not-recently (long-term) dropped packets, which balance the packet loss ratio by system's current and historical operation. In this work we consider a mean between these two metrics as follow:

$$packetLossRatio = \frac{|shortTermPacketLoss - longTermPacketLoss|}{2} \tag{3.3}$$

3.4.5 Dependability

The dependability (AVIZIENIS; RANDELL, 2001; MEYER; SANDERS, 1993) of a system, that can be understood as the ability to deliver services that can be trusted, is related to issues such as fault tolerance, which is the ability of not failing even when there are faulty components in the system, and reliability, which is the probability that the system will deliver a set of services for a given period of time (EBELING, 1997). Also, availability is quantification of the effect of failure and repair process in a system (KUO; ZUO, 2003).

For any given time period represented by the interval $(0, t)$, the Reliability $R(t)$ is the probability that the component has not failed, i.e., has continued to function from 0 until t. Considering an exponentially distributed Time to Failure (TTF), reliability can be represented by:

$$R(t) = exp\left[-\int_0^t \lambda(t')dt'\right] \tag{3.4}$$

where $\lambda(t')$ is the instantaneous failure rate.

On the other hand, availability (A) can be defined as the ratio of the expected system uptime, i.e., the system working properly, by the expected system up and down times:

$$A = \frac{E\left[Uptime\right]}{E\left[Uptime\right] + E\left[Downtime\right]} \tag{3.5}$$

3.5 Flow Networks

A Flow network can be defined as: "consider a graph consisting of nodes and lines. There is an injection of some commodity at every node, and there are two flows over each line. One flow enters the line from an endpoint and the second flow leaves from the other endpoint. Depending on the sign of its injection, each node can be considered as a supplier or consumer" (FATTAHI; LAVAEI, 2015).

Let G = (V,E), where G is a directed graph composed by nodes $\in$ V and edges (u,v) $\in$ E. Note that if (u,v) $\notin$ E, the capacity c of this edge is equals to 0. Let two special vertices, s, the source, which creates the flow in the network, and t, the target, which consumes the flow. Also, let c the network capacity. A flow in G satisfies (CORMEN, 2009):

- The capacity constraint: $\forall$ u, v $\in$ V, 0$\leqslant$c(u,v)$\leqslant$ f(u,v), where c is the edge capacity and f the edge current flow.
- Flow conservation: $\forall$ u $\in$ V - {s,t}: $\sum$ f(u,v) = $\sum$ f(v,u)

Flow Networks are an important modelling and planning toll for different problems, such as transportation- and logistics-related problems. By using that, we can represent the network's metrics and topology. Also, virtual topologies can be extracted from the physical topology, i.e., a smaller flow network can be extracted from a larger flow network. Therefore, real topologies may be abstracted using this tool to optimise specific metrics such as power consumption.

3.6 Final Remarks

This chapter presented concepts related to the proposed models, ranging from the definition of computer networks infrastructure to flow network. Firstly, computer networks as well as green networking concepts were presented. Then, we disccussed about software-defined networking (SDN) and virtual software-defined networking (vSDN). After that, the metrics adopted in this research were presented (e.g. power consumption, PUE, delay, packet loss, availability and reliability). Finally, flow networks were also presented.

4 Model

In this Chapter, the model adopted to represent networks scenarios called Network Flow Model (NFM) is presented. First, the bases of this model, which relies on the Networkx library (NETWORKX, 2016) is shown. Then, the flow network model, which verifies the data flow between the system components, considering the topology representation as well as the process of adding metrics and restrictions is presented. Finally, the optimisation process to reduce power consumption proposed in this research is described. The proposed model contains information in the devices for supporting the quantification of metrics such as power consumption, PUE and availability. Therefore, this models the physical network topology and bandwidth. In order to build this model, different techniques such as flow networks are employed. In this chapter the details of the model construction as well as its characteristics are shown.

4.1 Networkx

There are many different programing libraries that implement flow networks, such as jung (JUNG, 2010) and grph (GRPH, 2016). Furthermore, there is a widely used open source implementation in Python language called Networkx, which can be defined as "a Python language software package for the creation, manipulation, and study of the structure, dynamics, and functions of complex networks" (NETWORKX, 2016). Among its features Networkx offers, there are (NETWORKX, 2016):

1. Python language data structures for graphs.
2. Graph algorithms (e.g. maximum flow).
3. Automatic generation of classic and random graphs as well as synthetic networks.
4. Widely tested.
5. Fast prototyping, easy to teach, multi-platform.

4.2 Network Flow Model

The main goal of this section is to illustrate the applicability of the proposed model that considers Networkx to build a simple network scenario. The evaluation of this model provides the metric results. First, we present how to build a topology using the network model representation.

```
1    import networkx as nx
2    G = nx.Graph()
3    G.add_node(r0)
4    G.add_node(r1)
5    G.add_edge(r0, r1)
```

Listing 4.1 – Construction of a simple graph G using the Networkx library

Topology

Figure 4 depicts a network topology, in which the router r0 is connected to the router r1 though a serial connection, and both are Cisco 1841 routers. The algorithm presented in listing 4.1 represents this network topology. Firstly, line 1 instantiates the graph G, which is an undirected-cyclic graph. Lines 2 and 3 adds respectively the vertices 0 and 1, which are abstractions of routers r0 and r1. Finally, line 4 connects the vertices. Note that the edges are undirected.

Figure 4 – A simple network scenario.

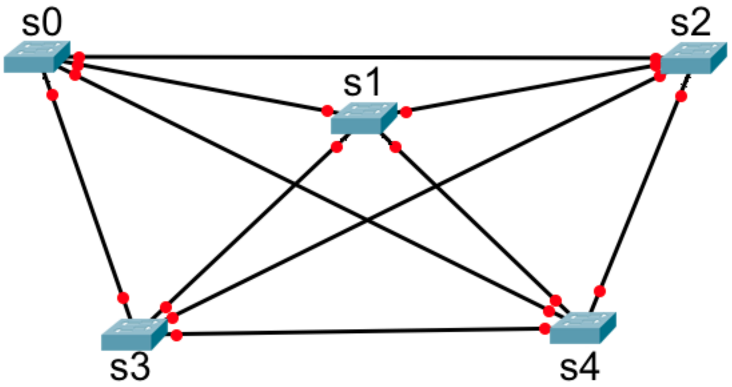
Source: The author

The same steps are followed to build a more complex scenario in which 5 switches (s0 to s4) are connected to each other in Figure 5. Listing 4.2 shows the representation of this scenario using Networkx. Firstly, line 1 instantiates the graph G. Lines 2 to 6 adds all the vertices (0, 1, 2, 3, 4), which are abstractions of all switches (s0, s1, s2, s3, s4). Finally, lines 7 to 19 connects all vertices with undirected edges.

4.2.1 Adding metrics and restrictions

Metrics and network capacity can also be modelled by Networkx. Figure 6 presents a network scenario composed of three fully-connected 1841 Cisco routers (r0, r1, r2). This scenario considers the link capacity, which is illustrated in the figure, and the power consumption as well as the delay (not shown in Figure 6) as presented in Table 2. Listing 4.3 shows the model that represents this scenario. The line 1 instantiates the undirected graph G and lines 2, 3 and 4 add its vertices, i.e., abstraction of routers r0, r1 and r2. Note that each node is defined along with its metrics (e.g., Line 3 represents the router r0 that consumes 10 watts and takes 100 ms). Finally, lines 5 to 7 connects all the vertices.

Figure 5 – Fully-connected network scenario.



Source: The author

```
1  import networkx as nx
2  G = nx.Graph();
3  G.add_node(s0)
4  G.add_node(s1)
5  G.add_node(s2)
6  G.add_node(s3)
7  G.add_node(s4)
8  G.add_edge(s0, s1)
9  G.add_edge(s0, s2)
10 G.add_edge(s0, s3)
11 G.add_edge(s0, s4)
12 G.add_edge(s1, s2)
13 G.add_edge(s1, s3)
14 G.add_edge(s1, s4)
15 G.add_edge(s2, s3)
16 G.add_edge(s2, s4)
17 G.add_edge(s2, s5)
18 G.add_edge(s3, s4)
19 G.add_edge(s3, s5)
20 G.add_edge(s4, s5)
```

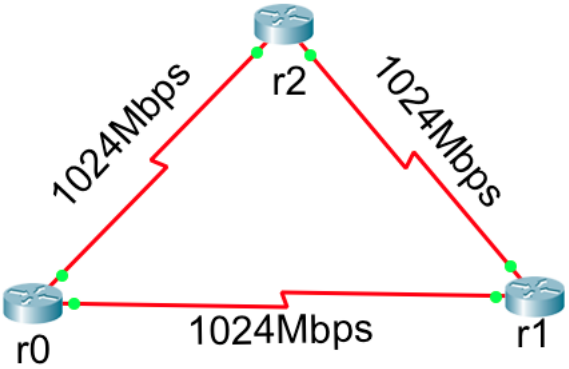
Listing 4.2 – Construction of a more complex graph G using the Networkx library

Router	Power Consumption (kW)	Delay (ms)
r0	10	100
r1	5	150
r2	3	200

Table 2 – Power consumption and delay of routers r1, r2 and r3)

Note that the connections definition also presents the links capacity (e.g., Line 6 shows the connection of router r0 and r1 with the capacity of 1024 ms).

Figure 6 – Network scenario considering links capacity and metrics on devices.



Source: The author

```python
import networkx as nx
G = nx.Graph()
G.add_node(r0, "powerConsumption"=10, "delay"=100)
G.add_node(r1, "powerConsumption"=5, "delay"=150)
G.add_node(r2, "powerConsumption"=3, "delay"=200)
G.add_edge(r0, r1, "capacity" = 1024)
G.add_edge(r0, r2, "capacity" = 1024)
G.add_edge(r1, r2, "capacity" = 1024)
```

Listing 4.3 – Construction of a graph G using the Networkx library considering vertices metrics and link capacities

4.3 Optimisation Process

The optimisation process takes into account the capacity that the tenants need as well as their constraints. Note that this optimisation process is divided to two steps: (i) route selection, this step focuses in the optimisation of the routes that the data packets will flow through; (ii) and power devices off, which aims to reduce the power consumption by turning off some peace of equipments that compose the system without impacting the metrics constraints (e.g., availability, delay). Listing 4.5 shows the implementation of our strategy, which adopts the following parameters:

- Metric: The desired metric, or desired set of metrics (e.g. power consumption, PUE, delay, packet loss, availability, reliability).
- Graph: A graph representing the modelling system.
- Source: The source of data traffic.
- Destination: The destination of data traffic.
- Flow: The data flow that the tenant aims to be able to send.

- List of Weights: A list of weights applied only to multiple metrics. The main goal of this list is to let one to be able to give higher priority to some metrics than to others.

In Listing 4.5, the function *getSortedPaths* aims to sort the topology paths considering the desired metrics. As input, this function considers the metrics, which is a list of desired metrics expressed as Strings; the source, an int value that represents the device that is considered the data source; the target, an int value that represents the device that is considered the data sink; the flow, a double value that represents the bandwidth required by the tenant; and the list of weights, which represents the importance of each metric for the tenant.

Lines 2 and 3 check if the network supports the desired flow and returns an empty list if it does not. Otherwise, the algorithm proceeds putting all possible routes into the variable *sortedPaths* (Line 5). Note that the method called *sortPathsByDesiredMetrics* aims to sort the paths considering the metrics and weights presented by the tenant, and the source and target are also considered. Finally, the sorted list is returned (Line 6).

In Listing 4.5, the function called *getVirtualTopology* aims to provide the virtual topology for each tenant. As input, this function considers the sorted paths, which is the output of the *getSortedPaths* function and contains the all available paths sorted considering the metrics. Thus, the source and target are considered as int values and the flow as a double value.

Firstly, the virtual topology is built in Line 2 as a new empty graph. The capacity needed is also store in the *originalFlow* variable as well as the current considered path in the *currentPath* variable (Lines 3 and 4). Note that they are control variable and they will be used throughout the algorithm operation.

Secondly, From line 5 to line 24 a while loop is presented in order to allocate the vSDNs properly. Considering that the flow is higher than 0 (Line 5), the best path is put in the *subgraph* variable (Line 6). After that, the control variable *currentPath* is incremented in Line 7 ir order to be used in future iterations.

In Line 8, a verification of the maximum flow of the best path is conducted. If the path cannot afford any flow, i.e., its capacity is equals to 0, the algorithm move up to the next iteration (Line 9). However, if this path presents a higher capacity, the capacity of the tenant is partially or totally afforded (depending on the available link capacity) in Line 11. Note that the variable *flow* holds the capacity that the tenant still needs.

Next, in Lines 12 and 13, all the edges of the best path are allocated to the virtual topology. Note that the function called *allocate* is applyied to the virtual topology considering the *originalFlow* variable. After that, in Lines 15 and 16, all nodes of the best path are allocated to the virtual topology. Thus, the *allocateNode* funciont in applyed to the virtual topology and associates the node with the topology.

Finally, in Lines 18 and 19, if the tenant's capacity is not fully afforded, a new iteration is made. On the order hand, if the tenant's capacity is afforded, Line 21 stops the iteration. In Line 24 the virtual topology is given as the function return with the nodes and edges that are present in the best path. Note that this function is executed in every vSDN instance and consider the current network state (state-aware).

```
getSortedPaths(String[] metrics, Graph G, int source, int target, double
    flow, double[] listOfWeights){
  if(flow > G.maximumFlow(source, target)) then
      return([]) // empty list
  end if
  sortedPaths <- G.sortPathsByDesiredMetrics(source, target, metrics,
    listOfWeights)
  return(sortedPaths)}
```

Listing 4.4 – The process of sorting paths considering metrics and weights.

```
1  getVirtualFullTopology(int[][] sortedPaths, int source, int target, double
       flow){
2   virtualTopology <- (new Graph())
3   originalFlow <- flow
4   currentPath <- 0
5   while(flow>0) do
6     subgraph <- sortedPaths[currentPath]
7     currentPath <- currentPath+1
8     if (subgraph.maximumFlow(source,target)<=0) then
9       continue;
10    end if
11    flow <- flow - subgraph.maximumFlow(source,target)
12    for (edge in subgraph) do
13      virtualTopology.allocate(edge, originalFlow)
14    end do
15    for (node in subgraph) do
16      virtualTopology.allocate(node)
17    end do
18    if (flow>0) then
19      continue
20    else
21      break
22    end if
23  end do
24  return(virtualTopology)}
```

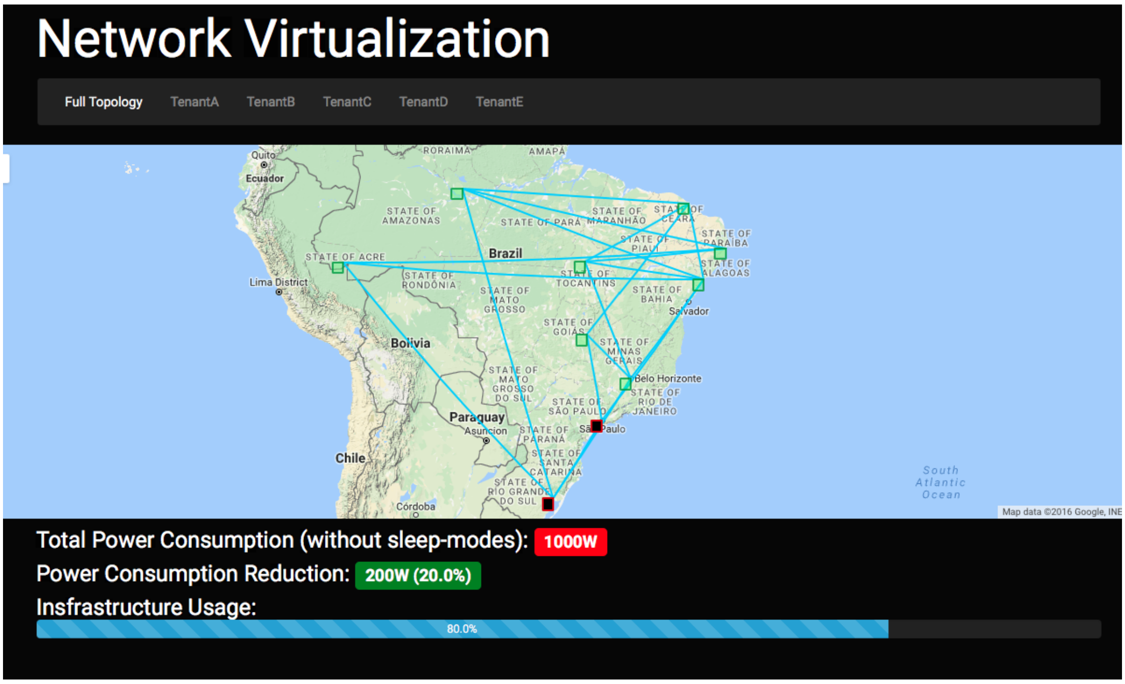
Listing 4.5 – Construction of virtual topologies based on the selected metrics

4.4 Graphical User Interface (GUI)

A Graphical User Interface (GUI), which is a web-based GUI that uses bootstrap (BOOTSTRAP, 2016) and the Google maps API (GOOGLE, 2016), to support designs to have a view of the physical and virtual networks topology including the location of nodes in the map is also proposed. Firstly, a static web page without any interaction with our optimisation algorithm is developed. This page considers five different tenants, but it can be easily extended to support more by including new tabs and instances of the map. Then, this page is integrated with our algorithm which allow one to add nodes to the map considering the coordinates given as parameters and draw the edges among them.

Figure 7 depicts an example of the GUI. The first tab, full topology, presents the physical topology. The other tabs presents virtual topologies that tenants are able to use. Note that, in the physical topology, green nodes represent working and available nodes, and black nodes represent nodes that can be turned off to reduce power consumption.

Figure 7 – Example of the GUI.



Network Virtualization

Total Power Consumption (without sleep-modes): 1000W

Power Consumption Reduction: 200W (20.0%)

Insfrastructure Usage:

Source: The author

4.5 Final Remarks

In this research, models are proposed to conduct the vSDN allocation and power consumption reduction and this chapter presented those models. Firstly, we presented the Networkx library, which is used to build our model. Next, we introduced the Network Flow Model (NFM), which models the data and network considering metrics and restrictions.

Finally, the optimisation process to reduce power consumption proposed, which turns off idle devices, is presented as well as the proposed Graphical User Interface (GUI).

5 Case Studies

This chapter presents three case studies. The main goal of the first study is to show the applicability of the proposed strategy in a small scenario considering three different tenants. After that, we present a case study in the same scenario assuming five different tenants. Finally, the last case study assumes a larger scenario with two different tenants.

In order to illustrate the results of our strategy, we consider wide network scenarios connecting different states of a country. In all case studies the values of the metrics (e.g. delay and packet loss) are attributed considering common values presented in the literature, such as (VEREECKEN et al., 2011), (PINGMAN, 2014), (PINGMAN, 2015), (PINGPLOTTER, 2016), (SHUTE, 2014) and (RAMASWAMY; WENG; WOLF, 2004).

5.1 Case Study I

This case study aims to show the applicability of our proposed strategy considering the topology presented in Figure 8. In this example, we have eight routers connected to each other and each link capacity is illustrated in the same figure (5000 Gbps in each communication link). Table 3 presents the availability as well as the power consumption values adopted for each router. Additionally, Table 4 presents three different tenants and the respective requirements (the adopted metric, data flow needed, source and target nodes) for each one. Therefore, the paths are selected through the availability metric and each tenant has the capacity of 5000Gbps for data. Note that we assumed the period of time of five years in order to compute the metrics. Finally, the availability of each path is presented in Table 5.

Experiments

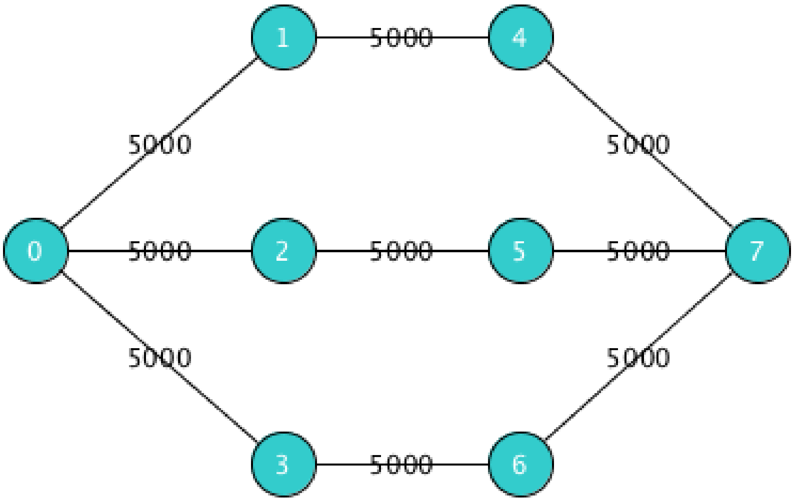
Router	Power Consumption (kW)	Availability (%)
0	16	94
1	8.5	97
2	12	95
3	8	94
4	7.76	91
5	9	97
6	13	98
7	16	99

Table 3 – Availability of devices in case study I.

Tenant	Desired metric	Flow (Gbps)	Source	Target
A	Availability	2500	0	7
B	Availability	2500	0	7
C	Availability	5000	0	7

Table 4 – Tenants' requirements of case study I.

Figure 8 – Physical network topology.



Source: The author

Path	Availability
0-1-4-7	0.82
0-2-5-7	0.85
0-3-6-7	0.85

Table 5 – Availability of each path in case study I.

Firstly, the virtual topology of tenant A is built. As all tenants adopt availability as their desired metric, the path selection from the source to the target is chosen based on this metric. Thus, the path with the highest availability value is chosen as illustrated in Figure 9. Each path availability is given by the multiplication of the availability of each node of the path. The selected path has 85.7% of availability. Without this approach, path such as 0-1-4-7 that has 82.1% of availability, i.e., a lower availability, may be chosen. Furthermore, note that the capacity of the virtual topology respects the tenant's restrictions.

Figure 9 – Virtual topology of tenant A in case study I.

Source: The author

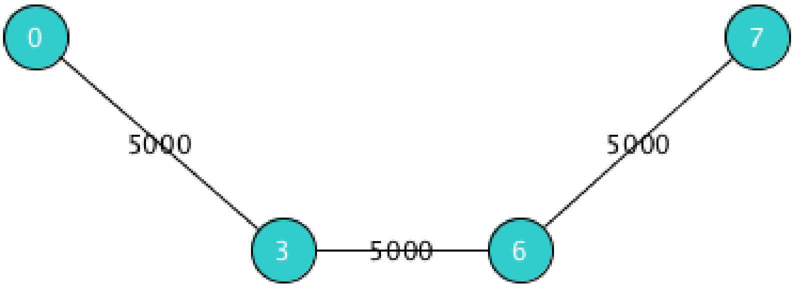
The following step is to build the virtual topology of tenant B. The path used by the tenant A has 2500 Gbps of capacity, which corresponds to the tenant's B need. Note that as the desired metric of tenant B is availability and the best path in terms of this metric can afford the tenant's flow, the path 0-2-5-7 is chosen. Figure 10 presents the virtual topology of the tenant B as well as each link capacity.

Figure 10 – Virtual topology of tenant B in case study I.

Source: The author

Finally, the virtual topology of the tenant C is built. The path that represents the best availability is the 0-3-6-7, considering that the path 0-3-5-7, that is used by tenants A and B, has no capacity to afford the flow of the tenant C. Figure 11 illustrates the path given to the tenant C as well as the capacity. Note that this path has 85.72% of availability, but the other available path (0-1-4-7) has a lower availability (82.14%).

Figure 11 – Virtual topology of tenant C in case study I.



Source: The author

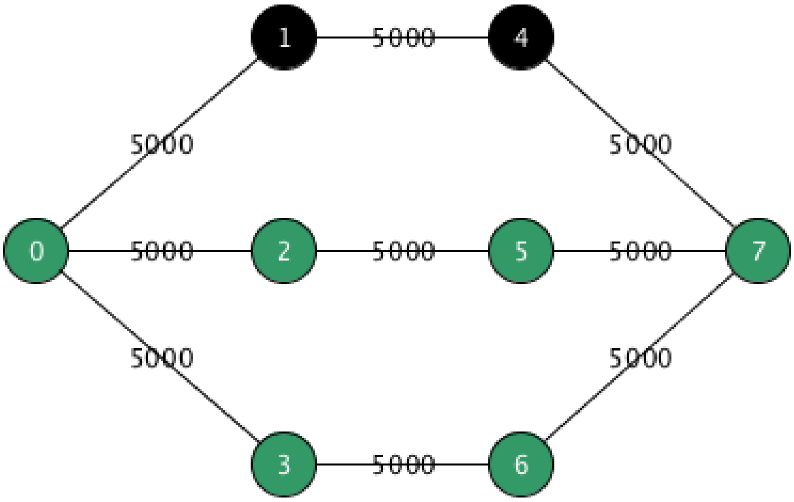
Insight

After allocating all vSDNs to the tenants, the residual physical topology is presented in Figure 12. The green nodes represent the nodes that are being used, i.e., working nodes and the black nodes represent the nodes that are not being used. Thus, following our proposed method, the black nodes can be turned off without impacting the operation of tenants A, B and C. Finally, considering the power consumption values presented in Table 3, our strategy was able to reduce 16.26kW, which represents 18.01% of the power consumption of the whole topology.

5.1.1 Graphical User Interface (GUI)

The proposed GUI shows the physical and virtual topologies and their locations on the map. Figure 13 illustrates the physical topology of case study I. Note that the

Figure 12 – Physical network topology in case study I: used (green) and not used (black) nodes.



Source: The author

black nodes represent idle nodes, which are turned off, but the green nodes are working nodes. The power consumption without sleep modes is also presented along with the power consumption reduction achieved with our strategy.

Figure 14 illustrates all virtual topologies allocated to the tenants. Figures 14a, 14b and 14c present the virtual topology of tenants A, B and C respectively. Although tenants A and B have independent virtual topologies, they use the same physical topology, with 53W of power consumption. Tenant C adopts a different topology, which also consumes 53W.

Figure 13 – Case study I: Full topology of in GUI.

Total Power Consumption (without sleep-modes): 90.26kW

Power Consumption Reduction: 16.26kW (18.0146244183%)

Source: The author

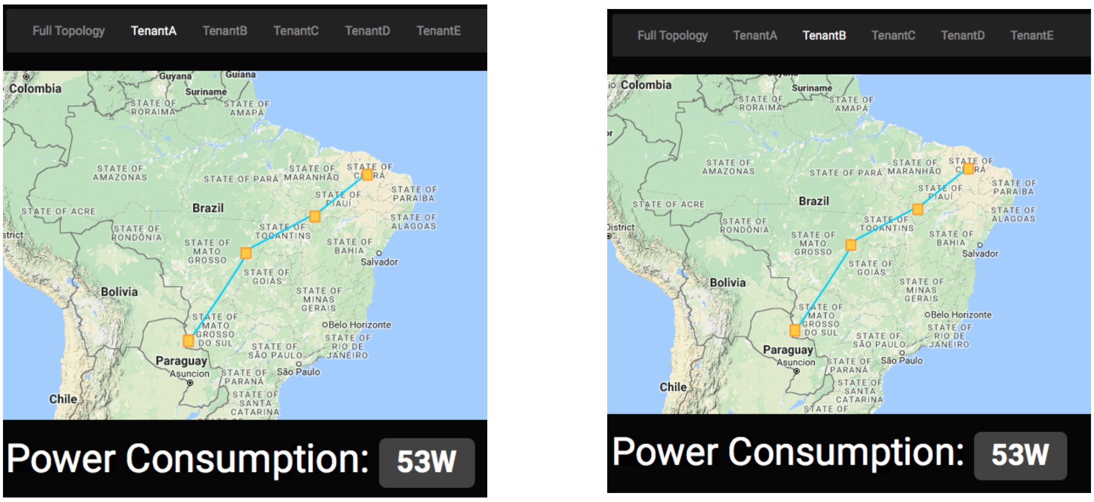
5.2 Case Study II

This case study aims to show the applicability of the proposed strategy considering the same topology presented in the case study I, but assuming five tenants with different requirements. Each device has its metrics values illustrated in Table 6 and each link has its capacity illustrated in the Figure 8. Furthermore, three different tenants with the requirements presented in Table 7 are considered. In this table, the source and target nodes of each tenant are illustrated, as well as the demanded capacity in Gbps and the weights of each metric. Note that, in this case study, each tenant considers two desired metrics, and each metric has a weight. These weights are used to calculate the best path by putting higher priority into a specific desired metric. For instance, the tenant C considers availability with weight 2 and PUE with weight 8, which means that PUE has a higher importance in the path selection process. Note that we assumed the period of time of five years in order to compute the metrics. Finally, Table 8 shows the values of the combination of metrics for each tenant.

Experiments

As illustrated in Table 7, the tenant A considers Packet loss and availability as desired metrics, both with the same importance. Thus, the path that presents the best combination of packet loss and availability is chosen and illustrated in Figure 15. The selected path (0-2-5-7) has 88.7% of availability and mean packet loss of 1.95%. Without this approach, path such as 0-3-6-7 that presents 83.9% of availability and mean packet

Figure 14 – Case study I: virtual topology of the tenants.



(a) Tenant A

(b) Tenant B

(c) Tenant C

Source: The author

	r0	r1	r2	r3	r4	r5	r6	r7
Power Consumption (kW)	15.4	10.4	11.8	7.9	14	13	13	18
PUE	5	1.5	5	1.8	4	2	2	4
Delay (ms)	30	35	29	29	28	43	30	30
Packet Loss (%)	2	2.5	1.8	5	3.3	2	1.4	2
Availability	99	99	98	93	99	97	97	94
Reliability	94	97	95	94	91	97	97	98

Table 6 – Metrics of devices in case study II.

Tenant	Desired Metrics	Weights	Flow (Gbps)	Source	Target
A	Packet Loss; availability	0.5;0.5	2000	0	7
B	PUE; delay	0.6;0.4	1000	0	7
C	Availability; PUE	0.2;0.8	2000	0	7
D	Reliability	1	1000	0	7
E	Delay; Packet Loss	0.5;0.5	1500	0	7

Table 7 – Tenants' requirements of case study II.

Path	Tenants				
	A	B	C	D	E
0-1-4-7	8.97	9.16	9.06	0.81	8.81
0-2-5-7	9.84	8.40	8.33	0.84	9.50
0-3-6-7	8.35	10.0	9.84	0.83	8.75

Table 8 – Values of combined metrics in each path in case study II.

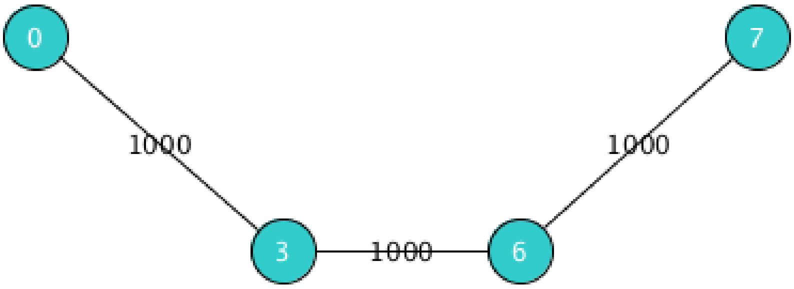
loss of 2.6% may be chosen. Furthermore, note that the capacity of the virtual topology is respected.

Figure 15 – Virtual topology of Tenant A in case study II

Source: The author

Secondly, we built the virtual topology of tenant B. As illustrated in Table 7, the tenant B considers PUE and delay, with 0.6 and 0.4 of weights respectively, as desired metrics. Figure 16 depicts the best path taking into account the combination of PUE and delay considering the weights chosen. The selected path (0-3-6-7) presents a mean PUE of 3.2 as well as a mean delay 29.75 ms. Without this approach, paths such as 0-1-4-7, that presents a mean PUE of 3.625 and a mean delay of 30.75 ms, may be chosen. Note that the capacity of the virtual topology matches the tenant's need.

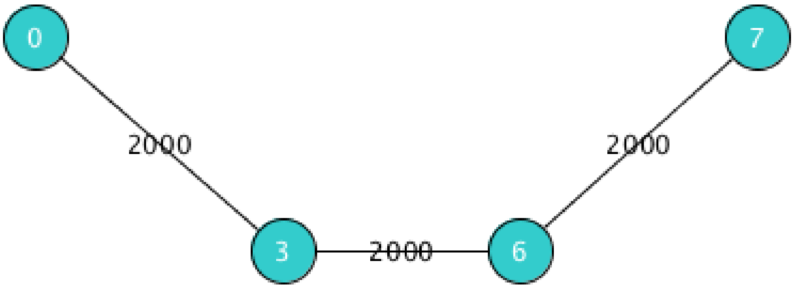
Figure 16 – Virtual topology of Tenant B in case study II



Source: The author

Thirdly, the virtual topology of tenant C is build . As illustrated in Table 7, the tenant C considers availability and PUE. Note that the weights given to each metrics are very different (0.2 and 0.8). The path, that presents the best combination of availability and PUE considering the weights and illustrated in Fig. 17, is composed by the nodes 0, 3, 6 and 7. This topology presents a mean PUE of 3.2 and availability of 83.94%. Although other paths present a better availability, such as 0-2-5-7 that presents 88.7%, this path presents the best combination of PUE and availability. Note that the PUE has a larger importance in the path selection in this case.

Figure 17 – Virtual topology of Tenant C in case study II



Source: The author

Then, we build the virtual topology of tenant D. As illustrated in Table 7, the tenant D considers only reliability. The path 0-2-5-7 presents the best reliability (88.46%). This topology and its bandwidth is presented in Fig. 18.

Figure 18 – Virtual topology of Tenant D in case study II

Source: The author

Finally, the virtual topology of tenant E is built. As illustrated in Table 7, the tenant E considers delay and packet loss with the same importance, i.e., with the same weights. The path 0-2-5-7 that presents the best combination of delay and packet loss (mean of 33 ms and 1.95% respectively) is chosen. This topology is presented in Figure 19. Note that the path 0-3-6-7 presents a lower delay. However, the chosen path considers both metrics, delay and packet loss, and presents the best combination of both.

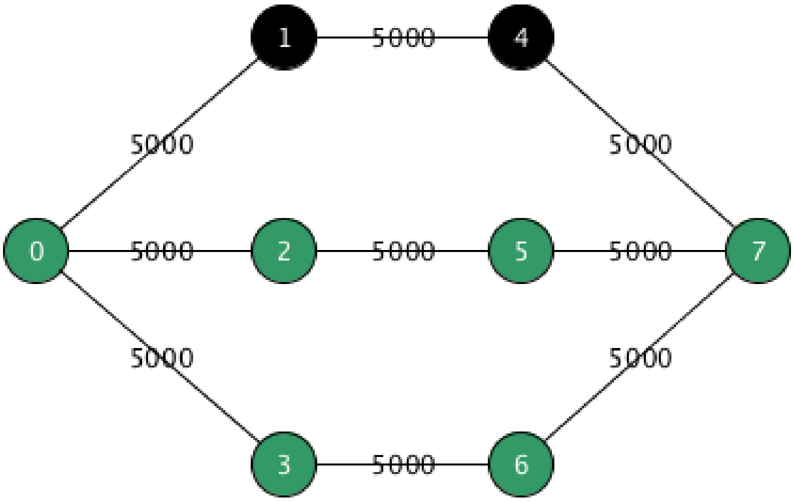
Figure 19 – Virtual topology of Tenant E in case study II

Source: The author

Insight

The physical topology after the vSDNs allocation is presented in Figure 20. The green nodes represent the nodes that are being used, i.e., working nodes and the black nodes represent the nodes that are not being used. Thus, the black nodes can be turned off without comprising the operation of tenants A, B, C, D and E. Finally, considering the power consumption values presented in Table 6 and similarly to case study I, the proposed strategy can reduce 24.4kW of power consumption. As the whole topology consumes 103.5kW, the proposed approach was able to reduce 23.57% of this power consumption.

Figure 20 – Physical network topology in case study II: used (green) and not used (black) nodes.

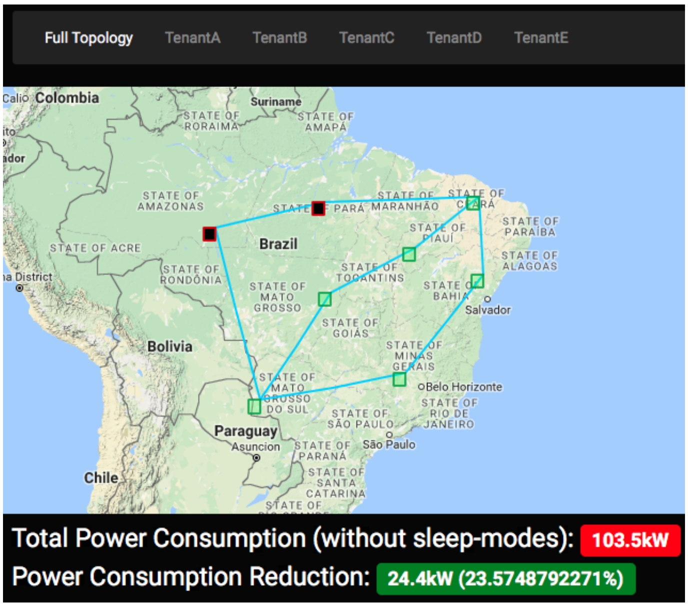


Source: The author

5.2.1 Graphical User Interface (GUI)

In case study II we consider the same topology adopted in case study I, i.e., the position of each device on the map is the same. Figure 21 illustrates the physical topology. The power consumption without sleep modes is 103.5kW whereas our strategy presents 79.1kW.

Figure 21 – Case study II: Full topology of in GUI.



Total Power Consumption (without sleep-modes): 103.5kW
Power Consumption Reduction: 24.4kW (23.5748792271%)

Source: The author

Figure 22 illustrates the virtual topology of all tenants. Figures 22a illustrates the virtual topology of tenant A, which consumes 58.2kW. Figures 22b and 22c show the topologies of tenants B and C, which are composed by the same nodes and consume 54.3kW. Finally, Figures 22d and 22e present the virtual topologies of tenants D and E, which are the same physical topology of tenant A.

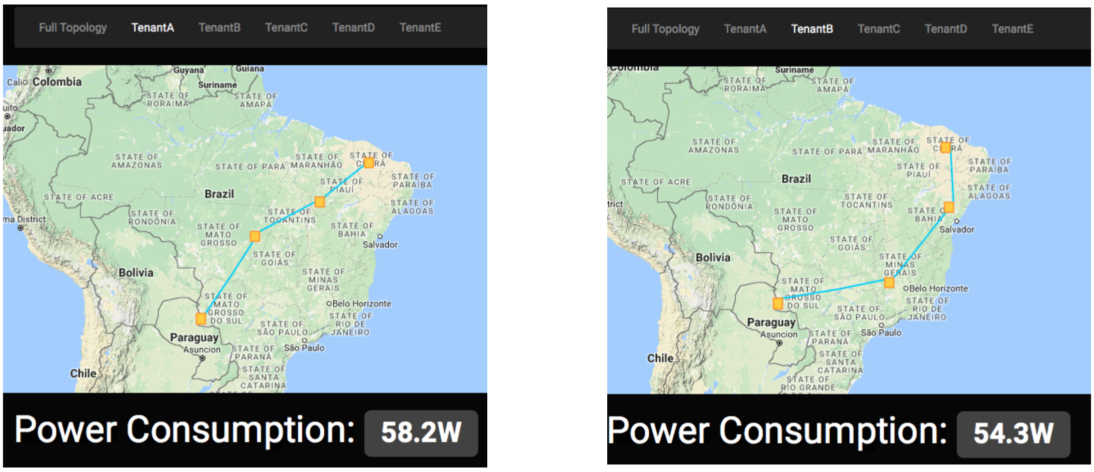
5.3 Case Study III

This case study aims to show the applicability of our strategy in a larger scenario as shown in Figure 23. In this example, twenty routers scattered around the world are considered. Each device has its metrics values and power consumption illustrated, respectively, in Table 9 and 10, and each link has its bandwidth (capacity) illustrated in the same figure. Two tenants are considered and their requirements are illustrated in Table 11, which shows the desired metrics, the weights adopted as well as the flow (Gbps), source and target nodes of each tenant. Note that we assumed the period of time of five years in order to compute the metrics. Finally, the values of the metrics combination of tenants A and B are illustrated, respectively, in Tables 12 and 13. The reader should note that the paths of the tenants are different because the source of each tenant is different.

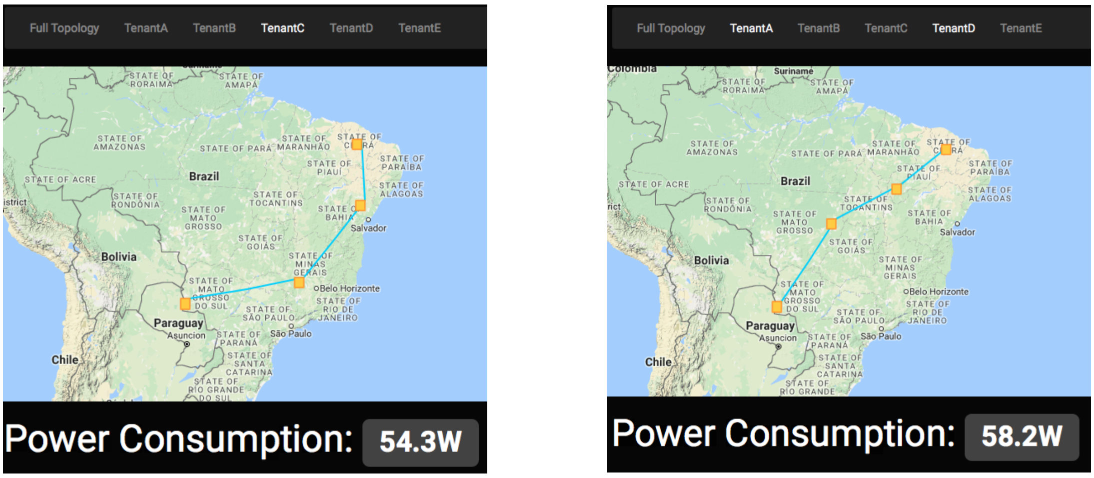
Experiments

Tenant A assumes availability and reliability with the same importance (weight), and the best path is 0-1-2-6-10-11 (74.76% of availability and 69.52% of reliability) is

Figure 22 – Case study II: virtual topology of the tenants.



(a) Tenant A (b) Tenant B



(c) Tenant C (d) Tenant D

(e) Tenant E

Source: The author

Node	PUE	Delay (ms)	Availability (%)	Reliability (%)
0	1.2	30	97	95
1	1.2	35	97	95
2	1.5	40	98	96
3	1.9	45	99	93
4	1.8	32	94	97
5	2.0	31	94	92
6	2.8	32	91	92
7	2.5	28	88	90
8	1.5	22	97	95
9	1.9	23	98	92
10	1.7	31	99	98
11	2.9	35	90	89
12	2.7	27	90	90
13	1.5	25	98	94
14	1.5	35	98	95

Table 9 – Metrics of devices in case study III.

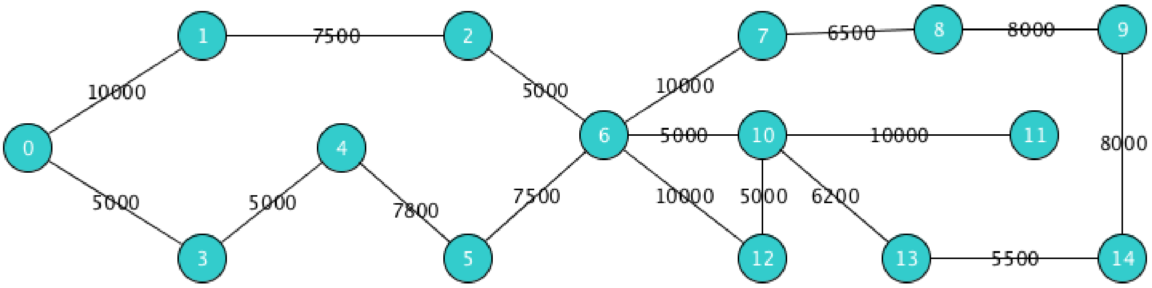
Node	Power Consumption (kW)
0	12
1	15
2	10
3	8
4	10
5	12
6	16.6
7	11
8	10
9	9.8
10	9.2
11	10
12	11
13	12
14	14

Table 10 – Power consumption of devices in case study III.

Tenant	Desired Metrics	Weights	Flow	Source	Target
A	Availability; Reliability	0.5;0.5	5000	0	11
B	Delay; PUE	0.5;0.5	5000	6	11

Table 11 – Tenants' requirements of case study III.

Figure 23 – Physical network topology in case study III.



Source: The author

Paths	Tenant A
0-1- 2- 6-10-11	10
0-3-4-5-6-10-11	9.15
0-1-2-6-12-10-11	9
0-3-4-5-6-12-10-11	8.23
0-1-2-6-7-8-9-14-13-10-11	7.52
0-3-4-5-6-7-8-9-14-13-10-11	6.89

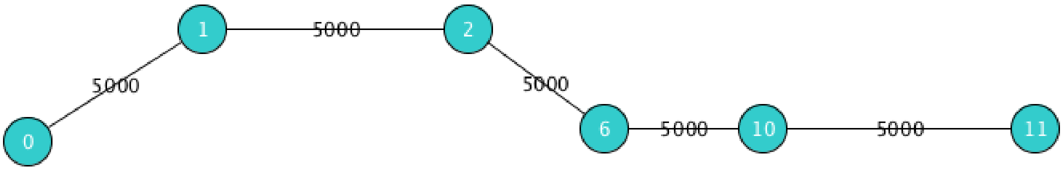
Table 12 – Values of the metrics combination of tenant A in case study III.

Paths	Tenant B
6-10-11	10
6-12-10-11	7.60
6-7-8-9-14-13-10-11	3.91

Table 13 – Values of the metrics combination of tenant B in case study III.

chosen. The topology of tenant A is illustrated in Figure 24. Note that the capacity some links corresponds to the tenant's flow, i.e., this path cannot support any other data flow. Furthermore, without our strategy, a path that presents lower availability and reliability may be chosen.

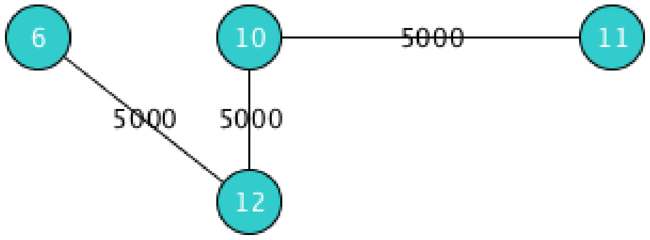
Figure 24 – Virtual topology of Tenant A in case study III.



Source: The author

The following step is to build the virtual topology of tenant B. Tenant B considers delay and PUE with the same priority. Thus, the best path is 6-12-10-11 (31.25ms of delay and 8.075 of PUE). The topology of tenant B is illustrated in Figure 25. Note that the source node for tenants A and B are different. The reader should also observe that tenant B is not able to use the link that connects routers 6 an 10 because it cannot afford the tenant B's flow demands. Furthermore, without our strategy, a path that presents lower delay and PUE may be chosen.

Figure 25 – Virtual topology of Tenant B in case study III

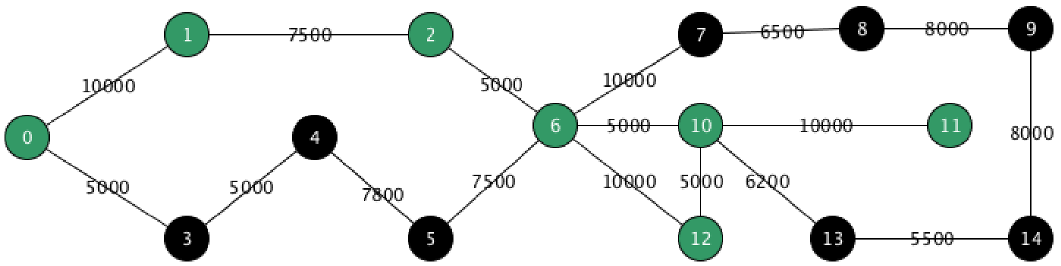


Source: The author

Insights

After allocating all vSDNs to the tenants, the physical topology is presented in Figure 26. The black nodes can be turned off without comprise the operation of tenants A and B. Finally, considering the power consumption values presented in Table 10, the power consumption of nodes 3, 4, 5, 7, 8, 9, 13 and 14 altogether is 86.8kW. Our strategy can reduce the power consumed of these devices by turning them off, which represents 50.87% of the power consumption of the whole topology.

Figure 26 – Physical network topology in case study III: used (green) and not used (black) nodes.

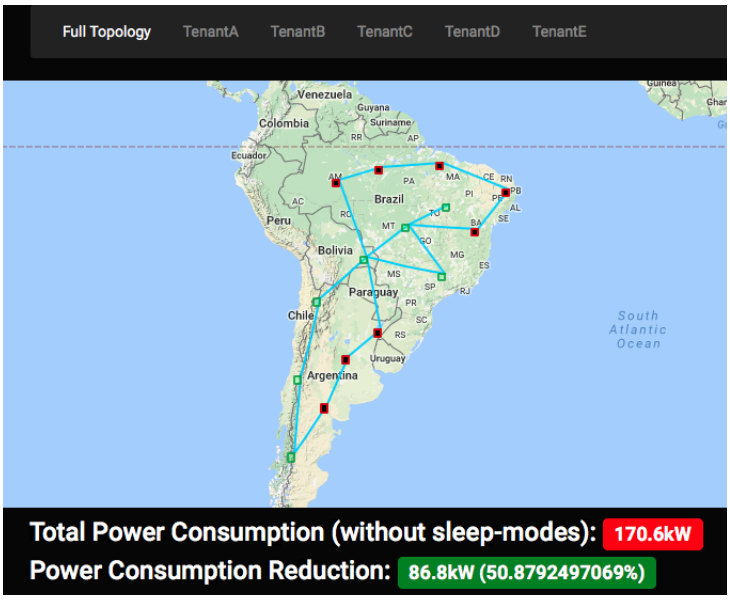


Source: The author

5.3.1 Graphical User Interface

Case study III adopts a larger scenario with physical topology illustrated in Figure 27, which shows that the network is connecting different countries in South America. Also, this figure presents the power consumption without our strategy (170.6kW) as well as the reduction achieved by our strategy (86.8kW).

Figure 27 – Case study III: Full topology of in GUI.



Total Power Consumption (without sleep-modes): 170.6kW

Power Consumption Reduction: 86.8kW (50.8792497069%)

Source: The author

Figure 30 illustrates the topology of all tenants. Figure 30a illustrates the virtual topology of tenant A. This topology is composed by six nodes and presents a power consumption of 72.8kW. Finally, Figure 30b illustrates the virtual topology of tenant B, which is composed by four nodes and consumes 46.8kW.

Figure 28 – Case study III: Topology of tenant A in GUI.

Power Consumption: 72.8W

Source: The author

Figure 29 – Case study III: Topology of tenant B in GUI.

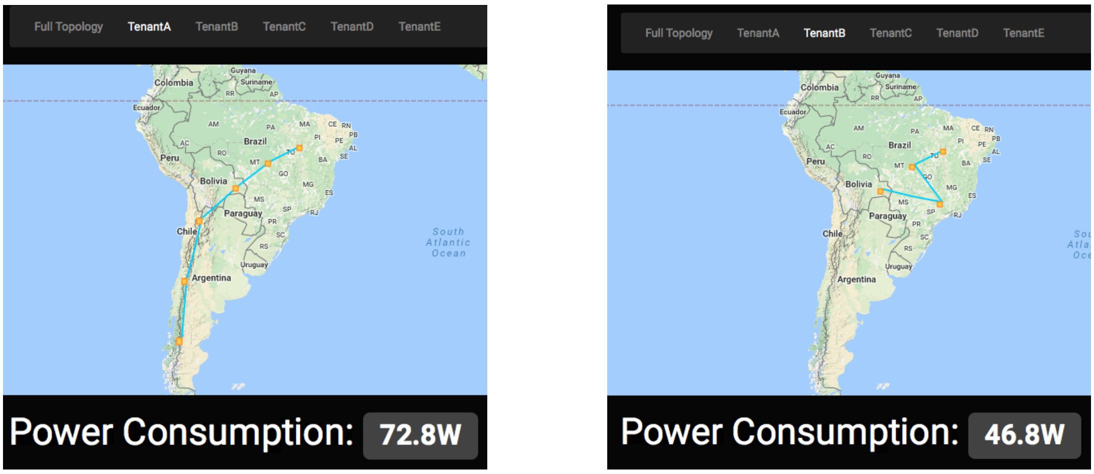
Power Consumption: 46.8W

Source: The author

5.4 Final Remarks

This chapter presented the case studies considered to illustrate the applicability of our models and methodology to allocate vSDN to different tenants and reduce the power consumption of the whole system topology. In addiction, these case studies considered

Figure 30 – Case study III: virtual topology of the tenants.



Power Consumption: 72.8W

Power Consumption: 46.8W

(a) Tenant A

(b) Tenant B

Source: The author

different metrics (e.g. power consumption, PUE, delay, packet loss, availability and reliability) and multiple tenants. Finally, the proposed optimisation process provided interesting results without impacting the system's operation.

6 Conclusion

As the Internet grows, many companies need to reach a broader area. In order to share the same physical network among different tenants OpenVirteX creates virtual software-defined instances of slices of the physical topology and give it to each tenant.

Many companies need, in order to have the a better network control and programmability, addopt software-defined networks (SDN). However, a network topology can be shared among different tenants with technologies such as OpenVirteX. This network division provides control and programmability for each tenant to manage the network slice. Furthermore, the selection of virtual topologies can be made in different ways, with different objectives such as power consumption reduction. This work proposes a strategy to reduce power consumption of software-defined networks scenarios by selecting the most appropriate paths according to specific metrics and respecting the network capacity and turning off idle devices. An optimisation algorithm is proposed, which provides higher priority to the metrics chosen by tenants, and extracts from the physical topology the virtual topology that afford the tenants' demands.

The following sections describe the main contributions of this monograph and provides directions of possible future directions of this research.

6.1 Contributions

This work presented a set of models for representing and analysing computer networks infrastructures considering power consumption, PUE, delay, packet loss, availability and reliability. An optimisation process is also proposed in order to quantify and reduce the power consumption of the whole system. Finally, a graphical user interface (GUI) is proposed to offer a better visualisation of the results of our strategy. For a better understanding, the contributions are detailed below:

- **Modeling:** This work presented a set of models for representing and analysing computer networks infrastructures considering power consumption, PUE, delay, packet loss, availability and reliability. This model can represent many different scenarios considering all described metrics and the capacity of each link. Note that it can also represent the devices that are turned off in order to reduce the power consumption.

- **Algorithm:** An optimisation process is proposed in order to quantify and reduce the power consumption of the whole system. This algorithm prioritises devices in path selection by tents' constraints as well as turns off all devices that may be not in use. Note that this optimisation strategy considers the network capacity and creates virtual topologies to be controlled by each tenant.

- **Graphical User Interface (GUI):** a graphical user interface (GUI) is also proposed to offer a better visualisation of the results of our strategy. This GUI shows the network in the map, highlighting when each node is located and their status (on or off). Finally, the power consumption and power consumption reduction are also shown.

6.2 Future work

Although this work aims to address some specific topics, there are many possibilities to extend it. Possible extensions of this research, and some future directions are:

- **Resilience:** The resilience of the physical topology is not considered in this work. Problems such as virtual topology recovering or virtual topology redefinition are important to a infrastructure provider.
- **Path migration:** In order to optimise infrastructure providers' operation, migrations may be needed, however this work does not consider the process of building new virtual topologies and remove paths that are not being used by specific tenants.
- **Datacenter Networks:** In datacenter networks different metrics may be employed. Problems such as VM migration may impact on path selection and power consumption reduction may present similar results to those presented in this work.

6.3 Final Remarks

This work proposed a strategy to reduce power consumption of networks scenarios by selecting the slices to be given to the tenants, selecting specific metrics and considering network capacity. An optimisation algorithm is proposed, which prioritises the metrics chosen by tenants, and extracts from the physical topology the virtual topology that afford the tenants' needs.

The results show that our strategy present a significant reduction on power consumption and maintain the system working properly. In our case studies our strategy achieved, in some cases, 50% of reduction in the power consumption. As future directions many points can be explored, such as resilience and migration.

Bibliography

ABDUL-QAWY, A. S.; POTLURI, A. Performance analysis of parameters affecting power efficiency in networks. In: *2015 7th International Conference on Communication Systems and Networks (COMSNETS)*. [S.l.: s.n.], 2015. p. 1–8. ISSN 2155-2487.

AHMAD, I. et al. Towards software defined cognitive networking. In: *2015 7th International Conference on New Technologies, Mobility and Security (NTMS)*. [S.l.: s.n.], 2015. p. 1–5. ISSN 2157-4952.

AL-SHABIBI, A. et al. Openvirtex: A network hypervisor. In: *Open Networking Summit 2014 (ONS 2014)*. Santa Clara, CA: USENIX Association, 2014. Disponível em: <https://www.usenix.org/conference/ons2014/technical-sessions/presentation/al-shabibi>.

AL-SHABIBI, A. et al. Openvirtex: Make your virtual sdns programmable. In: ACM. *Proceedings of the third workshop on Hot topics in software defined networking*. [S.l.], 2014. p. 25–30.

AMOROSI, L. et al. Sleep to stay alive: Optimizing reliability in energy-efficient backbone networks. In: *2015 17th International Conference on Transparent Optical Networks (ICTON)*. [S.l.: s.n.], 2015. p. 1–4. ISSN 2162-7339.

ANSARI, N.; HAN, T.; TAHERI, M. Gate: greening at the edges. *IEEE Wireless Communications*, v. 23, n. 2, p. 62–69, April 2016. ISSN 1536-1284.

ATTIAH, A. et al. Adp: An adaptive feedback approach for energy-efficient wireless sensor networks. In: *2016 International Conference on Computing, Networking and Communications (ICNC)*. [S.l.: s.n.], 2016. p. 1–7.

AVIZIENIS, J. L. A.; RANDELL, B. *Fundamental concepts of dependability*. [S.l.]: Technical Report Series-University of Newcastle upon Tyne, Computing Science, 2001.

BECKMANN, E. C.; JAUCO, L. M.; KOO, S. G. M. Green networking: Developing sustainable computer networks. In: *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. [S.l.: s.n.], 2014. p. 3785–3790. ISSN 1062-922X.

BIANZINO, A. P. et al. A survey of green networking research. *IEEE Communications Surveys Tutorials*, v. 14, n. 1, p. 3–20, First 2012. ISSN 1553-877X.

BIANZINO, A. P. et al. A survey of green networking research. *IEEE Communications Surveys Tutorials*, v. 14, n. 1, p. 3–20, First 2012. ISSN 1553-877X.

BOLLA, R. et al. Enabling backbone networks to sleep. *IEEE Network*, v. 25, n. 2, p. 26–31, March 2011. ISSN 0890-8044.

BOLLA, R. et al. Energy efficiency in the future internet: A survey of existing approaches and trends in energy-aware fixed network infrastructures. *IEEE Communications Surveys Tutorials*, v. 13, n. 2, p. 223–244, Second 2011. ISSN 1553-877X.

BOLLA, R. et al. Saving energy by delegating network activity to home gateways. *IEEE Transactions on Consumer Electronics*, v. 61, n. 4, p. 445–453, November 2015. ISSN 0098-3063.

BONETTO, E.; MELLIA, M.; MEO, M. Energy profiling of isp points of presence. In: *2012 IEEE International Conference on Communications (ICC)*. [S.l.: s.n.], 2012. p. 5973–5977. ISSN 1550-3607.

BONOMI F., M. R. Z. J. A. S. Fog computing and its role in the internet of things. In: *Proceedings of 1st Workshop on Mobile Cloud Computing (MCC 2012), pp. 13–16. ACM, 2012.* [S.l.: s.n.], 2012.

BOOTSTRAP. *Bootstrap · The world's most popular mobile-first and responsive front-end framework.* 2016. Available in: <http://getbootstrap.com>. Accessed: November 2016.

BUYYA, C. S. Y. R.; VENUGOPAL, S. Market-oriented cloud computing: Vision, hype, and reality for delivering it services as computing utilities. In: *High Performance Computing and Communications, 2008. HPCC '08. 10th IEEE International Conference on, Dalian, 2008, pp. 5-13.* [S.l.: s.n.], 2008.

CALLOU, G. et al. Models for dependability and sustainability analysis of data center cooling architectures. In: *IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN 2012)*. [S.l.: s.n.], 2012. p. 1–6. ISSN 2325-6648.

CAMMARATA, G. et al. Assdn: A framework for adaptive strategies for software-defined networking. In: *2016 IEEE 25th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)*. [S.l.: s.n.], 2016. p. 101–106.

CHIARAVIGLIO, L. et al. Increasing device lifetime in backbone networks with sleep modes. In: *Software, Telecommunications and Computer Networks (SoftCOM), 2013 21st International Conference on*. [S.l.: s.n.], 2013. p. 1–6.

CHIARAVIGLIO, L. et al. Sleep modes effectiveness in backbone networks with limited configurations. *Computer Networks*, v. 57, n. 15, p. 2931 – 2948, 2013. ISSN 1389-1286. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1389128613002028>.

CHIARAVIGLIO, L. et al. Lifetime awareness in backbone networks with sleep modes. In: *Reliable Networks Design and Modeling (RNDM), 2015 7th International Workshop on*. [S.l.: s.n.], 2015. p. 281–287.

CHIARAVIGLIO, L.; MELLIA, M.; NERI, F. Energy-aware backbone networks: A case study. In: *2009 IEEE International Conference on Communications Workshops*. [S.l.: s.n.], 2009. p. 1–5. ISSN 2164-7038.

CHIARAVIGLIO, L.; MELLIA, M.; NERI, F. Minimizing isp network energy cost: Formulation and solutions. *IEEE/ACM Transactions on Networking*, v. 20, n. 2, p. 463–476, April 2012. ISSN 1063-6692.

CHOI, H.-H.; LEE, J.-R. A biologically-inspired power control algorithm for energy-efficient cellular networks. *Energies*, v. 9, n. 3, p. 161, 2016. ISSN 1996-1073. Disponível em: <http://www.mdpi.com/1996-1073/9/3/161>.

CISCO. *White paper: Cisco VNI Forecast and Methodology, 2015-2020*. 2014. Available in: <http://www.cisco.com/c/en/us/solutions/collateral/service-provider/ipngn-ip-next-generation-network/white_paper_c11-481360.pdf>. Accessed: November 2016.

CORIGLIANO, S.; TRUNFIO, P. Exploiting sleep-and-wake strategies in the gnutella network. In: *Collaboration Technologies and Systems (CTS), 2014 International Conference on*. [S.l.: s.n.], 2014. p. 406–412.

CORMEN, T. H. *Introduction to algorithms*. [S.l.]: MIT press, 2009.

DIXIT, J. K. M.; KUMAR, R. Internet of things and its challenges. In: *Green Computing and Internet of Things (ICGCIoT), 2015 International Conference on, Noida, 2015, pp. 810-814, 2015*. [S.l.: s.n.], 2015.

DUAN, L.; ZHAN, D.; HOHNERLEIN, J. Optimizing cloud data center energy efficiency via dynamic prediction of cpu idle intervals. In: *2015 IEEE 8th International Conference on Cloud Computing*. [S.l.: s.n.], 2015. p. 985–988. ISSN 2159-6182.

EBELING, C. *An introduction to reliability and maintainability engineering*. [S.l.]: Waveland Press, 1997.

FATTAHI, S.; LAVAEI, J. Convex analysis of generalized flow networks. In: *2015 54th IEEE Conference on Decision and Control (CDC)*. [S.l.: s.n.], 2015. p. 1569–1576.

FEIZI, S.; ZHANG, A.; MEDARD, M. A network flow approach in cloud computing. In: *Information Sciences and Systems (CISS), 2013 47th Annual Conference on*. [S.l.: s.n.], 2013. p. 1–6.

FERREIRA, J.; CALLOU, G.; MACIEL, P. A power load distribution algorithm to optimize data center electrical flow. *Energies*, v. 6, n. 7, p. 3422, 2013. ISSN 1996-1073. Disponível em: <http://www.mdpi.com/1996-1073/6/7/3422>.

FORD, L. R. J.; FULKERSON, D. R. Flows in networks. In: *Princeton University Press, Princeton, N.J.* [S.l.: s.n.], 1962.

FRANCOIS, F. et al. Optimization for time-driven link sleeping reconfigurations in isp backbone networks. In: *2012 IEEE Network Operations and Management Symposium*. [S.l.: s.n.], 2012. p. 221–228. ISSN 1542-1201.

GOMES, R. L. et al. Bandwidth-aware allocation of resilient virtual software defined networks. *Computer Networks*, v. 100, p. 179 – 194, 2016. ISSN 1389-1286. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1389128616300524>.

GOMES, R. L. et al. A combined energy-bandwidth approach to allocate resilient virtual software defined networks. *Journal of Network and Computer Applications*, v. 69, p. 98 – 106, 2016. ISSN 1084-8045. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1084804516300728>.

GOMES, R. L. et al. State-aware allocation of reliable virtual software defined networks based on bandwidth and energy. In: *2016 13th IEEE Annual Consumer Communications Networking Conference (CCNC)*. [S.l.: s.n.], 2016. p. 411–416.

GOOGLE. *Google Maps API*. 2016. Available in: <https://developers.google.com/maps/?hl=pt-br>. Accessed: November 2016.

GRPH. *High erformance graph library for Java*. 2016. Available in: <http://www.i3s.unice.fr/~hogie/software/index.php?name=grph>. Accessed: November 2016.

HASAN, M. S. et al. Cloud energy broker: Towards sla-driven green energy planning for iaas providers. In: *High Performance Computing and Communications, 2014 IEEE 6th Intl Symp on Cyberspace Safety and Security, 2014 IEEE 11th Intl Conf on Embedded Software and Syst (HPCC,CSS,ICESS), 2014 IEEE Intl Conf on*. [S.l.: s.n.], 2014. p. 248–255.

HASSAN, H. A. H. et al. Renewable energy usage in the context of energy-efficient mobile network. In: *2015 IEEE 81st Vehicular Technology Conference (VTC Spring)*. [S.l.: s.n.], 2015. p. 1–7. ISSN 1550-2252.

HE, F. et al. Ict, trade in financial service and economic growth. In: *Logistics, Informatics and Service Sciences (LISS), 2015 International Conference on*. [S.l.: s.n.], 2015. p. 1–4.

JIN, C.; SHENG, X.; GHOSH, P. Energy efficient algorithms for electric vehicle charging with intermittent renewable energy sources. In: *2013 IEEE Power Energy Society General Meeting*. [S.l.: s.n.], 2013. p. 1–5. ISSN 1932-5517.

JUNG. *ava Universal Network/Graph Framework*. 2010. Available in: <http://jung.sourceforge.net>. Accessed: November 2016.

KHOSRAVANI, H. R. et al. A comparison of energy consumption prediction models based on neural networks of a bioclimatic building. *Energies*, v. 9, n. 1, p. 57, 2016. ISSN 1996-1073. Disponível em: <http://www.mdpi.com/1996-1073/9/1/57>.

KOOMEY, J. Growth in data center electricity use 2005 to 2010. *A report by Analytical Press, completed at the request of The New York Times*, v. 9, 2011.

KUO, W.; ZUO, M. J. *Optimal Reliability Modeling - Principles and Applications*. [S.l.]: Wiley, 2003. ISBN 978-0-13-212695-3.

LAMB, J. Green it and use of private cloud computing in south africa. In: *Emerging Technologies for a Smarter World (CEWIT), 2011 8th International Conference Expo on*. [S.l.: s.n.], 2011. p. 1–6.

MCKINSEYCOMPANY. *Revolutionizing Data Center Energy Efficiency*. 2008. Available in: <http://www.sallan.org/pdf-docs/McKinsey_Data_Center_Efficiency.pdf>. Accessed: November 2016.

MEYER, J. F.; SANDERS, W. H. Specification and construction of performability models. In: *Proceedings of the Second International Workshop on Performability Modeling of Computer and Communication Systems*. [S.l.: s.n.], 1993. p. 28–30.

MINERAUD, J. et al. Hybrid renewable energy routing for isp networks. *INFOCOM, 2016 Proceedings IEEE*, p. 1–9, 2016.

NANANUKUL, S.; KOODLI, R.; DIXIT, S. Controlling short-term packet loss ratios using an adaptive pushout scheme. In: *High Performance Switching and Routing, 2000. ATM 2000. Proceedings of the IEEE Conference on*. [S.l.: s.n.], 2000. p. 49–54.

NETWORKX. *High-productivity software for complex networks*. 2016. Available in: <https://networkx.github.io>. Accessed: November 2016.

NWIZEGE, K. S. et al. The impact of ict on computer applications. In: *Computer Modeling and Simulation (EMS), 2011 Fifth UKSim European Symposium on.* [S.l.: s.n.], 2011. p. 435–439.

OKONOR, O. et al. Link sleeping and wake-up optimization for energy aware isp networks. In: *2014 IEEE Symposium on Computers and Communications (ISCC).* [S.l.: s.n.], 2014. p. 1–7. ISSN 1530-1346.

ONF, O. N. F. *OpenFlow.* 2016. Available in: <https://www.opennetworking.org/sdn-resources/openflow>. Accessed: November 2016.

ONF, O. N. F. *Software-Defined Networking (SDN) Definition.* 2016. Available in: <https://www.opennetworking.org/sdn-resources/sdn-definition>. Accessed: November 2016.

ON.LAB, O. N. L. *Programmable virtual networks.* 2015. Available in: <http://ovx.onlab.us>. Accessed: November 2016.

PAUL, D.; ZHONG, W. D.; BOSE, S. K. Energy efficient scheduling in data centers. In: *2015 IEEE International Conference on Communications (ICC).* [S.l.: s.n.], 2015. p. 5948–5953. ISSN 1550-3607.

PINGMAN. *What's "normal" for latency and packet loss?* 2014. Available in: <https://www.pingman.com/kb/42>. Accessed: November 2016.

PINGMAN. *Packet loss or latency at intermediate hops.* 2015. Available in: <https://www.pingman.com/kb/24>. Accessed: November 2016.

PINGPLOTTER. *Common Network Problems.* 2016. Available in: <https://www.pingplotter.com/common-network-problems/>. Accessed: November 2016.

POHRADSKý, P.; LONDáK, J.; ČAčíKOVá, M. Application of ict in pre-school education. In: *ELMAR, 2010 PROCEEDINGS.* [S.l.: s.n.], 2010. p. 159–162. ISSN 1334-2630.

RAMASWAMY, R.; WENG, N.; WOLF, T. Characterizing network processing delay. In: IEEE. *Global Telecommunications Conference, 2004. GLOBECOM'04. IEEE.* [S.l.], 2004. v. 3, p. 1629–1634.

SHRUTHI, D. Information and communication technologies for sustainable renewable energy sources promoting green environmental growth. In: *Issues and Challenges in Intelligent Computing Techniques (ICICT), 2014 International Conference on.* [S.l.: s.n.], 2014. p. 820–824.

SHUTE, G. *Network Delays and Losses.* 2014. Available in: <https://www.d.umn.edu/~gshute/net/delays-losses.xhtml>. Accessed: November 2016.

TANENBAUM, A. S.; WETHERALL, D. J. *Computer Networks.* 5th. ed. [S.l.]: Prentice Hall, 2011. ISBN 978-0-13-212695-3.

TSAI, X. S. W. T.; BALASOORIYA, J. Enterprise usability of cloud computing environments: Issues and challenges. In: *Enabling Technologies: Infrastructures for Collaborative Enterprises (WETICE), 2010 19th IEEE International Workshop on, Larissa, 2010, pp. 54-59.* [S.l.: s.n.], 2010.

TSAI, X. S. W. T.; BALASOORIYA, J. Service-oriented cloud computing architecture. In: *Information Technology: New Generations (ITNG), 2010 Seventh International Conference on, Las Vegas, NV, 2010, pp. 684-689.* [S.l.: s.n.], 2010.

VEREECKEN, W. et al. Power consumption in telecommunication networks: overview and reduction strategies. *IEEE Communications Magazine*, IEEE, v. 49, n. 6, p. 62–69, 2011.

WANG, B. W. X.; HUANG, J. Cloud computing and its key techniques. In: *Computer Science and Automation Engineering (CSAE), 2011 IEEE International Conference on, Shanghai, 2011, pp. 404-410, 2011.* [S.l.: s.n.], 2015.

WANG, Y. et al. An energy-efficient {SDN} based sleep scheduling algorithm for {WSNs}. *Journal of Network and Computer Applications*, v. 59, p. 39 – 45, 2016. ISSN 1084-8045. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1084804515000910>.

YANG, T.; LEE, Y.; ZOMAYA, A. Collective energy-efficiency approach to data center networks planning. *IEEE Transactions on Cloud Computing*, PP, n. 99, p. 1–1, 2016. ISSN 2168-7161.

ZENG, M. K. W.; WATSON, P. Opacity in internet of things with cloud computing (short paper). In: *2015 IEEE 8th International Conference on Service-Oriented Computing and Applications (SOCA), Rome, 2015, pp. 201-207, 2015.* [S.l.: s.n.], 2015.