Energy-Aware Inter-Data Center Virtual Machine Migration over Elastic Optical Networks

by

Fatima Salehnejad Amri
M.Sc., K.N. Toosi University of Technology, 2020
B.Sc., Babol Nooshirvani University of Technology, 2018

A Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

in the Department of Computer Science

Energy-Aware Inter-Data Center Virtual Machine Migration over Elastic Optical Networks

by

Fatima Salehnejad Amri
M.Sc., K.N. Toosi University of Technology, 2020
B.Sc., Babol Nooshirvani University of Technology, 2018

Supervisory Committee

Dr. Jianping Pan, Supervisor
(Department of Computer Science)

Dr. Sudhakar Ganti, Departmental Member
(Department of Computer Science)

Supervisory Committee

Dr. Jianping Pan, Supervisor
(Department of Computer Science)

Dr. Sudhakar Ganti, Departmental Member
(Department of Computer Science)

ABSTRACT

The rapid growth of data processing demands in large-scale data centers (DCs) has increased the network's brown energy (BE) consumption. The BE is generated from fossil fuels and has an adverse effect on the environment. Since most DCs are now powered by both BE and renewable energy (RE), migrating workloads from DCs with insufficient RE to DCs with sufficient RE can decrease the total BE consumption in the network. However, selecting a destination DC under the dynamic nature of the underlying network and without any prior information is challenging. In addition, although migration can help reduce BE consumption, it comes with an additional cost due to using network devices for the migration.

This thesis focuses on minimizing the total cost, which includes BE consumption costs, migration costs, and optical network device costs. To achieve this goal, we optimize both the DC selection process and the efficient transfer of virtual machines (VMs) between DCs. In Chapter 3, we formulate the DC selection as a two-stage Multi-Armed Bandit (MAB) problem. First, we define an arm as a destination DC, and select a DC with the lowest power consumption and available RE. In the second stage, we define the arm as a path, and we implement MAB to find a path with the lowest delay from the source to the selected destination DC. Proposing and utilizing the modified sliding-window lower confidence bound (MSW-LCB), we estimate the lowest power consumption among DCs and the lowest migration cost at each round to find a proper destination DC and path, respectively. Additionally, we adopt optical grooming techniques to minimize the cost of optical network devices used during VM transfer.

Furthermore, to validate the effectiveness of our algorithm, we conduct an evaluation using three different real-world datasets to provide different inputs for the algorithm. This evaluation is assessed on USNET topology. In comparison to the sliding-window lower confidence bound (SW-LCB) and two other MAB-based algorithms, namely the knapsack–based upper confidence bound (KUBE) and ϵ-Greedy, the MSW-LCB approach reduces the total cost by about 15%, 23%, and 34%, respectively, while having low regret. The MSW-LCB regret and migration costs are demonstrated to be around 15% lower than SW-LCB, respectively. We also evaluate our algorithm in two scenarios—one with and one without optical grooming—to show the efficiency of optical grooming in decreasing optical network costs. The results indicate a 12% drop in network costs. The results of this evaluation provide valuable insights. This indicates that our algorithm is capable of handling real-world data and effectively addressing the challenges associated with inter-DC VM migration.

Contents

List of Figures

List of Acronyms

Anycast-JRE	Anycast with joint resources ergodic
BE	Brown energy
CPU	Central processing unit
DC	Data center
DWDM	Dense wavelength-division multiplexing
EON	Elastic optical network
KUBE	Knapsack–based upper confidence bound
MAB	Multi-armed bandit
MSW-LCB	Modified sliding window lower confidence bound
OEO	Optical-Electrical-Optical
OFDM	Orthogonal frequency-division multiplexing
OTN	Optical transport network
PUE	Power usage effectiveness
RAM	Random-access memory
RE	Renewable energy
ROADM	Reconfigurable optical add/drop multiplexer
SBVT	Sliceable bandwidth variable transponder
SDN	Software-defined networking
SLA	Service level agreements
SW-LCB	Sliding window lower confidence bound
TDM	Time-division multiplexing
VM	Virtual machine
WDM	Wavelength-division multiplexing
WSS	Wavelength-selective switches

List of Symbols

$G(N, L)$	Network graph G with sets of nodes N and links L
$P_{r,s}$	Power consumption of server r in DC s
$P_{r,s}^{\text{Static}}$	Static power consumption of server r in DC s
$P_{r,s,\theta}^{\text{Dyna}}$	Dynamic power consumption of server r in DC s in round θ
P_{Idle}	Idle power consumption of server
P_{Peak}	Peak power consumption of server
η	Power usage effectiveness of the DC
$u_{r,s,\theta}$	CPU utilization of the r-th server in the s-th DC in round θ
$P_{s,\theta}$	Power consumption of DC s in round θ
R_s	Available renewable energy in DC s
$\Gamma_{s,\theta}$	Brown energy power consumption in DC s in round θ
C_{DC}	Cost of brown energy consumption
C_{Net}	Cost of optical network elements
C_{Mig}	Migration cost
α_s	Energy cost per unit for the DC s
a_θ	Selected path in round θ
m_θ	Selected destination DC in round θ
c_θ^{s,a_θ}	Migration cost from DC s to the destination DC through path a_θ in round θ
ρ	Migration cost per unit
d_θ^{s,a_θ}	Transmission delay from source s to destination DC through path a_θ at round θ
C_{Net}	Cost of optical network elements
C_{T}	Cost of SBVTs
C_{ROADM}	Cost of ROADMs
C_{Cap}	Capital expenditure cost
β	Migration costs per unit associated with transponder consumption

T_θ	The number of transponders used for the VM migration in round θ
M	The degree of the ROADM
c_{Amp}	Cost of amplifier
c_{WSS}	Cost of wavelength selective switches
$\bar{f}_\theta(i,\tau)$	The function to calculate the average of component i on τ previous round
$E_\theta(i,\tau)$	The adjustment term required for MAB algorithm
τ	The length of the sliding window
$w(k,\tau)$	Weight associated with the chosen arm in round k having window size τ
ω	Maximum number of CPU cores on the server

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my supervisor Professor Jianping Pan, for his invaluable guidance, support, and encouragement throughout my research. His expertise and insightful comments have been invaluable in shaping the direction of this work.

I would also like to thank my parents for their unwavering love, encouragement, and support throughout my academic journey. Without their constant support and encouragement, I would not have been able to achieve this milestone.

Finally, I would like to extend my gratitude to all those who have supported me in various ways during the course of my research, including my colleagues, friends, and family. Thank you all for your support, encouragement, and inspiration.

"As you start to walk on the way, the way appears."

Rumi

DEDICATION

To future generations of researchers and scholars, who will build on the foundation that we have laid. May your work be guided by curiosity, compassion, and a commitment to making the world a better place.

Chapter 1

Introduction

The increasing demand for computing and data processing has put a significant strain on cloud data centers (DCs) and their computing and storage systems [7]. In recent years, the energy efficiency of DCs has received extensive attention due to its high economic, environmental, and performance impacts. The energy consumption of DCs has reportedly surpassed 20% of all energy used in information and communications technologies [8]. Essentially, DCs are powered by both brown energy (BE) and renewable energy (RE). The BE is generated from fossil fuels, which produce carbon emissions. In contrast, RE is collected from renewable resources, e.g., solar and wind, which are environmentally friendly [9]. As workloads and renewable resources vary among different DCs, workload migration has been considered a promising solution to reduce BE consumption [10].

Services in the cloud are delivered via virtualized servers known as virtual machines (VMs). Considering the time-varying workloads of end users, it is possible to maximize server resource utilization via VM migration [11]. To be more specific, VM migration provides the opportunity to move VMs to DCs based on the availability of RE [12]. In this work, we explore the problem of migrating VMs from DCs that have insufficient RE resources to those that have sufficient RE. The goal is to continue the migration process until the energy consumption of workloads in source DCs matches their corresponding available RE. The migration process is conducted over a dynamic network, which introduces extra complexity to the problem. The objective of this research is to develop an effective strategy for VM migration that ensures optimal utilization of renewable energy resources in DCs and reduces the total BE cost consumption. However, VM migration is not a trivial task for the following reasons. First, multiple DCs with sufficient RE may exist at the time of migration to host the

VMs. It is critical to choose the one that implies the lowest cost at the end. Second, migration can incur additional costs as a result of using optical network resources for migration as well as the time required to migrate a specific VM through the network. During the migration, network resources are reserved by the VM migration process. Third, the source DC has no prior knowledge of the dynamic network. Therefore, we first need to select an appropriate destination DCs under the network dynamics, and then migrate the VM while minimizing the additional costs.

Selecting a destination DC is an instance of the online decision-making problem. Hence, we formulate the DC selection as a multi-armed bandit (MAB) problem. Essentially, MAB is a classic problem that a gambler plays with multiple arms of slot machines in a sequence of trials so as to maximize his reward or minimize his cost [13]. Therefore, we define an arm as a DC that will be chosen to host the migrated VM, and we aim to minimize the overall cost, which comprises three components: (i) BE consumption cost incurred by the DCs; (ii) cost of optical network devices used for migration; and (iii) migration cost, which is the cost associated with the migration time or transmission delay of VMs. To that end, by selecting each arm and migrating a VM, the migration cost, which is a function of a random variable, is revealed. These data are collected by the algorithm in order to improve the arm selection in each round.

It is noteworthy to mention that the demand for cloud services and the need to establish efficient communication between DCs are driving the development of elastic optical networks (EONs). One of the benefits of EONs is the ability to use bandwidth-variable devices such as bandwidth-variable transponders [14]. Transponders are optical devices that convert the signal from the client of the optical network, e.g., an IP router, into C band signals (1530–1565 nm wavelength) for transmission across an optical network [15]. The client optical signal is generally carried at the 1310 nm wavelength, which is used in single-mode fiber optics. On the other hand, the optical network works in the C band, as it has the lowest fiber loss and attenuation in comparison to the S and L bands. Later, sliceable bandwidth variable transponders (SBVTs) were introduced to improve infrastructure flexibility and utilization efficiency of EONs [16]. An SBVT can allocate its capacity to one or more independent optical flows that are sent to one or more destinations. As a result, if an SBVT is used for a low-bitrate channel, the remaining capacity can be used to transmit a separate flow. This feature realizes optical traffic grooming, which can save cost and energy in EONs [16]. Without the use of SBVT, separate transponders would be

needed for each optical flow, resulting in higher costs for optical devices. Therefore, the use of SBVT and applying optical grooming can help reduce these costs. Since the migration between DCs requires a significant amount of bandwidth, SBVTs can play a critical role in reducing the number of transponders used for migration, thereby lowering the network cost.

The reconfigurable optical add/drop multiplexer (ROADM) is an important component in the EON system that is also considered in this work. ROADMs are a type of optical switch that have access to all wavelengths on a fiber and enable specific wavelengths to be dropped or added at a site while allowing other wavelengths to optically pass through the site without being terminated [5]. Using optical bypasses in the architecture helps reduce optical-electrical-optical (OEO) conversions. The OEO conversion refers to the process of converting an optical signal to an electrical signal, processing it in the electrical domain, and then converting it back to an optical signal for transmission. This process can be expensive in terms of cost, power consumption, and signal degradation [5]. Optical bypasses, on the other hand, allow the optical signal to bypass some or all of the OEO conversions, reducing the cost and power consumption associated with these conversions. In addition, using optical grooming can help reduce the number of ports required in ROADMs. By combining multiple lower-rate optical signals into a single higher-rate signal, the overall traffic load on the network is reduced, which in turn reduces the number of ports required in ROADMs. This can lead to a more cost-effective and efficient use of optical network resources. In the context of DC network architecture, using optical bypasses can reduce the cost and improve the performance of optical network devices used for migration. This can ultimately lead to more efficient and cost-effective inter-DC migration solutions.

1.1 Main Contributions

In summary, our contributions are listed as follows.

- We investigate the problem of inter-DC VM migration over a dynamic EONs network, aiming to minimize the total cost that comprises BE consumption costs, optical network device costs, and migration costs. To tackle this problem, we formulate it as an online decision-making problem.

- We introduce a novel approach to the DC selection problem by formulating it as an MAB problem. In this innovative formulation, we adopt a two-stage

MAB framework. In the first stage, each arm represents a potential destination DC, allowing us to dynamically select the optimal target. In the second stage, arms correspond to various network paths connecting the source to the chosen destination DC.

- We present the Modified Sliding-Window Lower Confidence Bound (MSW-LCB) algorithm to solve the problem while minimizing the regret[1];
- We contribute to the reduction of the cost of the optical network by applying the optical grooming algorithm to decrease the number of transponders and ROADM's ports used for migration requests. Additionally, we utilize ROADMs to minimize the OEO conversion, further reducing the cost of the optical network.
- We compare the performance of MSW-LCB with SW-LCB and other existing MAB algorithms, including ϵ-Greedy and Knapsack-based upper confidence bound (KUBE). The evaluation results indicate that MSW-LCB outperforms existing approaches in terms of total cost and regret.
- We conduct a simulation using real-world datasets, and we were able to assess the algorithm's performance and enable the validation and evaluation of the proposed algorithm in a semi-realistic setting.

1.2 Thesis Overview

The rest of the thesis is organized as follows. Chapter 2 provides background information and related works. In Chapter 3, we present the network model and migration algorithms, and formulate the online DC selection problem based on the MAB. In Chapter 4, we conduct extensive numerical evaluation and implement the simulation using real-world datasets to validate our algorithm using real-world data. Finally, Chapter 5 concludes the thesis.

1.3 Accepted Research Paper

This thesis is grounded in a foundational research project that received acceptance at the 2023 IEEE Global Communications Conference on Green Communications

[1]Regret is a performance loss between the learning algorithm and the best action in hindsight.

Systems and Networks symposium [17]. As the lead author, I collaborated closely with my supervisor and fellow lab members to execute the research and successfully publish the findings.

Our journey began with tackling the DC selection problem, aiming to reduce the BE consumption cost. Employing the classical MAB formulation, we developed the SW-LCB algorithm, a novel approach to this challenge. This method yielded exceptional results, as showcased through an extensive evaluation where SW-LCB consistently outperformed both the ϵ-Greedy and KUBE algorithms. By focusing on migration cost reduction, total cost minimization, and regret reduction, SW-LCB demonstrated its superiority.

Expanding on these accomplishments and incorporating insights from both internal and external feedback, we have further developed our work in an expanded section of the thesis. With the proposal of the MSW-LCB algorithm, we addressed the DC selection problem in a novel two-stage MAB framework. This approach allows us to effectively decrease the migration cost, and BE consumption cost, achieving a superior regret compared to our previous work. Furthermore, in response to feedback and to enhance the authenticity of our experiments, we harnessed a diverse range of real-world datasets, which closely simulate practical scenarios. Given the complexity of our context, which involves multiple interconnected DCs, real-world experimentation proved challenging. However, our approach with real-world datasets provided valuable insights, validating the feasibility and effectiveness of our proposed solutions in a semi-realistic environment.

1.4 Author's Research Background

My pursuit of a second master's degree in computer science, with a focus on virtualization, VM migration, and machine learning (ML) in cloud environments, is a strategic and purposeful step that builds upon the foundation of my previous academic achievements. While my initial master's degree concentrated on the physical layer of optical networks in the cloud, my current educational endeavor delves into the upper layers of the network ecosystem. By transitioning my expertise to encompass the intricate realm of ML-based migration, I am expanding my knowledge to encompass a broader and more comprehensive understanding of cloud computing.

This progression represents a logical evolution in my academic journey, allowing me to bridge the gap between the physical infrastructure and the software-driven

aspects of cloud networks. This new field of study complements and enhances my prior expertise, enabling me to address a more holistic range of challenges in the cloud computing domain. With a solid foundation in telecommunications systems and optical networks, my second master's degree equips me with the versatile skill set needed to tackle complex issues related to virtualization, resource allocation, and network optimization in cloud environments.

In essence, my choice to pursue a second master's degree is not merely repetitive; instead, it is a deliberate effort to enhance, contribute significantly, and broaden my expertise in a complementary direction. By combining and contributing my existing knowledge with an in-depth exploration of VM migration and cloud network management, I am uniquely positioned to make significant contributions to the evolving landscape of cloud computing.

Chapter 2

Background and Related Works

2.1 Cloud Networks

The advancement of technology has transformed communication from point-to-point to user-to-DC interaction. Cloud networks, a global array of interconnected DCs, create a unified ecosystem that functions like a massive interconnected computer system. Cloud networking provides the delivery of computing resources over the Internet. These resources can include networking equipment, computing and storage capabilities, and more. Rather than having to build and maintain your own physical infrastructure, the cloud allows you to access these resources on-demand and pay only for what you use. In addition to on-demand and pay-as-you-go benefits, the cloud has other important advantages such as cost savings, scalability, reliability, flexibility, security, and disaster recovery.

Cloud networks come in three models. They can be public, private, or hybrid, depending on how they are deployed and managed. Public cloud networks are hosted by third-party providers, such as Amazon Web Services (AWS), Microsoft Azure, or Google Cloud Platform. These networks are accessible to anyone with an Internet connection and provide a range of services that can be used by businesses and individuals. Private cloud networks are hosted by organizations themselves, either on-premises or in a private DC. These networks are used to deliver cloud services within the organization and are not accessible to the public. Hybrid cloud networks are a combination of public and private cloud networks, allowing organizations to leverage the benefits of both. For example, an organization might use a private cloud network for sensitive data and applications, while using a public cloud network for

less sensitive workloads.

A pivotal aspect worth highlighting is that cloud computing is fundamentally built upon virtualization technology. Virtualization allows multiple VMs to run on a single physical server. In the next section, we discuss virtualization in more detail.

2.1.1 Virtualization

Virtualization is the process of creating virtual counterparts of various components, such as VMs, virtual storage devices, and virtual networks. It involves abstracting computing resources such as the CPU, memory, storage, and network from their physical hardware and presenting them as independent entities that can be managed and allocated dynamically. This allows multiple virtual resources to run on a single physical machine, improving resource utilization, scalability, and flexibility. Users can quickly and easily provision and deprovision VMs as needed. Furthermore, virtualization provides a layer of abstraction between the physical hardware and the VMs.

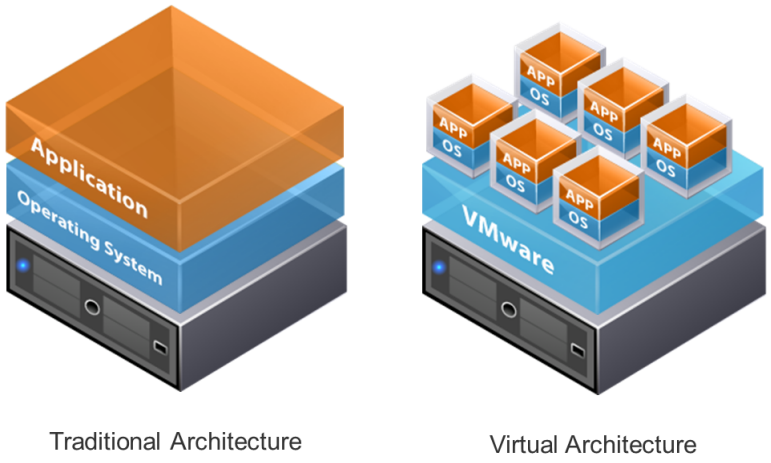

Figure 2.1: Traditional vs. virtualized architecture [1].

A common example of using virtualization in cloud networks is a company that runs multiple applications on a single server using VMs. Each VM is isolated from the others, allowing for easy management and maintenance. The company can easily add or remove VMs as needed to meet changing business needs, and it can also scale resources up or down without having to purchase additional hardware. When the company needs to relocate its VMs to a different physical server, it requires a process

known as VM migration.

2.1.2 VM Migration

VM migration is the process of moving a VM from one physical host to another without disrupting the VM's operation. This is done by transferring the VM's memory, storage, and network connectivity from the source host to the destination host. In the following, we will discuss the various phases of a VM migration. Fig. 2.2 shows the VM migration from physical machine (PM) 1 to PM 2. Both PMs can be in the same DC (intra-DC) or two geographically different DCs (inter-DC).

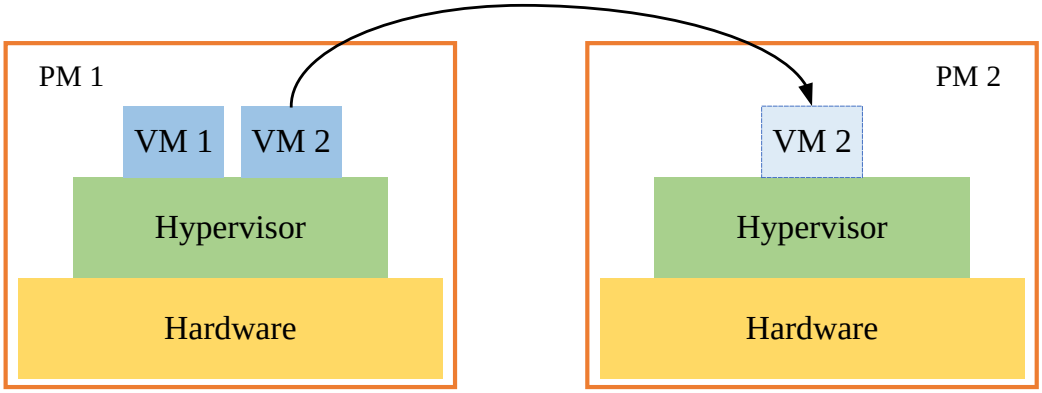

Figure 2.2: Virtual machine migration.

VM migration is a powerful tool that can be used for a variety of purposes. In intra and inter cloud management, it plays a critical role in enabling seamless workload transfers between cloud environments [18]. Specifically, VM migration enables zero-downtime hardware maintenance, load balancing, server consolidation, across-site management, hybrid cloud capabilities, cloud federation, and adaptability to user mobility or demand changes. It enables uninterrupted operations during hardware maintenance, optimizes resource allocation, reduces costs, supports disaster recovery, facilitates hybrid and federated cloud deployments, and allows for dynamic response to user needs.

2.1.2.1 Different Migration Types

There are two types of VM migrations: live (hot) and cold migrations. Hereunder, we explained each of them in detail [19].

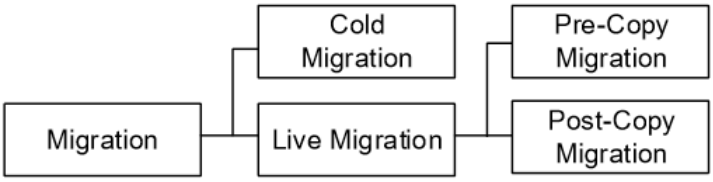

Figure 2.3: Categories of migration types [2].

1. In live migration, also known as hot migration, the VM is migrated to the destination host without any downtime. The VM state is transferred from the source host to the destination host over the network while the VM is running. Live migration is further divided into two types:

   - Pre-copy live migration: In pre-copy live migration, the VM memory is transferred to the destination host in small chunks. The VM is paused briefly to transfer the final delta memory page, which is the difference between the initial and final memory states.
   - Post-copy live migration: In post-copy live migration, the VM is migrated to the destination host with minimal downtime, and then the remaining memory pages are transferred to the destination host after the migration. The VM continues to run on the source host while the remaining memory pages are being transferred to the destination host.

2. In non-live migration, also known as cold migration, the VM is powered off before it is migrated to the destination host. The VM's files are copied to the destination host, and then the VM is started on the destination host. This type of migration requires more downtime compared to live migration.

The choice between live and cold migration depends on the specific use case and requirements. Live migration is preferred for minimal disruption and maintaining the VM's state, while non-live migration is suitable when downtime is acceptable or technical limitations prevent live migration. Non-live migration has a simpler process and consumes fewer resources. Ultimately, the decision should be based on the specific needs and constraints of the migration scenario.

2.1.2.2 Migration Network Environments

Migration can happen in different spans, such as inter-DC or intra-DC. Understanding the migration span is crucial as it directly impacts the management of the migration [2]. This thesis work primarily focuses on the inter-DC environment, and as such, the discussion of the other environments is beyond the scope of this work.

Inter-DC migration refers to the process of migrating VMs between different DCs. Live migration is widely adopted for inter-DC management due to its various benefits, including cost management, emergency recovery, energy consumption optimization, performance and latency improvement, reduction in data transmission, and regulatory compliance based on administrative domains. In an inter-DC environment, the network connecting the DCs is an optical network. The optical network uses light to transmit data over optical fibers, providing high bandwidth and low-latency communication between DCs. In the next section, we will explore the optical networks in more detail.

2.2 Optical Networks

With the annual growth of Internet traffic, the demand for bandwidth has become a key concern for telecommunications companies and analysts. Optical networks play a vital role in supporting the communication infrastructure of cable networks, mobile networks, and the Internet, offering a multitude of advantages. These networks provide high transfer speeds, ample bandwidth, cost-effectiveness compared to copper wires, and minimal signal attenuation. Additionally, optical networks are popular for their reliability, security, immunity to interference, and leveraging of advanced encryption technologies.

As mentioned, high-capacity fiber-optic networks have been deployed to ensure the delivery of traffic and the Quality of Services. In a cloud network, relatively large amounts of data are frequently exchanged between DCs, and these traffic patterns vary over time. In such situations, using static and inflexible connections will only increase costs because high connection capacity must be considered for peak hours, while this capacity will remain unused at other times. It is clear that a new technology is needed to connect DCs that are adjustable and compatible with the transmission network, while also reducing costs and enhancing resource efficiency.

2.2.1 Elastic Optical Networks

Modern optical networks, known as EONs, provide dynamic bandwidth allocation and adaptation to changing traffic demands. By utilizing advanced modulation techniques and flexible grid systems, EON optimizes the allocation of bandwidth with greater efficiency and granularity. These networks can transmit data over long distances without signal degradation and support high-speed communication and large data transfers. EONs play a crucial role in cloud computing and big data analytics, enabling fast and efficient data transfers between DCs.

2.2.2 Two Main Elements of EONs

While the EON comprises various optical components, this thesis will specifically focus on two of the main components, namely ROADMs and SBVTs.

2.2.2.1 ROADMs

ROADMs are a key component of optical networks, particularly in EONs. A ROADM allows any wavelength to be added/dropped at any node, and the choice of add/drop wavelengths can be readily changed without impacting any of the other connections terminating at or transiting the node. Furthermore, a ROADM can be remotely configured through software as opposed to requiring manual intervention [5]. This enables greater flexibility and efficiency in managing network traffic and allows for rapid reconfiguration of the network connections in response to changing demands. A ROADM typically consists of a series of optical switches, amplifiers, and filters that can be electronically controlled to dynamically route and manage optical traffic. Fig. 2.4 shows a simple ROADM node architecture. It is constructed by WSSs and amplifiers. Wavelength-selective is a term used to classify devices that are capable of treating each wavelength differently [5]. For instance, $1 \times N$ WSS is a switch that can direct any wavelength on the one input port to any of the N output ports [5], which serves as a demultiplexer. As it is shown in Fig. 2.4, the transiting traffic is bypassing the node without converting into the electrical domain. This will save a huge amount of cost and energy.

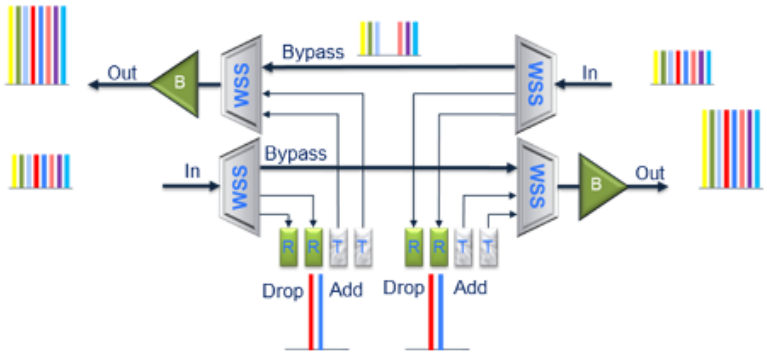

Figure 2.4: Degree-two ROADM architecture [3].

2.2.2.2 SBVTs

In the context of optical communication networks, a transponder typically refers to a device that converts the wavelength of the incoming optical signal into an electrical signal, performs some processing and amplifying, and then converts it back into an optical signal in the C band for transmission over the optical networks. The C band is chosen because of the relatively low fiber attenuation, and its spectrum is from 1530 nm to 1565 nm.

Flex or bandwidth variable transponders (BVTs) are key components in EONs that dynamically adjust their transmission rate and bandwidth allocation based on network traffic demands. They offer flexibility in resource allocation and optimize bandwidth utilization. BVTs are particularly important in wavelength division multiplexing (WDM) systems, where multiple optical signals are transmitted simultaneously over different wavelengths. By employing BVTs at the transmitting and receiving ends of optical channels, bandwidth allocation can be dynamically adjusted to meet varying traffic requirements. For instance, during high-demand periods, BVTs can allocate wider bandwidth to specific channels, enabling higher data rates and maintaining the Quality of Service [4].

However, BVTs do not provide fine-grained control over the spectrum, and the bandwidth adjustment is limited to the entire channel. Moreover, if the capacity of a BVT is not fully utilized due to various reasons, such as disturbances in the optical path, then a portion of its capacity is wasted. To solve these problems, SBVTs, which are the advanced version of BVTs and offer more granular control over the spectrum, were introduced. Fig. 2.5 shows the difference between BVTs and SBVTs. SBVTs can allocate their capacity to one or more independent optical lightpaths that

are transmitted to one or more destinations. This flexibility helps optimize network performance, increase capacity, and decrease the number of transponders. When an SBVT is used for a channel with a low bit rate, the remaining capacity can be used to transmit another independent optical lightpath. This feature leads to realizing optical traffic grooming, which we will discuss in the next subsection in detail.

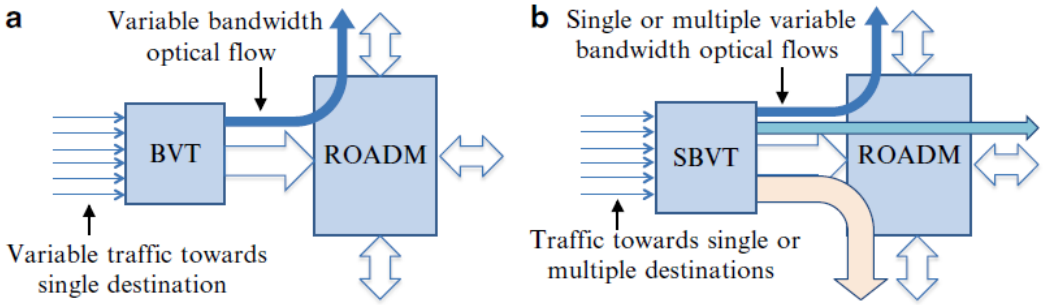

Figure 2.5: Schematics of (a) bandwidth variable transponder (BVT) and (b) sliceable bandwidth variable transponder (SBVT) [4].

Basically, from the higher-layer perspective, an SBVT can act like a high-capacity BVT or multiple independent BVTs with low capacity, depending on the application. Hence, the use of SBVTs in the network leads to reducing the number of transponders needed, as each sub-carrier in an SBVT has the capability to be sent to different destinations. The number of destinations supported by each SBVT depends on its granularity. For example, an SBVT with a capacity of 400 Gbps using 40 Gbps granularity can support up to 10 destinations, but with a granularity of 100 Gbps, it can only support up to 4 destinations [4].

2.2.3 Traffic Grooming

Considering all the advancements in networking technology, a large portion of traffic is operating at service rates lower than a full wavelength. While backbone networks often support high-speed wavelengths of 40 or 100 Gb/s, the majority of demands require rates of 10 Gb/s or lower. Henceforth, we will refer to demands at a lower bit rate as subrate traffic. There are various methods available for accommodating subrate traffic. However, accommodating this type of traffic in network environments requires efficient methods to avoid wasting capacity. Assigning a full wavelength to carry a subrate demand leads to significant inefficiency, with examples like using a

100 Gb/s wavelength for a 10 Gb/s demand, wasting 90% of the capacity, which is significantly inefficient [5].

The preferred approach is to use traffic grooming. Traffic grooming refers to the process of efficiently aggregating low-capacity traffic streams into higher-capacity wavelengths or lightpaths. It involves combining multiple smaller streams of data into a larger stream, optimizing the use of network resources. The purpose of traffic grooming is to improve the utilization of the network infrastructure, reduce the number of required optical components, and minimize the overall network cost [20]. There are different types of traffic grooming techniques employed in optical networks, as follows:

1. **End-to-End Multiplexing:** This type of grooming involves consolidating subrate traffic demands with identical sources and destinations onto the same wavelengths. This grouping of subrate demands allows them to be treated collectively as a single demand. In Fig. 2.6, an example of end-to-end multiplexing is depicted, where a line rate of 40 Gb/s is assumed. For instance, a single wavelength accommodates the two 10 Gb/s connections between Nodes A and H, resulting in a utilization rate of 50%.

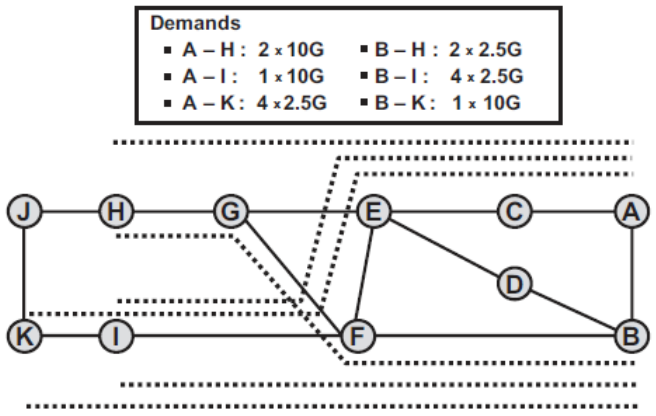

Figure 2.6: Example of end-to-end multiplexing [5].

However, there are certain drawbacks associated with this type of grooming. Firstly, ensuring compatibility among individual subrate demands becomes crucial. If different demands have conflicting requirements, such as avoiding specific network links, finding a suitable path that satisfies all demands can be challenging. Secondly, the efficiency of multiplexing depends on the volume of traffic

between node pairs relative to the line rate. Low traffic levels can result in underutilized wavelengths [5].

2. **Electrical Grooming:** Unlike end-to-end multiplexing, electrical grooming enables the reconfiguration of wavelengths at intermediate nodes rather than just at the source and destination. The aim of grooming is to maximize the utilization of wavelengths between specific grooming sites, rather than solely considering the source and destination of the subrate demands. Fig. 2.7 shows an example of electrical grooming. Regardless of their ultimate destinations, all demands from Node A are consolidated onto a single wavelength for transmission to Node E. Similarly, Node B's demands are combined onto another wavelength destined for Node E. At Node E, grooming equipment is deployed to separate and reassemble the wavelengths according to different groupings. Consequently, a single wavelength generated by Node E efficiently carries all demands destined for Node H, regardless of their original sources [5].

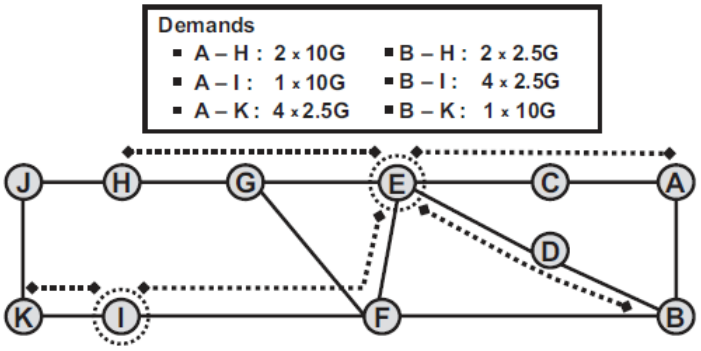

Figure 2.7: Example of electrical grooming [5].

Different architectural implementations exist for electrical grooming, which can occur at various layers, such as the IP layer or the optical transport network (OTN) layer, using a grooming switch. OTN switches offer cost and power advantages over IP routers, consuming approximately 23% less power per bps [21]. Additionally, ROADM technology provides significant cost and power savings. A ROADM consumes about half of the power per bps compared to an OTN switch [22]. Therefore, whenever possible, bypassing the IP and OTN layers is preferred, leading to increased capacity requirements.

3. **Optical Grooming:** Optical grooming refers to the process of aggregating and

manipulating subrate traffic in optical networks. It has been introduced as a method to conserve energy in EONs. This type of grooming is performed at the optical layer without the involvement of electrical processing at intermediate nodes. It is achieved through the use of SBVTs or multi-flow transponders. Multiple optical paths with different destinations can be routed through a single SBVT, reducing the number of active transponders in the network. As discussed in Section 2.2.2, one of the advantages of SBVT is that each optical path is independent of other optical paths and has its own dedicated wavelength and modulation format. As it is shown in Fig. 2.8, traffic from higher layers enters the optical network through routers. An SBVT can be divided into multiple sub-transponders, each having its own independent wavelength, modulation format, and demodulation. Ultimately, the groomed traffic is transmitted through a single optical path. As depicted in the figure, intermediate nodes also have the capability to rearrange the traffic. This means that traffic can be offloaded at intermediate nodes, and new traffic can be added to the network.

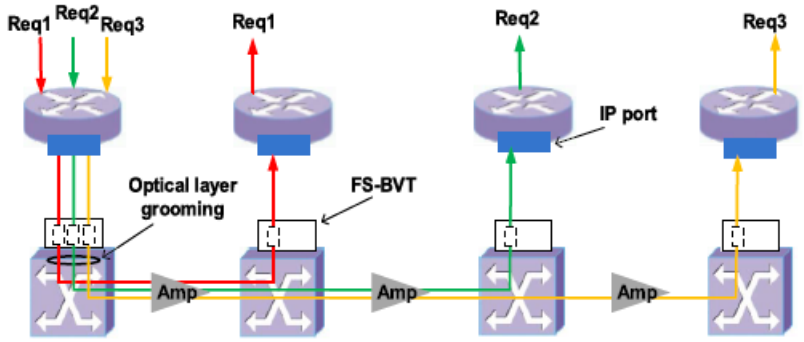

Figure 2.8: Example of optical grooming [6].

2.3 Learning Algorithms

Learning algorithms are computational procedures or methods designed to enable computers or machines to learn from data and improve their performance on specific tasks. These algorithms are the core components of machine learning. Learning algorithms are used to train models or systems by analyzing and processing large amounts of data. They discover patterns, relationships, and insights within the data, and use this information to make predictions, classify new data, or make decisions. The goal

of learning algorithms is to enable machines to automatically learn and adapt without being explicitly programmed for every specific task [23]. Learning algorithms employ mathematical and statistical techniques to optimize their models and make accurate predictions or decisions. They often involve processes such as feature selection, model training, evaluation, and validation. Different learning algorithms have different strengths and are suited for specific types of problems or data.

There are various types of learning algorithms, including supervised learning, unsupervised learning, and reinforcement learning. In this work, we focus solely on reinforcement learning, and although supervised and unsupervised learning are valuable approaches, they are not directly applicable to this study and are beyond the scope of our research. In supervised learning, a model learns from labeled examples to make predictions or classifications, while in unsupervised learning, the model identifies patterns in unlabeled data. In contrast, reinforcement learning algorithms make decisions by interacting with an environment and receiving feedback or rewards for their actions [24]. Therefore, reinforcement learning is chosen in this study due to its suitability for our research objectives.

2.3.1 Online Learning

Online learning, also known as incremental or sequential learning, is a machine learning paradigm where the learning algorithm receives data instances one at a time and updates its model incrementally with each new observation. In general, it is the process of taking actions based on the given knowledge of the rewards of the previous actions and available information [25]. This allows the model to adapt and respond quickly to changes in the data distribution or incoming data points. The model is updated iteratively, and previous observations may be forgotten or given less weight over time, depending on the algorithm's design. Online learning is well-suited for scenarios where data arrives continuously or in streams, and where it is not feasible or efficient to store and process the entire dataset. It is often used in real-time applications, where decisions or predictions need to be made quickly based on the most recent data.

In the context of VM migration in an inter-DC scenario, online learning is a suitable approach due to its continuous adaptation to changing network conditions and resource availability. It can handle large-scale data by processing it incrementally, making it scalable for managing VM migrations. Online learning enables real-time

decision-making, allowing the system to quickly react to workload patterns and optimize migration decisions.

2.3.2 Multi-Armed Bandit

The multi-armed bandit is a reinforcement learning framework that facilitates online decision-making. It involves a situation in which an agent is presented with a series of options, typically referred to as "arms", and must choose which arm to use in order to maximize a cumulative reward over time. Each arm represents a different action or strategy, and the agent's goal is to learn which arm provides the highest reward through a series of sequential interactions. Multi-armed bandit algorithms can be considered as an online learning algorithm as they operate in an online setting where decisions are made sequentially based on limited information. The agent explores different arms to gather data and exploit the arms that appear to be more rewarding. Over time, the algorithm adapts its decision-making strategy to make more informed choices, leveraging the acquired knowledge.

Leveraging MAB techniques in the context of Inter-DC migration offers a powerful advantage by embracing network dynamicity. This is achieved through the integration of a probabilistic model that encompasses fluctuations in network link delays and variations in DCs' traffic patterns. This model continually evolves as historical data is collected, enabling more accurate cost estimations.

By harnessing MAB algorithms, the VM migration system gains the ability to dynamically adapt to shifting network conditions. Over time, it learns and selects the optimal destination DC intelligently. Consequently, this adaptive approach translates into tangible benefits such as reduced migration times and more cost-effective network communication, outperforming traditional methods reliant on static policies or heuristics.

2.4 Related Works

The progress in VM and container migration has been truly remarkable so far. Although the majority of the migration focused on intra-DC migration, a few papers addressed inter-DC migration. In [26], the performance of parallel and sequential live migrations in cloud DCs was investigated. The results showed that using parallel migration can reduce VM downtime during relocation. Ayoub et al. in [27] used on-

line inter-DC VM migration to avoid service outages in the event of a disaster. They formulated the VM migration as an integer linear programming (ILP) problem and proposed heuristic algorithms to increase the number of migrated VMs while decreasing network resource utilization. Moreover, Teyeb et al. in [28] addressed the placement of VMs in a distributed cloud environment while taking into account location and system performance constraints. They improved the utilization of computational resources by developing an ILP formulation. In addition, many approaches have been presented to reduce the energy consumption of DCs in particular [29]. Zhang et al. in [30] used VM migration among DCs based on RE availability. After presenting it as a many-manycast communication problem, they developed an ILP formulation. They also proposed a heuristic algorithm to solve this problem by splitting many-manycast into many anycast communications, namely Anycast algorithm with joint resources ergodic (Anycast-JRE) routing. Moreover, in [31], VM placement in both inter and intra-DC networks is used to reduce the energy consumption of all DCs and optical network components.

Aside from that, the MAB approach has been used to address VM migration and placement in a variety of scenarios. End-to-end delay minimization in mobile edge computing is addressed in [25] by using contextual MAB while taking the service migration cost and user mobility into account. Furthermore, the authors in [32] implemented a budget-limited MAB problem to solve the server selection problem in edge computing. They defined reward and cost in terms of the amount of time and energy required for each offloading round from a user's device to a server. They also demonstrated that their algorithm could solve the problem efficiently in a dynamic environment. As previously stated, we use the MAB framework to represent the DC selection problem. Our strategy is more similar to [32]. However, in our work, the network's dynamic nature is taken into account while migrating VMs over EON infrastructure. We define the cost in terms of BE consumption, optical network elements, and migration costs, with the intention of lowering them through implementing lower confidence bounds and optical grooming algorithms.

Numerous research studies have underscored the substantial advantages offered by EONs as a foundational infrastructure for the Internet. Lohani et al. delved into the different elements and capabilities that EON introduces to the next generation of networking and high-speed communication in their work [33]. Furthermore, EON has been hailed as an efficient and cost-effective solution for provisioning cloud computing traffic. In a comprehensive analysis, researchers in [34] evaluated EON based

on various critical parameters, including deployment cost, energy consumption, and bandwidth utilization over the US backbone network. Their findings demonstrated that EON had a significant positive impact on all these crucial criteria when compared to alternative solutions.

Furthermore, the utilization of traffic grooming emerges as a highly effective technique for enhancing wavelength and spectrum efficiency within EONs [35]. In a compelling study, researchers in [6] demonstrated that the integration of SBVT and optical traffic grooming can yield substantial reductions in network power consumption. This accomplishment caught the interest of others in the area, with Zhu et al. later concentrating on energy consumption reduction in the context of network virtualization in their work [36]. Moreover, the cost of network elements has been a persistent concern, which was addressed in [14] through the application of optical grooming techniques. The authors presented an ILP formulation that effectively mitigated transponder costs, resulting in an impressive reduction of approximately 30%. In summary, numerous research studies have introduced traffic grooming as an effective strategy to minimize the number of transceivers and subsequently lower network costs, as evidenced by notable contributions in [37] and [38]. Recognizing the compelling potential of this technique, we have incorporated optical traffic grooming into our research approach. This decision aligns with our objective to further optimize the cost associated with inter-DC VM migration.

Chapter 3

Inter-DC VM Migration Using MAB

In this chapter, we introduce our proposed algorithm designed for VM migration in an inter-DC environment. We begin by outlining the network model and the problem formulation, which is grounded in the MAB framework. Subsequently, we present the MSW-LCB algorithm, a solution aimed at minimizing the total cost, which includes the costs of BE consumption, optical network devices, and migration, while minimizing the regret.

Leveraging MAB in the context of VM migration presents a compelling array of advantages. Firstly, MAB algorithms excel in adaptability, efficiently responding to dynamic conditions in DCs and the network. This adaptability enables them to optimize migration strategies as network traffic, server workloads, and DC conditions evolve over time. Secondly, MAB algorithms are inherently designed for optimization. They strike an optimal balance between exploration (trying various options) and exploitation (selecting the best-known option), a crucial aspect in VM migration. By using MABs, VM migrations can significantly reduce resource consumption, downtime, and meet Quality of Service requirements more effectively. Thirdly, these algorithms are skilled at handling the dynamic feature of the network, which is a factor in VM migrations due to fluctuating network delays and workload changes in DCs. MAB algorithms continuously update their decision strategies based on observed outcomes, enhancing the decision-making process. Additionally, they support real-time decision-making, enabling quick responses to changing conditions, ultimately preventing bottlenecks and service disruptions. In essence, MAB algorithms empower

VM migration processes with data-driven, adaptable, and optimized decision-making capabilities, enhancing DC efficiency and ensuring a smoother migration experience.

3.1 Network Model

The inter-DC network is modeled as a graph $G(N, L)$, where N is the set of DCs as nodes and L is the set of links between nodes in the graph. We assumed that all DCs in the network are equipped with hybrid energy resources and that each of the DCs processes its own workloads. If the energy consumption of a DC is greater than the available RE of the corresponding DC, we need to migrate VMs to diminish the mismatch. We assume each DC has a different amount of available RE. The reason for this is that certain DCs may be geographically near RE sites and hence employ more RE resources. Moreover, it is worth mentioning that DCs draw power from the grid, and the consumption from the grid cannot be separated as renewable or brown, so it is always a mixture of both. However, it is possible to estimate the contribution of RE and BE to the grid in different ways. One way is by utilizing the grid data. Some electricity grid operators provide data on the mix of energy sources they use to generate electricity. This data might include the percentage of energy generated from sources like coal, natural gas, wind, solar, hydro, etc. By using this information, it is possible to estimate the RE and BE contributions. In addition, some DCs have their own RE generation, such as solar panels or wind turbines, and the energy generation can be directly measured by these sources.

Servers' power consumption constitutes a substantial portion of a DC's overall energy usage. Therefore, this thesis focuses on the server power consumption within DCs, which consists of static and dynamic power consumption. The former includes server idle power consumption and the additional power to support the server's operation in DC. While the latter varies depending on the number of workloads being processed. In this work, we mainly focus on reducing the dynamic portion of the server's power consumption. The power consumption of server r in DC s is defined as

$$P_{r,s} = P_{r,s}^{\text{Static}} + P_{r,s}^{\text{Dyna}}, \tag{3.1}$$

where $P_{r,s}^{\text{Static}}$ and $P_{r,s}^{\text{Dyna}}$ are the static and dynamic power consumption of the server,

respectively. They are defined by

$$P_{\text{Static}} = P_{\text{Idle}} + (\eta - 1) \cdot P_{\text{Peak}},$$

$$P_{\text{Dyna}} = (P_{\text{Peak}} - P_{\text{Idle}}) \cdot u_{r,s},$$

where P_{Idle} and P_{Peak} are the server's idle and peak power consumption [30]. It is also worth emphasizing that $P_{r,s}^{\text{Static}}$ contains a portion of P_{Peak} because even an idle server consumes some percent of its peak power due to electricity, cooling, and to support the server's operation in a DC environment. P_{Idle} is the power consumption of the server when it is in an idle state, meaning it is powered on but not actively processing any workloads on VMs. The parameter η is the power usage effectiveness (PUE) of the DC [30], which is a metric that quantifies the energy efficiency of an DC. Additionally, $u_{r,s}$ represents the CPU utilization of the r-th server in the s-th DC, defined as

$$\text{CPU Utilization} = \frac{\text{The number of used CPU cores by requests}}{\text{The maximum number of CPU cores on the server}}.$$

Knowing the network model, in the following section, we delve into our proposed algorithm, which addresses the VM migration in an inter-DC scenario.

3.2 Proposed Algorithm

In our previous research work [17], we explored the use of the MAB algorithm for inter-DC VM migration. We proposed SW-LCB to decrease the total cost while having the lowest regret in arm selection in each round. While the SW-LCB algorithm provides a framework for sequential decision-making, it typically focuses on selecting a DC in each round and uses the shortest path to do the migration.

However, certain scenarios pose challenges of a different magnitude, where the decision-making complexity calls for a more sophisticated approach. These challenges include the need to enhance the DC selection problem by dynamically tracking their power consumption, ensuring the selection of a path that provides the lowest delay regardless of being the shortest path based on the length of the link, and also minimizing unnecessary migration costs. In response to these complexities, we've evolved our strategy. Instead of relying solely on a single MAB, we now employ a two-stage sequential MAB approach. This novel two-step MAB mechanism optimizes

the selection process in each round θ by first identifying the destination DC with the available RE and the lowest power consumption. Then, find a path with the lowest delay between the source and the selected destination. This refined methodology aims to enhance resource utilization efficiency, minimize delay, and meet the Quality of Service criteria, thus addressing the ever-evolving demands of inter-DC VM migration.

The objective is to reduce the total cost, which is described by (3.2) and is the sum of three expenses: the cost of BE consumption in DCs, denoted by C_{DC}, the cost of optical network elements defined by C_{Net}, and the migration cost, expressed by C_{Mig}.

$$C_{\text{Total}} = C_{\text{DC}} + C_{\text{Net}} + C_{\text{Mig}} \tag{3.2}$$

The cost of BE consumption in DCs is calculated as

$$C_{\text{DC}} = \sum_{s \in N} \sum_{\theta=1}^{\Theta} \alpha_s \Gamma_{s,\theta},$$

where α_s is the energy cost per unit for the s-th DC. In the formulation, the parameter Θ is the total number of rounds. Since the number of used CPU cores by requests can vary dynamically due to workload fluctuations, resource allocation, and user demand, we consider the CPU utilization as a random variable and redefine it as $u_{r,s,\theta}$. This notation shows the CPU utilization of the server r in s-th DC in round θ. Consequently, the power consumption of server r in (3.1) can be redefined as $P_{r,s,\theta}$. By expressing the s-th DC power consumption as $P_{s,\theta} = \sum_r P_{r,s,\theta}$, and the available RE in the s-th DC by R_s, we can now define the BE power consumption in DC s in round θ as

$$\Gamma_{s,\theta} = \max\{P_{s,\theta} - R_s, 0\}. \tag{3.3}$$

Based on (3.3), if $P_{s,\theta}$ is greater than R_s, the DC does not have enough RE and $P_{s,\theta} - R_s$ is the amount of the BE that the DC is consuming. If $P_{s,\theta}$ is lower than R_s, the BE consumption is considered zero.

Equation (3.4) defines the calculation of C_{Mig}. and a_θ is the path that is selected in round θ. The migration cost from DC s to the destination DC in round θ using path a_θ is denoted by c_θ^{s,a_θ}, and expressed as the amount of delay that a VM experiences

during the migration,

$$C_{\text{Mig}} = \sum_{s \in N} \sum_{\theta=1}^{\Theta} c_{\theta}^{s,a_\theta} = \sum_{s \in N} \sum_{\theta=1}^{\Theta} \rho \cdot d_{\theta}^{s,a_\theta}, \tag{3.4}$$

where ρ is the migration cost per unit associated with data transmission via the link. When selecting a migration path from the source DC s to the destination DC, a transmission delay is incurred during round θ, denoted as d_{θ}^{s,a_θ}. This delay encompasses various factors, including queueing, processing, and other latencies involved in the VM transfer process. The challenge lies in the fact that this transmission delay remains unknown at the beginning of each round, primarily due to the variable background traffic conditions in the underlying network. The longer the delay is, the greater the VM downtime will be. This downtime refers to the time that the VM is suspended as a result of the stop-and-copy, commitment and activation phases during the migration [2]. During this time, the service is unavailable to clients, which incurs a migration cost at the rate defined by ρ for each passing millisecond of unavailability.

For the calculation of the cost of optical network elements, we consider two important optical network devices, SBVTs and ROADMs, as follows:

$$C_{\text{Net}} = C_{\text{T}} + C_{\text{ROADM}} + C_{\text{Cap}}.$$

Where C_{T} is the SBVT's cost and is calculated as

$$C_{\text{T}} = \sum_{\theta=1}^{\Theta} \beta \cdot T_{\theta}.$$

Parameter β is the migration cost per unit associated with transponder consumption. Moreover, T_θ is the number of transponders used for the VM migration in round θ. The cost related to the ROADM can be calculated as

$$C_{\text{ROADM}} = \sum_{\theta=1}^{\Theta} (c_{\text{Amp}} + c_{\text{WSS}}). \tag{3.5}$$

As mentioned in the previous chapter, ROADMs are one type of optical switches, and they have built-in wavelength-selective switches (WSS) and amplifiers to direct and boost the optical signals. Therefore, the cost of amplifiers (c_{Amp}) and WSS switch (c_{WSS}) are considered in (3.5) [39]. Fig. 3.1 illustrates the node architecture in our

setting.

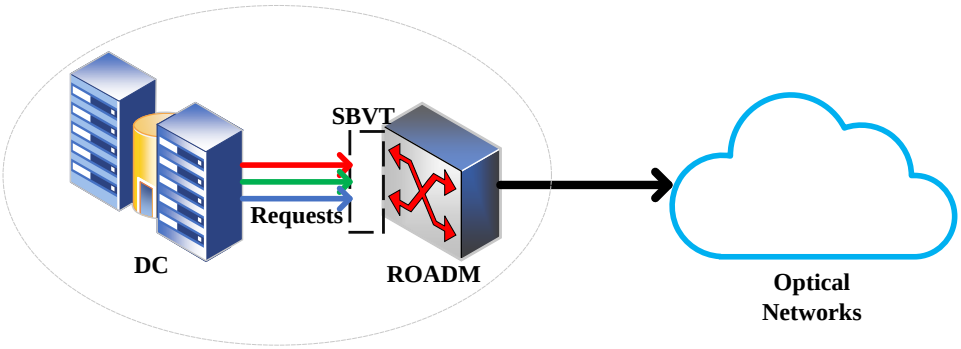

Figure 3.1: Node architecture.

The last term in the C_{Net} is the cost related to the Capital Expenditure (CapEx), and defined as

$$C_{\text{Cap}} = \text{Initial purchase cost of the device} \times \text{Number of purchases.}$$

We have considered this cost to reflect the fact that while some DCs might be equipped with the latest and most efficient devices, others may still rely on older and less capable equipment. Therefore, we include this cost in the formulation to ensure a seamless inter-DC migration so we can provide a more holistic view of the costs involved in our migration strategy. It is important to note that CapEx is only considered when a DC needs to acquire SBVTs and ROADMs. If a DC is already equipped with a sufficient number of components, it is not required to consider this portion of the cost.

Nevertheless, (3.2) is difficult to solve because the costs of migration and BE consumption cannot be predicted accurately beforehand. Therefore, the calculation is based on the sum of the time-varying costs, which is determined by the chosen destination DC and path in each round. Hence, we propose a new formulation. We define regret as the difference between the total cost calculated by an idealized entity, namely Oracle, with complete knowledge, and the total cost calculated by the MAB algorithm. Generally, in MAB, Oracle can choose the arm with the lowest expected cost, achieving the minimum possible cumulative cost over a given time horizon. We aim at minimizing cumulative regret, which is defined by

$$\text{Regret} = C_{\text{Total}} - C^*_{\text{Total}} = (C_{\text{DC}} - C^*_{\text{DC}}) + (C_{\text{Mig}} - C^*_{\text{Mig}}), \tag{3.6}$$

where C^*_{Total}, C^*_{DC}, and C^*_{Mig} are the total cost, BE consumption, and migration costs calculated by Oracle in round θ, respectively. Since the decision and output

of the oracle cannot be deterministically computed, we estimate them by getting the expectation of the random variable over its probability distribution. The reason for this consideration is the expectation is essentially the long-term average or the typical value of a random variable. It embodies a comprehensive understanding of the variable's behavior, encapsulating both its central tendency and dispersion. This makes it particularly suitable for estimating costs in scenarios where outcomes are inherently unpredictable.

3.2.1 MSW-LCB Algorithm

In this section, we propose the MSW-LCB algorithm, which is designed to minimize cumulative regret and is presented in Algorithm 1. When using MAB in two phases, we must specify separate arms for each phase. In the first phase, we define the arm as a destination DC that can host the VM, and in the second phase, an arm is a path from the source to the selected DC from the previous phase. In MSW-LCB, we employ a sliding window strategy to ensure that results are more focused on recent information and to promote faster convergence to the ideal arm based on historical information.

In our other work [17], we migrated the VM through the shortest path. This shortest path was calculated based on the length of the links in the topology. However, in today's traffic-varying network, propagation delay, which has a direct relationship with the length of the fiber optic link, is negligible. Transmission delay, on the other hand, can be significant because it is dependent on the unpredictable underlying network. Hence, in MSW-LCB, we do not necessarily transfer the VM through the shortest path. Instead, we consider a path from source to destination DC that has the lowest transmission delay. In this way, we can further reduce the migration time and cost.

As mentioned before, the transmission delay and workloads are random variables and are unknown at the beginning of each round. Therefore, the estimation of these parameters plays an important role in making informed decisions. To this end, we calculate the average of the random variable on the τ previous rounds. We formulate the average as a function f, given in (3.7) and (3.8). Using these definitions, we can use the equations for both random variables. In the first phase of MSW-LCB, where the focus lies on selecting the optimal destination DC, x is the selected arm or destination DC in round k. In the second phase, x is the path that its transmission

delay need to be estimated.

$$\bar{f}_\theta(x,\tau) = \frac{\sum_k^\theta w(k,\tau)\cdot f_k^x \cdot \mathbb{1}_{\{arm_k=x\}}}{\sum_k^\theta w(k,\tau)\cdot \mathbb{1}_{\{arm_k=x\}}}, \tag{3.7}$$

where $k = \max\{1,(\theta-\tau)+1\}$, and $w(k,\tau)$ represents the weight assigned to the arm x in round k, and we define it as an exponential function, defined as

$$w(k,\tau) = 1 - e^{-\frac{k}{\tau}}.$$

Hence, all element within the window τ is assigned a weight between $[0,1]$, with greater emphasis placed on recent arms relative to others. Also the adjustment term is calculated by

$$E_\theta(x,\tau) = \sqrt{\frac{\sigma \log(\min\{\theta,\tau\})}{\sum_k^\theta \mathbb{1}_{\{arm_k=x\}}}}. \tag{3.8}$$

σ is a pre-defined parameter, and the tunable parameter τ represents the length of the sliding window [40]. As a result, in order to make a decision on the current round, we need information from previous rounds. In this approach, instead of averaging over all previous observations, we only use the τ last rounds to estimate the transmission delay based on recent information.

Algorithm 1 shows the MSW-LCB algorithm. We assume that each request is expressed as a VM migration request, and each VM may have different CPU and bandwidth requirements. Additionally, we give special consideration to the Service Level Agreements (SLAs) associated with specific VMs. Some VMs have stringent requirements regarding downtime, ensuring that they should not be unavailable for more than a certain duration, often measured in minutes. This crucial aspect is factored into our path selection phase, where we prioritize paths not only based on their migration cost but also on their estimated transmission delay. The aim is to ensure that the chosen path aligns with the permissible downtime specified in the SLAs of the involved VMs.

In Algorithm 1, set D is the set of DC with insufficient RE, and set $N-D$ is the set of DCs with sufficient RE that can host VMs. Lines 1–6, the controller identifies the DCs that are experiencing a lack of RE and adds them to the list D. Then, the first phase of MAB starts by going through the DCs in set $(N-D)$ and selecting the one that has the lowest power consumption and BE consumption equal to zero. Based on the (3.3) formulation, when $\Gamma_{s,\theta}$ is equal to zero, the DC s available RE is higher than

Algorithm 1 Modified SW-LCB (MSW-LCB)

Input: $G = (N, L)$, Available RE of DCs, Window size τ, Parameter σ, $\theta = 0$
Output: Regret, Total cost

for all DCs in set N **do**
  Calculate the expected energy consumption of the DC;
  if expectation of the energy consumption of DC $\geq$ available RE of DC **then**
    Create a list D and append the corresponding DC to list D;
  end if
end for
for all DCs s in list D **do**
  while the remaining workload of DC $s \geq$ available RE of DC s **do**
    for each DC j in $(N - D)$ **do**
      Estimate the power consumption of the DC j using (3.7) and (3.8)
      $$\delta_\theta^j = \bar{f}_\theta(j, \tau) - E_\theta(j, \tau);$$
      Calculate BE consumption cost
    end for
    Select the DC m_θ with the lowest power consumption and the BE consumption equal to zero based on (3.3)
    if a destination DC is found **then**
      for each path i between s and m_θ **do**
        Estimate the transmission delays of path i using (3.7) and (3.8)
        $$\Delta_\theta^i = \bar{f}_\theta(i, \tau) - E_\theta(i, \tau);$$
        Calculate the migration cost using the estimated delay;
      end for
      Select the path a_θ with the lowest migration cost which the path estimated delay meets the VM downtime SLA.
      Do the migration based on the selected path a_θ using Algorithm 2;
      Observe the transmission delay and power consumption of DC m_θ;
      Check the remaining workload in DC s;
      $\theta = \theta + 1$;
    else
      Stop the Algorithm
    end if
  end while
end for

Algorithm 2 Optical Grooming

Input: Source DC s, path a_θ, Required resources of the VM (CPU and bandwidth)
Output: Number of used SBVTs and ROADMs

```
1: Do Operation 1 from s using path a_θ;
2: if Operation 1 succeeds then
3:    Update physical resources' capacity;
4: else
5:    Do Operation 2;
6:    if Operation 2 succeeds then
7:       Update physical resources' capacity;
8:    else
9:       Do Operation 3;
10:      if Operation 3 succeeds then
11:         Update physical resources' capacity;
12:      end if
13:   end if
14: end if
```

its power consumption. After selecting the destination DC, the algorithm begins to find a proper path between source and destination. It explores the paths and chooses the one with the lowest transmission delay that meets the VM's SLA. Then, the VM migrates to the selected DC through the selected path using Algorithm 2. Note that the path that is created in the second phase is not necessarily the shortest path. In Algorithm 2, this path is given as an input to the algorithm.

3.2.2 Optical Grooming

An SBVT can logically divide its capacity into several sub-transponders and allocate each one to a separate optical flow. As a result, using SBVTs provides a new way to perform traffic grooming, namely optical grooming [16].

Basically, traffic grooming refers to the process of aggregating many small flows into a larger one. In optical grooming, we can use one SBVT to transfer multiple optical flows instead of using several separate low-capacity transponders. In this way, the overhead power consumption in the optical network and the number of transponders will decrease. Algorithm 2 describes the routing and optical grooming from the s-th DC to the selected arm in the corresponding round. According to [6] and [41], for a new migration request, there are three possible optical grooming operations. Fig. 3.2 illustrates these operations.

Operation 1: Creating a new lightpath by using sub-transponders of the existing SBVTs at the source and destination. In other words, the new lightpath is groomed optically by existing optical flows in the corresponding nodes.
Operation 2: Exploiting optical grooming with an SBVT at the source/destination nodes as well as activating a new SBVT at the destination/source.
Operation 3: Provisioning a completely new lightpath by utilizing new SBVTs at the source and destination nodes.

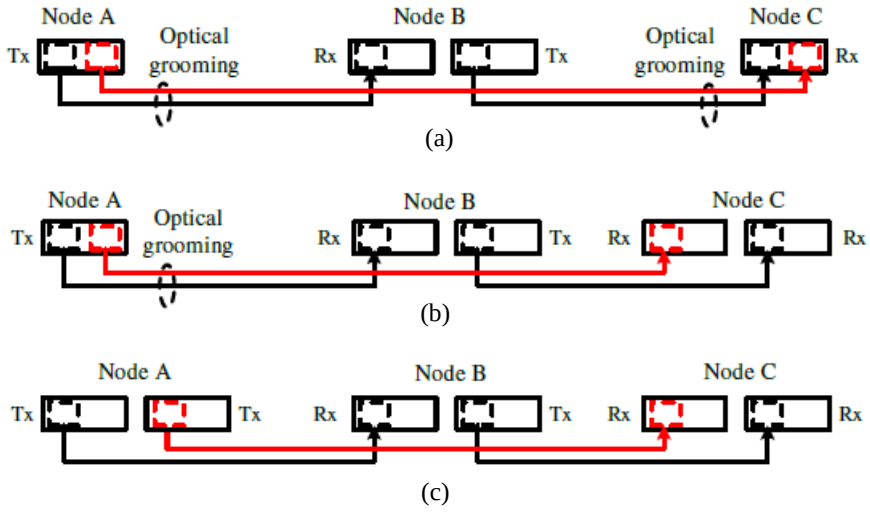

Figure 3.2: Optical grooming operations; (a) Operation 1, (b) Operation 2, (c) Operation 3 [6].

3.2.3 SDN-Enabled VM Migration

In the envisioned architecture for deploying our VM migration algorithm, we leverage the capabilities of Software-Defined Networking (SDN). This approach provides a flexible and programmable network infrastructure, allowing for dynamic management and optimization of resources. SDN offers a flexible and programmable infrastructure where the control plane is decoupled from the data plane. In our context, SDN controllers play a pivotal role in orchestrating the execution of the VM migration algorithm, and the MSW-LCB algorithm is being executed within the SDN controllers.

The SDN controllers, while distributed across the network, operate in a centralized manner. This centralized control ensures coordinated decision-making, preventing scenarios of DCs becoming either overloaded or underloaded. This centralized

approach is crucial for optimizing the VM migration process, as decisions regarding DC selection and path establishment can be made holistically, considering the global state of the network.

Furthermore, the connectivity among SDN controllers enables seamless communication and collaboration. This inter-controller communication is vital for sharing information, load balancing, and maintaining a cohesive network operation. Through this interconnected SDN controller architecture, we aim to enhance the efficiency and effectiveness of the VM migration algorithm by leveraging a unified view of the network's resources and demands. This SDN-based architecture not only aligns with contemporary networking paradigms but also ensures that our VM migration algorithm can adapt dynamically to the changing conditions of the network, thereby optimizing the energy-aware inter-DC VM migration process.

3.2.4 Algorithm Complexity

Since the MSW-LCB algorithm retains only the last τ rounds of the transmission delay, it occupies less space than algorithms that use the entire history. As a result, the space complexity is linear with respect to the τ.

In terms of time and computational complexity, finding the DCs with an insufficient amount of REs has an $O(|N|)$ complexity. The exploration and exploitation part has the complexity of $O(|N|^2+|A_{s,m_\theta}|\cdot|N|)$, where $|A_{s,m_\theta}|$ is the number of paths between source DC s and destination DC m_θ. Finally, the complexity of the optical grooming is $O(K \cdot H)$, where K and H are the total number of sub-transponders in an SBVT and the number of migrations, respectively.

Chapter 4

Performance Evaluation and Analysis

In this chapter, we conduct extensive numerical simulations to evaluate the performance of the MSW-LCB algorithm. First, we outline our settings and parameters, and then we discuss our results. We give an assessment based on real-world datasets that provides important insights into the strengths and advantages of our algorithms in a semi-realistic environment.

4.1 Parameters Settings

We implement our proposed algorithm using the Python programming language on a computer equipped with an i7-4510U processor and 12 GB of RAM. The performance of the proposed algorithm is evaluated on the USNET topology, a real-world network infrastructure representing DCs across the United States. This topology, acquired from [42], is a geographically diverse network of interconnected DCs. Fig. 4.1 showcases the USNET topology, with each node representing a DC strategically located across the country. The numbers assigned to the links between nodes denote the geographical distances between these DCs in kilometers. This topology, carefully integrated into our evaluation framework, mirrors the geographic distribution of DCs in the United States and serves as a robust testbed for our algorithm.

We assume that each DC has 100 servers, denoted by ν, and each has 16 CPU cores, denoted by ω. The idle and peak power consumptions of a server are tuned to 100 and 200 Watts, respectively. Additionally, each SBVT has a capacity of 400 Gbps

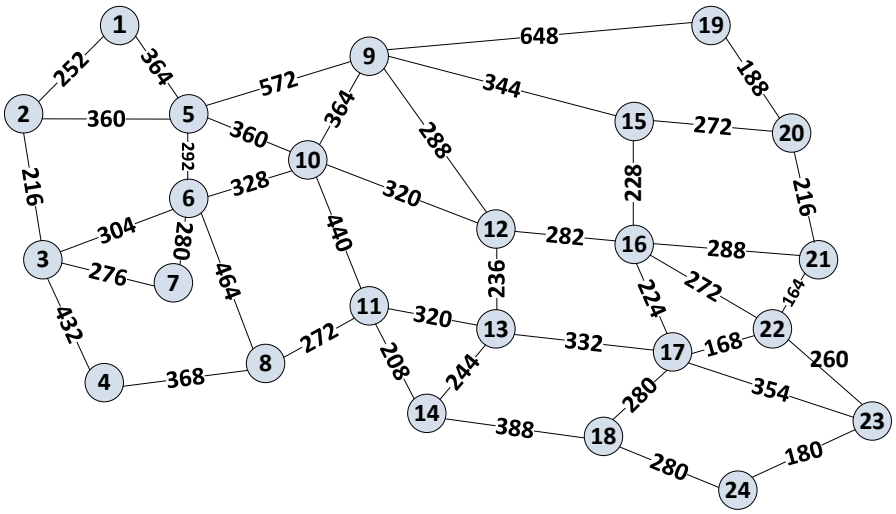

Figure 4.1: USNET topology.

and can be sliced into a maximum of 10 sub-transponders. The parameters σ and η are set to 1 and 1.2, respectively. Basically, the PUE value of 1.2 is chosen to strike a balance between ensuring that the DC receives sufficient power for its IT equipment (a PUE below 1) and minimizing power wastage (a very high PUE). This value optimally aligns with operational efficiency. Moreover, the amount of α for each DC is chosen from a range of [9–15] cents per unit. These values were determined based on a review of existing literature, considering the cost structures and financial models applied in DC operations. This range encompasses various pricing scenarios observed in different regions and states, reflecting the diverse energy market conditions [25]. The values of β and ρ are set to 22 and 0.1 cents per unit, respectively [30], [32]. These values are commonly utilized in research involving DC energy efficiency and operational cost studies [39]. The cost of the ROADM depends on the node degree in the USNET topology. USNET has six nodes with degree 5, five nodes with degree 4, eight nodes with degree 3, and five nodes with degree 2. In addition, for the calculation of c_{WSS}, we considered the ROADM node architecture as shown in (4.1) and Fig. 4.2. This figure provides a visual representation of (4.1) when the value of M is set to 2. The costs of WSS(1∗9), WSS(1∗20), and WSS(9∗9) are 4, 6, and 48, respectively. The cost of the built-in amplifier is set at 1.9 [39]. These values encapsulate the operational expenses incurred per request. They encompass a spectrum of costs, including power

consumption, equipment maintenance, and resource utilization.

$$\begin{aligned} C_{\text{ROADM}} &= \sum_{\theta=1}^{\Theta} 1.3 \cdot M \cdot (c_{\text{Amp}} + c_{\text{WSS}}) \\ &= \sum_{\theta=1}^{\Theta} 1.3 \cdot M \cdot (2 \cdot \text{WSS}(1 * 9) + \text{c}_{\text{Amp}}) \\ &\quad +2 \cdot (\text{WSS}(1 * 20) + \text{WSS}(9 * 9)). \end{aligned} \tag{4.1}$$

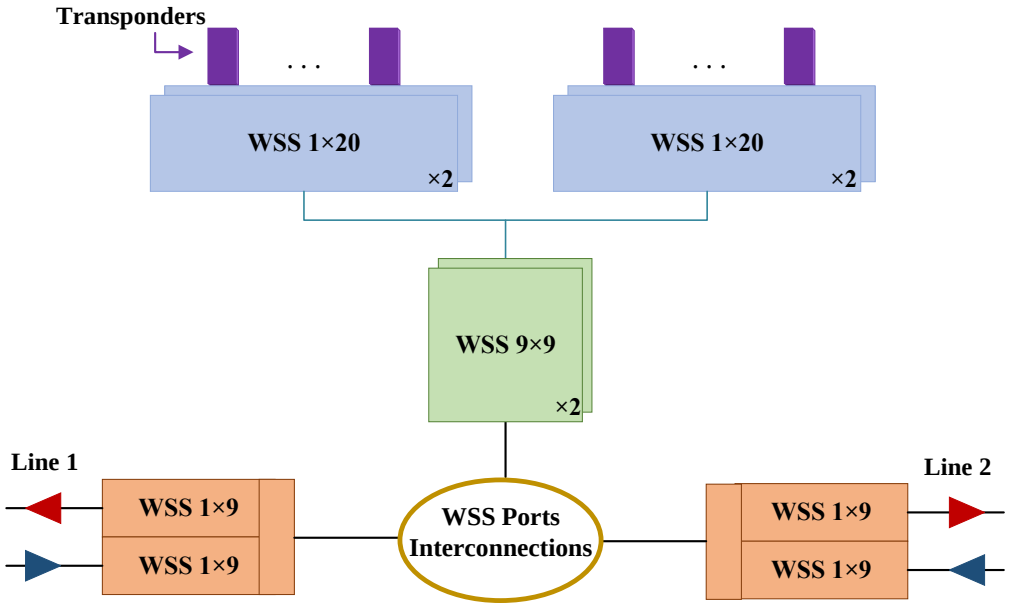

Figure 4.2: ROADM architecture in our setting with $M = 2$.

In (4.1), M is the degree of the ROADM. As mentioned earlier, ROADMs have built-in wavelength-selective switches (WSS) and amplifiers to direct and boost the optical signals. WSS($1 * \text{X}$) means the switch has 1 input and X output ports. A coefficient of 1.3 is applied for the increased cost associated with newer technology in ROADMs capable of supporting EON flexibility. Without this coefficient (i.e., when it equals 1), the formulation represents the cost for a ROADM in a fixed grid WDM node. However, it is estimated that with the introduction of advanced ROADMs, which offer enhanced capabilities and flexibility, the cost is approximately 1.3 times higher.

Regarding the value of R_s, we utilized a comprehensive dataset encompassing in-

formation about the availability of RE sources [43][1] in each DC. This dataset featured diverse data regarding various RE sources, including solar and wind. To establish a coherent representation, we collated data specific to these RE sources for the United States in 2020 and calculated their cumulative availability. Based on our findings, we determined that the lowest RE availability amounted to approximately 11% of a DC's peak power consumption. Therefore, the value of RE for each DC is generated from a range of $[0.11 \cdot P_{\text{peak}} \cdot \eta \cdot \nu, P_{\text{peak}} \cdot \eta \cdot \nu]$. It is noteworthy that this dataset was compiled in 2020, and over the years, this percentage has shown a progressive increase.

In addition, we consider the distribution of CPU core utilization for background workloads within a DC. This utilization is assumed to follow a uniform distribution, falling within the range of 30% to 50% of a server's total CPU capacity, denoted as ω. This range aligns with best practices in DC management, where server utilization between 30% and 50% is considered optimal. Servers operating below 30% utilization are often candidates for consolidation, maximizing resource efficiency. Conversely, maintaining an average utilization of around 50% is deemed ideal for server performance. Therefore, background workloads falling within the 30% to 50% range represent scenarios where servers are functioning optimally and have the capacity to host new VMs. This approach allows us to model realistic DC conditions, ensuring that our simulations closely resemble practical, real-world scenarios.

To simulate and evaluate the network delays and VM characteristics, we meticulously harnessed datasets from reputable sources. Our network delay dataset, sourced from [44][2], provides invaluable insights into the average latency of network links (as indicated in the dataset's "avg-lat-ms" column). Using this dataset not only ensures a more authentic representation of real-world scenarios but also equips our algorithm with the adaptability required to tackle an extensive range of potential delay situations. Additionally, the dataset "WorkloadTrace"[3] contributed significantly to deepening our comprehension of VM characteristics. Specifically, we leveraged the "CPU utilization percentage" column in the dataset to inform our algorithm about VM CPU requirements, enhancing the authenticity of our simulations. These carefully selected datasets, subjected to rigorous modification and synthesis, aid in establishing the accuracy and practical significance of our study.

[1] https://www.kaggle.com/datasets/programmerrdai/renewable-energy
[2] https://www.kaggle.com/datasets/dhruvildave/ookla-internet-speed-dataset
[3] https://www.kaggle.com/datasets/ashikhassan007/workloadtrace

Furthermore, in the realm of cloud computing, major providers often guarantee a remarkable 99.99 percent uptime for VMs. Translated into tangible terms, this is equivalent to a maximum allowable downtime of approximately 52.6 minutes per year or 1 minute per day (60,000 milliseconds). In our performance analysis, we can confidently assert that our algorithm stands resilient in preserving these stringent SLAs. Through meticulous evaluation against real-world datasets and input, our algorithm consistently ensures that VMs undergoing migration remain well within the bounds of their prescribed SLAs. This reliability underscores the effectiveness of our approach in meeting the high standards set by prominent cloud service providers.

We compared our algorithm with four approaches as follows:

- *SW-LCB*: This algorithm is proposed by us, and is discussed in [17]. In SW-LCB, each arm is defined as a destination DC. In each round, an arm is selected, and the VM will be migrated through the shortest path to the chosen arm.

- *ϵ-Greedy*: It is an MAB-based policy that in each round θ, an arm is uniformly chosen with probability $\epsilon = \frac{1}{\theta}$ [32].

- *KUBE*: It stands for knapsack–based upper confidence bound exploration and exploitation [45]. In KUBE, for each arm j, we estimate the delay by

$$\bar{d}_\theta(s,j) - \sqrt{2 \cdot \log \theta / \sum_{k=1}^{\theta} \mathbb{1}_{\{D_k=j\}}},$$

  where $\bar{d}_\theta(s,j)$ is calculated by

$$\bar{d}_\theta(s,j) = \sum_{k=1}^{\theta} d_k^{s,j} \mathbb{1}_{\{D_k=j\}} / \sum_{k=1}^{\theta} \mathbb{1}_{\{D_k=j\}}.$$

- *Anycast-JRE*: It does not include any online decision-making algorithms. This algorithm specifies DCs with insufficient RE first. Then it compares the K-shortest paths from the source DC to the destination DC with sufficient RE, and finally, it selects the path with the highest weight for migration [30].

4.2 Results and Discussions

We thoroughly evaluate and compare the results of MSW-LCB with the other benchmark algorithms, shedding light on various aspects and providing detailed analysis. All simulations are repeated 100 times, and the averaged results are shown in this section.

Fig. 4.3 depicts the average BE consumption cost of DCs before and after migration using the MSW-LCB algorithm. As it is shown, the gap between the two graphs narrows as the number of requests in each DC grows, and the MSW-LCB algorithm demonstrates a significant drop in BE usage in DCs due to the MAB-based migration. As illustrated, for 400 requests, we obtained a cost reduction of around 69%. However, for 680 requests, this improvement drops to 41%, as the capacity of DCs is limited.

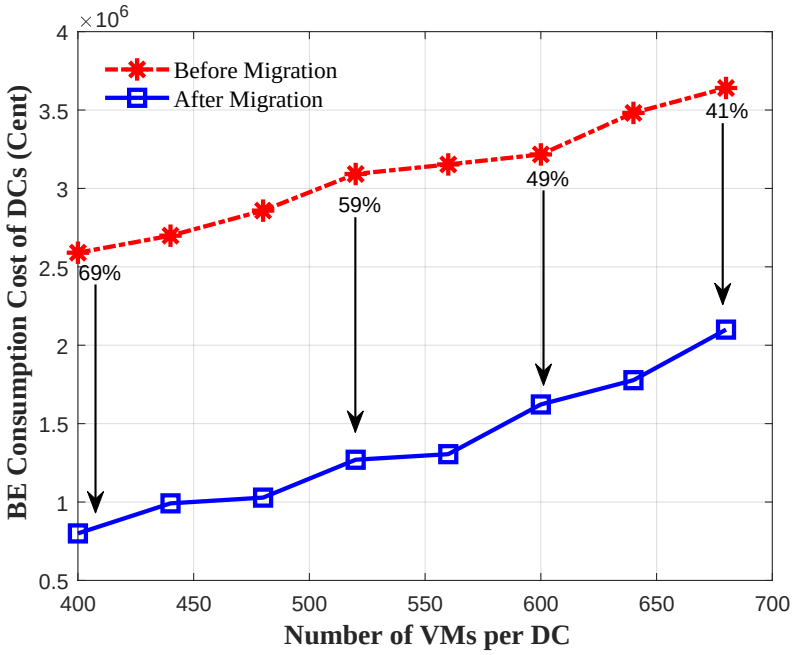

Figure 4.3: Comparison of DCs' BE consumption costs in the USNET before and after migration using Algorithm 1.

Fig. 4.4 compares the total cost of MSW-LCB with other benchmark methods. According to the graph, when the number of requests grows, a significant divergence appears, showing that MSW-LCB outperforms SW-LCB and other algorithms. In particular, MSW-LCB obtains a substantial reduction in total cost of 15% on average when the number of requests reaches 680 compared to SW-LCB. Moreover, the total

cost in a high number of requests has declined typically by 23%, 34%, and 39% when compared to KUBE, ϵ-Greedy, and Anycast-JRE, respectively, which demonstrates the effectiveness of the MSW-LCB algorithm.

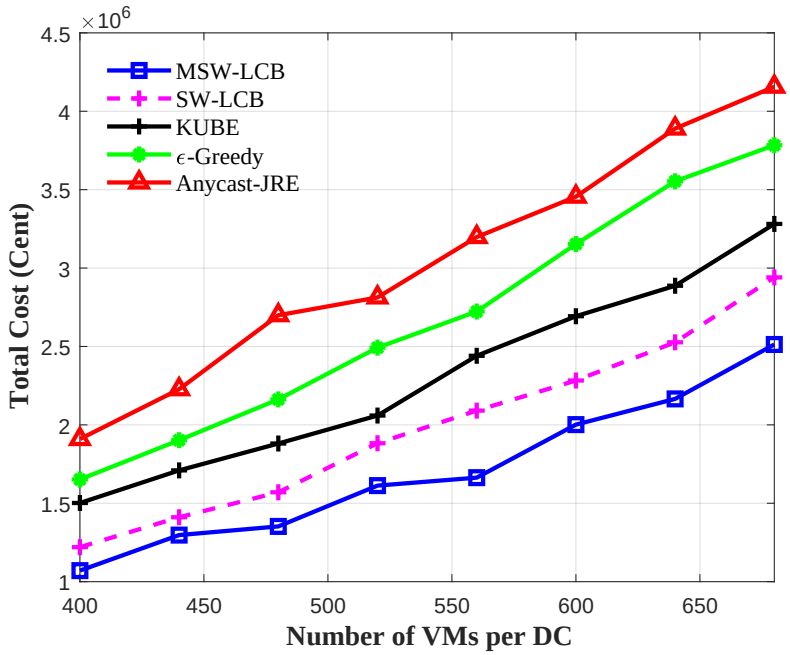

Figure 4.4: Total cost comparison of MSW-LCB and other algorithms in USNET.

The fundamental reason contributing to the lower total cost in contrast to the SW-LCB is the lower migration cost, which is the outcome of the significant modifications and improvements in path selection in the MSW-LCB. To assess the impact of these improvements, we conducted a comparison of migration costs. As illustrated in Fig. 4.5, on average, the migration cost using MSW-LCB exhibits a significant decrease of approximately 15% compared to the SW-LCB. Additionally, the MSW-LCB algorithm demonstrates its effectiveness as the number of requests in the DCs increases. By using a sliding window approach and focusing on recent network data, the delay estimation becomes more accurate, leading to less migration downtime and cost. This approach provides on average around 26% improvement over KUBE and over 37% decrease in ϵ-Greedy for the large number of requests.

This notable reduction in migration cost highlights the efficiency and effectiveness of the MSW-LCB algorithm in finding the best path with minimum delays for VM migration. This efficiency is reflected in the cost of the optical network as well. Fig. 4.6 depicts the cost comparison of optical network equipment. Optical grooming is

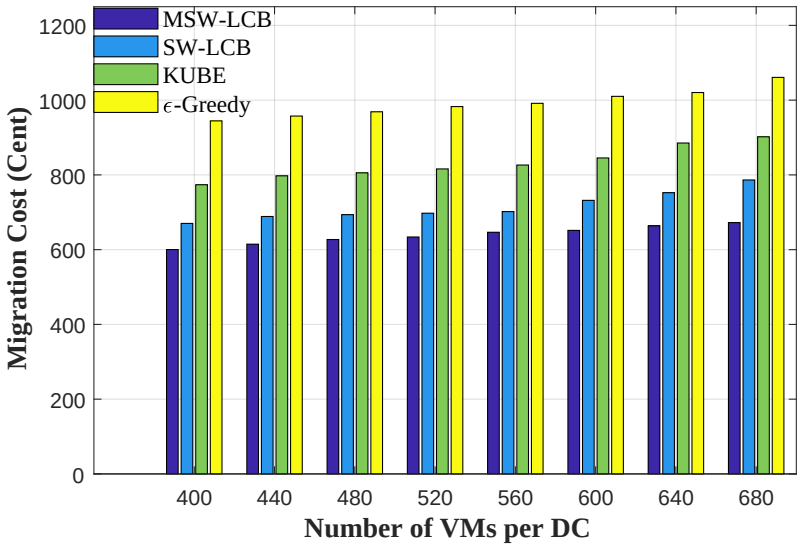

Figure 4.5: Comparison of migration cost in MSW-LCB and other algorithms in USNET.

the primary method that aids in lowering the network devices' cost. As shown in Fig. 4.6, optical grooming reduces device expenses by 12% for both MSW-LCB and SW-LCB. In addition, the network cost for MSW-LCB is approximately 10% lower than that achieved by ϵ-Greedy.

Fig. 4.7 shows the regret achieved by MSW-LCB. To demonstrate this figure, we selected a certain DC, set the number of requests to 500, and then plotted the regret curves. As shown in Fig. 4.7, the MSW-LCB's regret converges faster than the other algorithms. It reduces the regret by 15%, 27%, and 36% compared to SW-LCB, KUBE, and ϵ-Greedy. In general, MSW-LCB exhibits faster convergence in identifying the best arm, enabling quicker decision-making and improved performance.

In determining the number of migrations in our study, we acknowledge the dynamic nature of DCs, where the availability of RE and the ongoing background workload play pivotal roles. The interplay of these factors results in a scenario where the number of migrations varies for each DC. The availability of RE influences the energy-aware migration decision, as our algorithm aims to migrate VMs from DCs with limited RE to those with sufficient RE. Simultaneously, the background workload, reflected in the utilization of CPU cores, provides insight into the operational state of a DC. Fig. 4.7, showcases the trade-off between migration decisions and the associated regret. This figure not only visualizes the regret but also presents

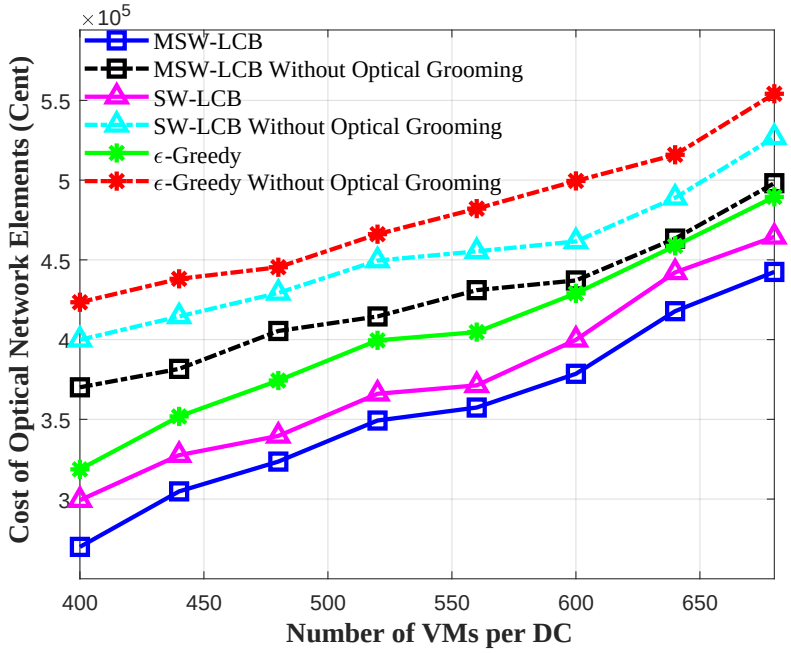

Figure 4.6: Cost comparison of optical network elements of MSW-LCB and other algorithms in USNET.

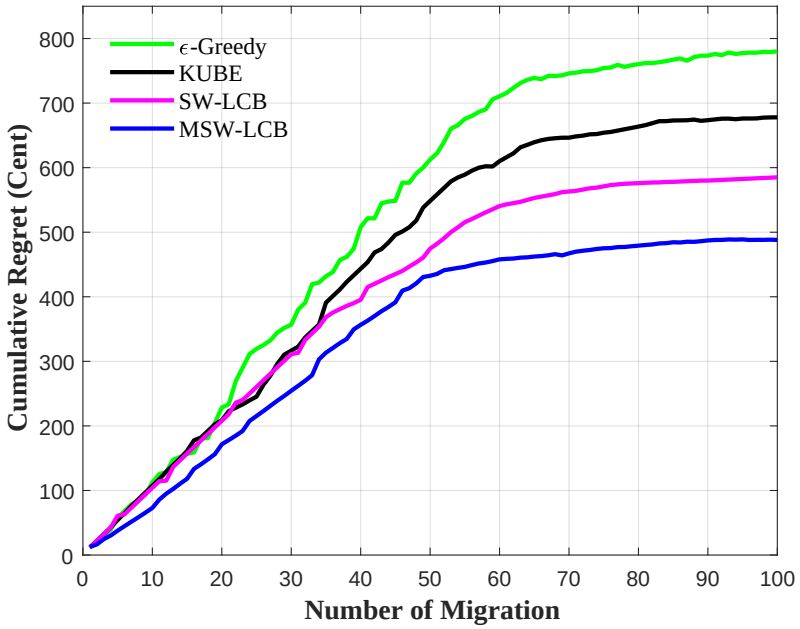

Figure 4.7: Comparison of regret for MSW-LCB and other algorithms in USNET.

the average number of migrations in DCs. It emphasizes the adaptability of our approach, highlighting how the algorithm dynamically responds to each DC's unique

conditions, optimizing migration decisions based on the availability of RE and the prevailing workload

Additionally, our investigation of the influence of window size within the MSW-LCB algorithm revealed noteworthy insights into its performance dynamics. As showcased in Fig. 4.8, we conducted a thorough analysis to evaluate how different window sizes affect the regret. It is evident from the figure that the choice of window size plays a pivotal role in shaping the algorithm's performance. Specifically, we observed that as the window size decreases, the regret tends to increase. This behaviour aligns with our expectations, as a smaller window limits the historical information considered during the decision-making process. Remarkably, our experiments indicate that for our specific setting, a window size of 50 strikes an optimal balance. This observation underscores the importance of tailoring algorithm parameters to the characteristics of the problem domain, reaffirming the adaptability and effectiveness of MSW-LCB in dynamic inter-DC VM migration scenarios.

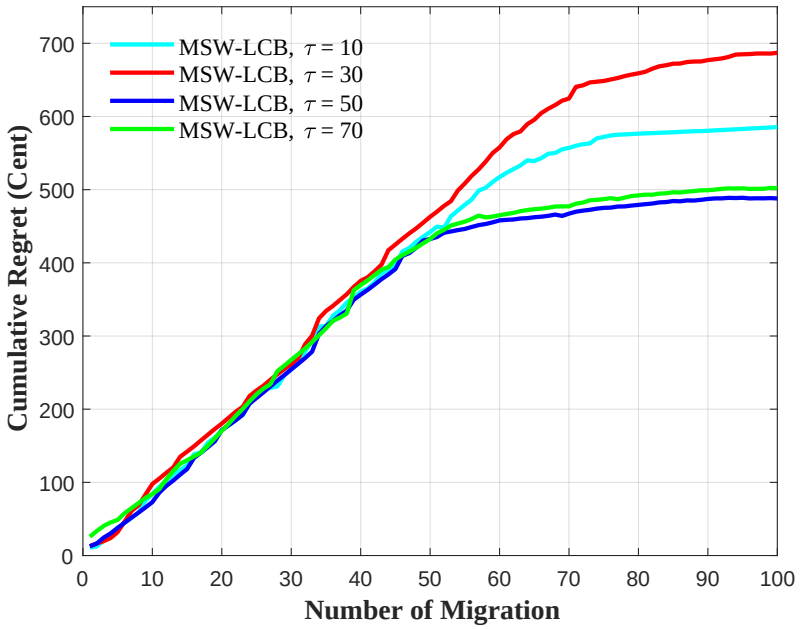

Figure 4.8: Comparison of regret for different values of τ.

The observed increase in regret as the window size expands to 70 within our MSW-LCB algorithm can be attributed to several key factors. First, a larger window size may lead to overfitting as the algorithm becomes heavily influenced by old historical data. Second, the delayed learning effect becomes more pronounced with larger windows, making it slower to adapt to new changes in the system. In scenarios where the

system dynamics are evolving rapidly, this can result in suboptimal arm selections and, consequently, increased regret. Lastly, as there is significant variability in delay and workload demands, a longer window size may not respond effectively to these fluctuations.

Table 4.1: Runtime comparison

Number of Requests	400	440	480	520	560	600	640	680
MSW-LCB (ms)	240	251	255	260	285	290	300	312
SW-LCB (ms)	124	131	144	152	166	173	179	186
KUBE (ms)	126	135	147	153	169	177	181	189
ϵ-Greedy (ms)	81	83	88	95	102	106	109	113

Table 4.1 provides a comprehensive comparison of the runtimes for different algorithms under varying numbers of requests. As observed, our MSW-LCB algorithm exhibits slightly longer runtimes compared to the SW-LCB and other benchmark algorithms. The higher runtime in MSW-LCB is primarily due to the sequential MAB framework employed in our approach, which involves making decisions in two stages: DC selection and path selection. To manage the computational and time complexities of path selection, we optimized the process by considering only the top 5 paths with the lowest delay from the previous round. This strategy allows us to strike a balance between path diversity and computational efficiency, resulting in an impressive runtime of below one second. Despite the slightly longer runtimes, it is important to emphasize that MSW-LCB delivers substantial advantages in terms of cost reduction and regret minimization, making it a compelling choice for scenarios where these objectives outweigh the modest increase in computation time. Furthermore, it is worth noting that the runtimes remain well within reasonable limits for practical applications, even with a high number of requests. Therefore, while MSW-LCB may involve a marginal increase in computational overhead, its superior performance in cost optimization positions it as a strong candidate for inter-DC VM migration in large-scale DC networks.

Chapter 5

Conclusions

This thesis explored the topic of inter-DC VM migration over EONs and proposed an innovative approach using the MAB formulation. In Chapter 3, the MSW-LCB algorithm was introduced to address the challenge of reducing DCs' BE consumption and migration costs using MAB formulation in two stages. In the first stage, the algorithm intelligently selects the destination DC for migration, taking into account the available RE and power consumption of the DCs. In the second stage, MSW-LCB identifies the most suitable path for migration with the lowest delay. Additionally, the optical grooming technique was implemented to optimize the migration process by aggregating optical flows and minimizing the required resources and optical network costs.

In order to evaluate the effectiveness of the MSW-LCB, we conducted extensive simulations on the USNET topology using real-world datasets. The results demonstrated significant improvements achieved through the MSW-LCB-based migration. Specifically, the BE consumption cost was substantially reduced by 41% in the high number of requests. Additionally, the total cost dropped by 15%, 23%, and 34% compared to the SW-LCB, KUBE, and ϵ-Greedy algorithms, respectively. As a result of optical grooming, the network cost has decreased significantly in comparison to the SW-LCB and ϵ-Greedy. In addition, implementing the MSW-LCB leads to 15% improvement in migration costs with respect to the SW-LCB, as well as 26% and 37% drop in comparison with KUBE and ϵ-Greedy, respectively. Finally, we presented the regret results and demonstrated that MSW-LCB converged faster and imposed 15%, 27% and 36% less regret compared to the SW-LCB, KUBE, and ϵ-Greedy, respectively. In general, our analysis has shown that MSW-LCB offers significant advantages in terms of cost reduction and faster convergence to optimal arm.

5.1 Future Works

There are several promising avenues for future research in the field of inter-DC VM migration. This section presents a set of valuable suggestions for future work that can further enhance the understanding and application of inter-DC VM migration.

One area of potential improvement lies in enhancing the decision-making process within DCs regarding which VMs should be migrated and which should remain in the source DC. In our research, we have assumed that all VMs possess equal priority and have not taken into account any specific SLAs related to compliance requirements for individual VMs. For instance, certain VMs may have restrictions on being migrated to different regions or states due to cost, privacy, or security constraints. Therefore, it is crucial to make informed decisions by considering the unique requirements of each VM in the context of inter-DC migration. By intelligently incorporating these considerations into the decision-making process, the efficiency and effectiveness of VM migration can be significantly enhanced.

Another possible improvement is combining electrical and optical grooming algorithms can offer the potential to minimize network costs by leveraging the advantages of each technique. It enables effective utilization of network resources, reduces network element power consumption, and provides more cost-efficient grooming options. Further research and development in this area can contribute to enhancing the overall performance and cost-effectiveness of inter-DC VM migration.

Bibliography

[1] S. Albahli, M. Shiraz, and N. Ayub, "Electricity price forecasting for cloud computing using an enhanced machine learning model," *IEEE Access*, vol. 8, 11 2020.

[2] T. He and R. Buyya, "A taxonomy of live migration management in cloud computing," *arXiv preprint arXiv:2112.02593*, 2021.

[3] John, "OADM vs. ROADM: What's the difference?" 16, July 2020, published in FS Community: https://community.fs.com/blog/oadm-vs-roadm.html.

[4] B. Mukherjee, V. López, and L. Velasco, *Elastic Optical Networks; Architectures, Technologies, and Control*. Springer, 2016.

[5] J. M. Simmons, *Optical Network Design and Planning*, 1st ed. Springer Publishing Company, Incorporated, 2008.

[6] J. Zhang, Y. Zhao, X. Yu, J. Zhang, M. Song, Y. Ji, and B. Mukherjee, "Energy-efficient traffic grooming in sliceable-transponder-equipped ip-over-elastic optical networks," *Journal of Optical Communications and Networking*, vol. 7, no. 1, pp. 142–152, 2015.

[7] B. Wang, Z. Qi, R. Ma, H. Guan, and A. V. Vasilakos, "A survey on data center networking for cloud computing," *Computer Networks*, vol. 91, pp. 528–547, 2015.

[8] M. Pickavet, W. Vereecken, S. Demeyer, P. Audenaert, B. Vermeulen, C. Develder, D. Colle, B. Dhoedt, and P. Demeester, "Worldwide energy needs for ICT: The rise of power-aware networking," in *2nd international symposium on advanced networks and telecommunication systems*. IEEE, 2008, pp. 1–3.

[9] M. Xu and R. Buyya, "Managing renewable energy and carbon footprint in multi-cloud computing environments," *Journal of Parallel and Distributed Computing*, vol. 135, pp. 191–202, 2020.

[10] U. Mandal, M. F. Habib, S. Zhang, B. Mukherjee, and M. Tornatore, "Greening the cloud using renewable-energy-aware service migration," *IEEE Network*, vol. 27, no. 6, pp. 36–43, 2013.

[11] A. Gupta, U. Mandal, P. Chowdhury, M. Tornatore, and B. Mukherjee, "Cost-efficient live VM migration based on varying electricity cost in optical cloud networks," *Photonic Network Communications*, vol. 30, no. 3, pp. 376–386, 2015.

[12] T. Wood, K. K. Ramakrishnan, P. Shenoy, J. Van der Merwe, J. Hwang, G. Liu, and L. Chaufournier, "Cloudnet: Dynamic pooling of cloud resources by live WAN migration of virtual machines," *IEEE/ACM Transactions on Networking*, vol. 23, no. 5, pp. 1568–1583, 2015.

[13] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire, "The nonstochastic multiarmed bandit problem," *SIAM Journal on Computing*, vol. 32, no. 1, pp. 48–77, 2002.

[14] J. Zhao and S. Subramaniam, "Virtual network mapping in elastic optical networks with sliceable transponders," *Photonic Network Communications*, vol. 40, no. 3, pp. 281–292, 2020.

[15] M. Jinno, H. Takara, Y. Sone, K. Yonenaga, and A. Hirano, "Multiflow optical transponder for efficient multilayer optical networking," *IEEE Communications Magazine*, vol. 50, no. 5, pp. 56–65, 2012.

[16] V. López, L. Velasco *et al.*, *Elastic optical networks*. Springer, 2016.

[17] F. Salehnejad-Amri, Z. Huang, K. Liu, and J. Pan, "Energy-aware inter-data center vm migration over elastic optical networks," *IEEE Global Communications Conference on Green Communications Systems and Networks*, 2023.

[18] F. Zhang, G. Liu, X. Fu, and R. Yahyapour, "A survey on virtual machine migration: Challenges, techniques, and open issues," *IEEE Communications Surveys and Tutorials*, vol. 20, no. 2, pp. 1206–1243, 2018.

[19] M. Imran, M. Ibrahim, M. S. U. Din, M. A. U. Rehman, and B. S. Kim, "Live virtual machine migration: A survey, research challenges, and future directions," *Computers and Electrical Engineering*, vol. 103, p. 108297, 2022.

[20] R. Dutta, A. E. Kamal, and G. N. Rouskas, *Traffic grooming for optical networks: foundations, techniques and frontiers*. Springer Science & Business Media, 2008.

[21] O. Tamm, C. Hermsmeyer, and A. M. Rush, "Eco-sustainable system and network architectures for future transport networks," *Bell Labs Technical Journal*, vol. 14, no. 4, pp. 311–327, 2010.

[22] R. S. Tucker, "Green optical communications—part ii: Energy limitations in networks," *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 17, no. 2, pp. 261–274, 2010.

[23] S. C. Hoi, D. Sahoo, J. Lu, and P. Zhao, "Online learning: A comprehensive survey," *Neurocomputing*, vol. 459, pp. 249–289, 2021.

[24] B. Mahesh, "Machine learning algorithms-a review," *International Journal of Science and Research (IJSR).[Internet]*, vol. 9, pp. 381–386, 2020.

[25] L. Wang, "Bandit-based delay-aware service function chain orchestration at the edge," Master's thesis, University of Calgary, 2021.

[26] T. He, A. N. Toosi, and R. Buyya, "Performance evaluation of live virtual machine migration in SDN-enabled cloud data centers," *Journal of Parallel and Distributed Computing*, vol. 131, pp. 55–68, 2019.

[27] O. Ayoub, A. De Sousa, S. Mendieta, F. Musumeci, and M. Tornatore, "Online virtual machine evacuation for disaster resilience in inter-data center networks," *IEEE Transactions on Network and Service Management*, vol. 18, no. 2, pp. 1990–2001, 2021.

[28] H. Teyeb, A. Balma, N. B. Hadj-Alouane, S. Tata, and A. B. Hadj-Alouane, "Traffic-aware virtual machine placement in geographically distributed clouds," *2014 International Conference on Control, Decision and Information Technologies (CoDIT)*, pp. 024–029, 2014.

[29] M. Ghamkhari and H. Mohsenian-Rad, "Energy and performance management of green data centers: A profit maximization approach," *IEEE transactions on Smart Grid*, vol. 4, no. 2, pp. 1017–1025, 2013.

[30] L. Zhang, T. Han, and N. Ansari, "Energy-aware virtual machine management in inter-datacenter networks over elastic optical infrastructure," *IEEE Transactions on Green Communications and Networking*, vol. 2, no. 1, pp. 305–315, 2017.

[31] B. Kantarci, L. Foschini, A. Corradi, and H. T. Mouftah, "Inter-and-intra data center VM-placement for energy-efficient large-scale cloud systems," *2012 Globecom Workshops*, pp. 708–713, 2012.

[32] S. Ghoorchian and S. Maghsudi, "Multi-armed bandit for energy-efficient and delay-sensitive edge computing in dynamic networks with uncertainty," *IEEE Transactions on Cognitive Communications and Networking*, vol. 7, no. 1, pp. 279–293, 2020.

[33] O. Gerstel, M. Jinno, A. Lord, and S. B. Yoo, "Elastic optical networking: A new dawn for the optical layer?" *IEEE communications Magazine*, vol. 50, no. 2, pp. s12–s20, 2012.

[34] M. Klinkowski and K. Walkowiak, "On the advantages of elastic optical networks for provisioning of cloud computing traffic," *IEEE Network*, vol. 27, no. 6, pp. 44–51, 2013.

[35] M. N. Dharmaweera, J. Zhao, L. Yan, M. Karlsson, and E. Agrell, "Traffic-grooming-and multipath-routing-enabled impairment-aware elastic optical networks," *Journal of Optical Communications and Networking*, vol. 8, no. 2, pp. 58–70, 2016.

[36] M. Zhu, P. Gao, J. Zhang, X. Zeng, and S. Zhang, "Energy efficient dynamic virtual optical network embedding in sliceable-transponder-equipped eons," *IEEE Global Communications Conference*, pp. 1–6, 2017.

[37] A. Attarpour, M. Ibrahimi, F. Musumeci, A. Castoldi, M. Ragni, and M. Tornatore, "Minimizing cost of hierarchical otn traffic grooming boards in mesh networks," pp. 3700–3705, 2022.

[38] M. Ibrahimi, O. Ayoub, A. Attarpour, F. Musumeci, A. Castoldi, M. Ragni, and M. Tornatore, "Minimizing equipment and energy cost in mixed 10g and 100g/200g filterless horseshoe networks with hierarchical otn boards," *Annals of Telecommunications*, vol. 78, no. 5-6, pp. 297–311, 2023.

[39] F. Rambach, B. Konrad, L. Dembeck, U. Gebhard, M. Gunkel, M. Quagliotti, L. Serra, and V. López, "A multilayer cost model for metro/core networks," *Journal of Optical Communications and Networking*, vol. 5, no. 3, pp. 210–225, 2013.

[40] A. Garivier and E. Moulines, "On upper-confidence bound policies for non-stationary bandit problems," *arXiv preprint arXiv:0805.3415*, 2008.

[41] M. Zhu, Q. Sun, S. Zhang, P. Gao, B. Chen, and J. Gu, "Energy-aware virtual optical network embedding in sliceable-transponder-enabled elastic optical networks," *IEEE Access*, vol. 7, pp. 41 897–41 912, 2019.

[42] Y. Zong, Y. Ou, A. Hammad, K. Kondepu, R. Nejabati, D. Simeonidou, Y. Liu, and L. Guo, "Location-aware energy efficient virtual network embedding in software-defined optical data center networks," *Journal of Optical Communications and Networking*, vol. 10, no. 7, pp. B58–B70, 2018.

[43] M. R. Hannah Ritchie and P. Rosado, "Energy," *Our World in Data*, 2020, https://ourworldindata.org/energy.

[44] D. Dave, "Speedtest by ookla global fixed and mobile network performance maps." 2020, https://www.kaggle.com/datasets/dhruvildave/ookla-internet-speed-dataset.

[45] L. Tran-Thanh, A. Chapman, A. Rogers, and N. Jennings, "Knapsack based optimal policies for budget–limited multi–armed bandits," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 26, no. 1, 2012, pp. 1134–1140.