



Faculty of Computer Science and Information Technology

**Energy Efficient High-Performance Computing Aware Proactive
Dynamic Virtual Machine Consolidation Technique in Cloud Computing**

Rukshanda Kamran Butt

**Doctor of Philosophy
2022**

Energy Efficient High-Performance Aware Proactive Dynamic Virtual
Machine Consolidation Technique in Cloud Computing

Rukshanda Kamran Butt

A thesis submitted

In fulfilment of the requirements for the degree of Doctor of Philosophy

(Computer Science)

Faculty of Computer Science and Information Technology

UNIVERSITI MALAYSIA SARAWAK

2022

DECLARATION

I declare that the work in this thesis was carried out in accordance with the regulations of Universiti Malaysia Sarawak. Except where due acknowledgements have been made, the work is that of the author alone. The thesis has not been accepted for any degree and is not concurrently submitted in the candidature of any other degree.

Rukshanda

.....

Signature

Name: Rukshanda Kamran Butt

Matric No.: 14010030

Faculty of Computer Science and Information Technology

Universiti Malaysia Sarawak

Date : 17 August 2022

ACKNOWLEDGEMENT

First of All, I thank ALLAH (SWT), the God Almighty, for granting me the health, knowledge, strength, ability, and patience to accomplish this research and blessing me with sympathetic and supportive supervisors and family members.

I am glad to express tremendous gratitude to my supervisor Dr Noor Alamshah B. Bolhassan, for his sympathetic character, knowledge sharing, ideas and continuous support from the first until the last day of this study. His sincere behaviour and constructive feedback enabled me to achieve significant research milestones within the required time frame.

I would also like to thank my Co-supervisor, Dr Nadianatra Binti Musa, for enriching me with innovative ideas and skills by sharing her expertise and knowledge. Due to her unlimited support for reviewing, improving and evaluating my research, I completed my thesis sustaining the high standard of practice in UNIMAS.

At various stages during this study, I faced several undesirable challenges which overburdened me with mental and physical stress. However, this never stopped me from progressing further due to encouraging, moral, and financial support from my husband Kamran butt and my son Muhammad Shayan. I am incredibly thankful to them for their understanding, kindness, belief, and trust in me.

I also express my profound appreciation to my father, mother, brothers and sisters for their unlimited prayers regarding my success during this study. This research would have never been possible without their wishes for my success.

Finally, I would like to thank the management of the Universiti Malaysia Sarawak for making it possible for me to complete my study here in Sarawak. Thank you all.

ABSTRACT

High-Performance Computing (HPC) applications have gained extensive interest in cloud computing. Cloud computing provides HPC applications with enormous on-demand resources; therefore, many studies have tried to investigate the currently available cloud service architectures for running HPC applications in the most effective approach. There is yet a promising research direction to explore techniques to efficiently schedule a mix of HPC and non-HPC applications to increase resource utilization in the cloud data centre. However, there are notable challenges that involve cross-application interference, energy issues, and maintaining service-level agreements (SLAs) in multiple metrics, e.g. response time (web application) vs execution time (HPC application). In High-performance Computing (HPC) applications, the resource usage pattern varies over time, making it challenging to track resource utilization. Better workload predictions will lower energy costs by accurately forecasting future workload demands. For this purpose, proactive dynamic Virtual Machine (VM) consolidation is among the most effective approaches to minimize energy consumption and enhance resource usage while balancing the efficiency of HPC and non-HPC applications in a cloud data centre. Since the VM consolidation problem is strictly NP-hard, many heuristic algorithms have been proposed to tackle the problem. However, these algorithms do not consider the workloads' different nature, incredibly individual classes of HPC workload in the cloud data centre. Thus, this thesis explores the opportunities and challenges of proactive, dynamic VM consolidation for HPC and non-HPC applications on the cloud datacenter. Additionally, this thesis proposes and implements an HPC-aware and energy-efficient scheduling algorithm for proactive, dynamic VM consolidation, which achieves better resource utilization and limits cross-application interference through careful co-location of HPC and Non-HPC applications on

the same pool of resources in a cloud datacenter. Experiments were conducted using CloudSim and real HPC workload traces of Metacentrum HPC Workload. The proposed approach, Energy-Aware Multi-Dimensional Online Bin Packing (EAMDOBP), was tested against Power-aware best fit decreasing algorithm (PABFD), Modified Worst Fit decreasing algorithm (MWFD) and Hybrid Local Regression Host Overload Detection (HLRHOD). Results indicate a relative improvement of 60% in Simulation time and a 73% increase in throughput. Additionally, the utilization of the CPU, RAM, and bandwidth increased, respectively, by 77%, 84%, and 70%. The results reveal that the proposed approach significantly improves all the critical metrics used over other heuristic algorithms.

Keywords: HPC (High-Performance Computing); power management; proactive virtual machine consolidation; host overload detection; multiple regressions.

***Teknik Penggabungan Mesin Maya Dinamik Proaktif yang Cepak Tenaga dan Sedar
Pengkomputeran Berprestasi Tinggi dalam Pengkomputeran Awan***

ABSTRAK

Aplikasi Pengkomputeran Berprestasi Tinggi (HPC) telah mendapat minat yang meluas dalam pengkomputeran awan. Pengkomputeran awan menyediakan aplikasi HPC dengan sumber atas permintaan yang besar; oleh itu, banyak kajian telah cuba menyiasat seni bina perkhidmatan awan yang tersedia pada masa ini untuk menjalankan aplikasi HPC dalam pendekatan yang paling berkesan. Masih terdapat hala tuju penyelidikan yang menjanjikan untuk meneroka teknik untuk menjadualkan gabungan aplikasi HPC dan bukan HPC dengan cekap untuk meningkatkan penggunaan sumber dalam pusat data awan. Walau bagaimanapun, terdapat cabaran ketara yang melibatkan gangguan merentas aplikasi, isu tenaga dan mengekalkan perjanjian peringkat perkhidmatan (SLA) dalam berbilang metrik, mis. masa tindak balas (aplikasi web) vs masa pelaksanaan (aplikasi HPC). Dalam aplikasi Pengkomputeran Berprestasi Tinggi (HPC), corak penggunaan sumber berubah mengikut masa, menjadikannya mencabar untuk menjejak penggunaan sumber. Ramalan beban kerja yang lebih baik akan mengurangkan kos tenaga dengan meramalkan permintaan beban kerja masa hadapan dengan tepat. Untuk tujuan ini, penyatuan Mesin Maya (VM) dinamik proaktif adalah antara pendekatan paling berkesan untuk meminimumkan penggunaan tenaga dan meningkatkan penggunaan sumber sambil mengimbangi kecekapan aplikasi HPC dan bukan HPC dalam pusat data awan. Memandangkan masalah penyatuan VM adalah NP-hard, banyak algoritma heuristik telah dicadangkan untuk menangani masalah tersebut. Walau bagaimanapun, algoritma ini tidak mengambil kira sifat beban kerja yang berbeza, iaitu kelas individu beban kerja HPC yang luar biasa dalam pusat data awan. Oleh itu, tesis ini meneroka

peluang dan cabaran penyatuan VM yang proaktif dan dinamik untuk aplikasi HPC dan bukan HPC pada pusat data awan. Selain itu, tesis ini mencadangkan dan melaksanakan algoritma penjadualan yang sedar HPC dan cekap tenaga untuk penyatuan VM dinamik yang proaktif, yang mencapai penggunaan sumber yang lebih baik dan mengehadkan gangguan merentas aplikasi melalui lokasi bersama yang teliti bagi aplikasi HPC dan Bukan HPC pada yang sama. kumpulan sumber dalam pusat data awan. Eksperimen telah dijalankan menggunakan CloudSim dan surih beban kerja HPC sebenar bagi Metacentrum HPC Workload. Pendekatan yang dicadangkan, Pembungkusan Tong Sampah Dalam Talian Berbilang Dimensi Sedar Tenaga (EAMDOBP), telah diuji terhadap Algoritma Penurunan Muatan Terbaik Sedar Kuasa (PABFD), Algoritma Penurunan Sesuai Tersuai (MWFD) dan Pengesanan Lebihan Hos Regresi Tempatan Hibrid (HLRHOD). Keputusan menunjukkan peningkatan relatif sebanyak 60% dalam masa Simulasi dan peningkatan 73% dalam pemprosesan. Selain itu, penggunaan CPU, RAM, dan lebar jalur meningkat, masing-masing, sebanyak 77%, 84% dan 70%. Hasilnya mendedahkan bahawa pendekatan yang dicadangkan meningkatkan dengan ketara semua metrik kritikal yang digunakan berbanding algoritma heuristik yang lain.

Kata kunci: HPC (Pengkomputeran Berprestasi Tinggi); pengurusan tenaga; penyatuan mesin maya proaktif; pengesanan lebihan hos; regresi berganda.

TABLE OF CONTENTS

	Page
DECLARATION	i
ACKNOWLEDGEMENT	ii
ABSTRACT	iv
<i>ABSTRAK</i>	vi
TABLE OF CONTENTS	viii
LIST OF TABLES	xii
LIST OF FIGURES	xiii
LIST OF ABBREVIATIONS	xiv
CHAPTER 1 INTRODUCTION	1
1.1 Study Background	1
1.2 Problem Background	4
1.3 Problem Statement	7
1.4 Objectives	9
1.5 Scope	10
1.6 Significance of the Study	11
1.7 Organization of Thesis	12
CHAPTER 2 LITERATURE REVIEW	13
2.1 Overview	13

2.2	Definitions of HPC and Cloud	13
2.2.1	High-Performance Computing (HPC)	13
2.2.2	Cloud	14
2.3	Cloud Workload	16
2.4	Types of Cloud Services	17
2.5	Virtualization	23
2.6	Virtual Machine Allocations	24
2.6.1	Static VM Provisioning	25
2.6.2	Dynamic VM Consolidation	26
2.7	Elasticity	28
2.8	Resource Allocation	29
2.9	HPC in the Cloud	30
2.10	Viability Studies on HPC Cloud	32
2.11	Studies on Facilitating the Use of HPC Application in the Cloud	34
2.12	Optimizing the Performance of HPC Clouds	35
2.13	Challenges Facing HPC Cloud	36
2.13.1	Cost vs Performance	38
2.13.2	Service Level Agreement	39
2.13.3	Energy Consumption in HPC Cloud Data centres	40
2.14	Solutions to the Challenges Faced by HPC Cloud	45

2.14.1	HPC-Aware Schedulers	46
2.14.2	VM Consolidation	47
2.14.3	Hybrid Local Regression Host Overload Detection Method (HLRHOD)	48
2.14.4	HPC Cloud Workload Prediction	49
2.15	Precis and Analysis of Literature Review	56
2.16	Summary	64
CHAPTER 3	RESEARCH METHODOLOGY	65
3.1	Overview	65
3.2	The Literature Review	67
3.3	Analysis	68
3.4	Design of the Proposed Energy-Aware Multi-Dimensional Online Bin Packing (EAMDOBP) Technique	70
3.4.1	Energy-efficient and HPC-Aware Proactive Dynamic VM Consolidation in Cloud Computing	71
3.4.2	Energy-Aware Multi-Dimensional Online Bin Packing (EAMDOBP) Architecture	73
3.5	Implementation	77
3.6	Evaluation Methodology and Experimental Setup	79
3.6.1	Power Model	82
3.6.2	Performance Metrics	83
3.6.3	Workload	85
3.6.4	CloudSim Architecture and Modelling	85
3.7	Summary	87

CHAPTER 4	EVALUATION AND RESULTS	89
4.1	Overview	89
4.2	Simulation Results and Analysis	89
4.2.1	Number of VMs	90
4.2.2	Algorithms Comparative Analysis	90
4.3	Contribution and Significance	94
CHAPTER 5	CONCLUSION AND RECOMMENDATIONS	96
5.1	Introduction	96
5.2	Conclusion	96
5.3	Future Research Recommendations	98
5.4	Closing Remarks	98
REFERENCES		100

LIST OF TABLES

	Page
Table 2.1: Comparisons of the Top-3 Cloud Vendors Based on Strength and Weakness	22
Table 2.2: Summarized Findings of Some Feasibility Studies (Netto et al., 2018)	33
Table 2.3: Summarized the Findings of Studies on the Usability of HPC Cloud (Netto et al., 2018)	35
Table 2.4: Literature Review for Gap Analysis	57
Table 3.1: The Original and Extended Classes of CloudSim	81
Table 3.2: Power Consumed by the Chosen Hosts at Various Load Levels in Watts	83
Table 3.3: PUE Efficiency Values	84

LIST OF FIGURES

	Page
Figure 2.1: Convergence of Various Advances Leading to the Advent of Cloud Computing (Adamuthe & Thampi, 2014)	15
Figure 2.2: IBM Presentations on Diverse Variations of Cloud Computing Workloads (Mateo, 2011)	17
Figure 2.3: Cloud Computing Service Models: IaaS, PaaS and SaaS (Muratore et al., 2018)	18
Figure 2.4: A Typical Data Centres Network	24
Figure 3.1: Research Methodology and Outcomes	66
Figure 3.2: Empirical Research Methodology	67
Figure 3.3: Energy Comparison of MDOBP vs Default Scheduler	69
Figure 3.4: Inputs and Outputs of (a) Static VM Consolidation and (b) Dynamic VM Consolidation	71
Figure 3.5: The Proposed Energy-Aware Multi-Dimensional Online Bin Packing (EAMDOBP) Architecture	73
Figure 3.6: Energy-Aware Multi-Dimensional Online Bin Packing (EAMDOBP)	77
Figure 3.7: CloudSim Architecture (Redkar & Guidici, 2011)	86
Figure 3.8: Communication Flow between Various Entities in CloudSim	87
Figure 4.1: Algorithms Comparison for Metacentrum HPC Workload, where the Simulation time and Throughput metrics are depicted in Log-Scale	91
Figure 4.2: Algorithms Comparison for Metacentrum HPC Workload, where the CPU, RAM and Bandwidth Utilization metrics are depicted in Log-Scale	92
Figure 4.3: EAMDOBP Results	94

LIST OF ABBREVIATIONS

AWS	Amazon Web Service
CPU	Central Processing Unit
DVFS	Dynamic Voltage and Frequency Scaling
HaaS	Hardware as a Service
HPC	High-Performance Computing
IVMPP	Interference-Aware Virtual Machine Placement Problem
MPI	Message Passing Interface
NaaS	Network as a Service
CRM	Customer Relation Management
PaaS	Infrastructure as a Service
PUE	Power Usage Efficiency
QoS	Quality of Service
QoE	Quality of Experience
SaaS	Software as a Service
SLA	Service Level Agreement
STaaS	Storage as a Service
TCA	Total Cost of Acquisition
VM	Virtual Machine
VMM	Virtual Machine Monitor

CHAPTER 1

INTRODUCTION

1.1 Study Background

The popularity of cloud computing has increased because of its many advantages available to customers. Cloud computing provides on-demand computing services, including infrastructure, platforms, software, and database for paying users (Sundarakani, 2019). It is considered an evolution enabling workload management and remote resource interaction. Cloud services' increasing popularity is mainly because of their unique features, including the possibility of resource sharing among cloud providers and users in a fee-based model. This feature is helpful for the users as it enables them to have a lower cost of ownership and an increasable capacity of dedicated infrastructures when necessary (Guyon, 2018).

High-Performance Computing (HPC) refers to using supercomputers, grid environments, and clusters of computers to solve computation-intensive problems (Regassa, (2022). HPC applications are regularly used for weather forecasting, aircraft crash simulation, and computational fluid dynamics for aerodynamics studies, among other computation-intensive purposes (Marques, 2021). Although HPC systems are challenging to obtain by research communities because of their high cost, advancement in computing technologies has enabled the use of HPC applications in the cloud, a revolutionary computing paradigm (Mohammadian, 2020). Running HPC applications on the cloud is a significant advancement that offers many advantages to the users.

In this sense, the HPC cloud uses cloud resources to administer HPC applications (Mell & Grance, 2011). The main advantage of HPC cloud computing is that it provides HPC infrastructure to individuals and businesses with no physical space, financial resources or technical and operational ability to construct their HPC infrastructure. Thus, small start-ups can obtain enormous computing power (Fernandes et al., 2016). Nevertheless, combining these two domains, HPC and cloud, is not short of shortcomings. Executing HPC applications on the cloud faces many challenges, including changing workloads. HPC operates in dedicated hardware with batch processing. HPC applications usually need direct physical access to the hardware (i.e. multi-core machines, grid and supercomputers) to have the best achievable performance. Accordingly, the traditional HPS platforms are fine-tuned perfectly to achieve the best parallel processing performance. Transferring HPC applications directly to the cloud will likely cause performance penalties and thus extinguish the advantage gained from the cloud services. Also, the increasing complexity of modern data centres increases concerns over energy usage, power efficiency, thermal stresses and cooling needs about the operating costs and their effect on the environment and societies (Singh & Chana, 2016). Specifically, a large-scale HPC application would ideally need dedicated cloud resources (compute and network). A cloud provider's per-hour charge for dedicating the resources would quickly make the proposition uneconomical for customers.

In most cases, servers work at 10-50% of their total capacity, causing additional costs on over-provisioning and, in this manner, additional Total Cost of Acquisition (TCA) (Egwutuoha et al., 2014). Since the end of the 2000s, cloud computing has become the subject of many studies as it drew the attention of researchers to its increasing maturity level. This thesis hypothesises that the presently available clouds are compatible with only

some HPC applications, for which clouds are considered more cost-effective than the traditional dedicated HPC platforms when applying intelligent scheduling. Based on the evaluation of performance and analysis, it was found that many HPC applications are not perfectly compatible with cloud environments (Tavares, 2021). The poor interconnect and I/O performance in the cloud, network virtualization overhead, HPC-agnostic cloud schedulers (Gupta et al., 2018), and the inherent heterogeneity and multi-tenancy in the cloud are some obstacles to the effective running of HPC applications in the cloud (Zhou et al., 2021). This study attempts to bridge the gap between HPC and clouds by proposing a complementary approach of clouds HPC aware through virtual machine consolidation.

Power awareness is a problematic issue in HPC as it is a crucial factor in operational costs. For efficient management of resources, virtual machine consolidation is proposed to control the overloading and underloading of physical machines (PMs) caused by the random distribution of VMs. Mainly, the VM consolidation process handles the movement of some VMs from the overloaded physical machines (PMs) to the underloaded PMs. Dynamic Virtual Machine consolidation is considered an efficient process to minimize energy consumption and enhance resource utilization in the data centres while maintaining the quality performance of HPC and non-HPC applications on the same platform. Because the problem of VM consolidation is mainly related to NP-hard, several heuristic algorithms have been suggested to solve the issue. However, all the proposed algorithms do not consider the variable nature of workloads in HPC and non-HPC applications in the cloud.

This introductory chapter presents the background to the study, which is focused on the High-Performance Computing (HPC) application performance in the cloud and

explores the best techniques that would enable the effective utilization of a shared pool of resources for efficient execution of HPC and Non-HPC applications in the cloud. Proactive, dynamic application-aware VM consolidation strategies are examined to check how to be tailored to application characteristics. The chapter starts with an overview of the topic under study, followed by the problem background, problem statement, research objectives, research scope, significance, thesis organization and summary.

1.2 Problem Background

Running HPC applications on the cloud is accompanied by the problem of wasting energy and resources. The exponential demand for the cloud and its dynamic nature increase pressure on resource consumption. It is estimated that servers consume 0.5% of the total electricity usage worldwide. According to the International Energy Agency, data centres utilize 1% of all global energy demand and will consume 1/5 of the world's power supply by 2025 (IEA – International Energy Agency, n.d.). Data centre power utilization efficiency is evaluated in units of Power Usage Effectiveness (PUE). A PUE of 1.0 indicates complete efficiency (i.e., all consumed energy is used only on IT equipment, with no power distribution losses). However, it is nearly hard to achieve. According to the most recent annual Uptime Institute survey, the average power use effectiveness (PUE) ratio for a data centre in 2021 was 1.58, only modestly better than seven years previously (George, 2022).

The main drawback of cloud computing is its lack of a stringent energy efficiency regime (Sigh & Bal, 2017). This flaw is even aggravated when executing HPC applications on the cloud because they use more resources and computing power than conventional cloud applications (Netto et al., 2018). HPC applications need more computing power

because they need a high CPU capacity, extensive memory and high network speed to execute correctly, and a particular execution mechanism. HPC applications do not run 24×7; instead, they usually are executed in batches (Netto et al., 2018). Simply put, a set of jobs are run by the users who wait until results are generated to decide the need for new jobs. Also, stacking several workloads on the same physical resources is not a proper approach for HPC applications, meaning they need a dedicated cloud that costs a lot of resources (Netto et al., 2018). Accordingly, running an HPC application on the cloud requires special resource allocation and optimizations in infrastructure.

For the HPC cloud applications to perform optimally, they require sufficient resources, which means the provision of adequate resources that are not less or more than needed. Maintaining sufficient resources is a bit difficult because the workload changes from time to time. Because of the users' usage patterns variance, a workload difference is likely to occur. The system must be adaptable to the workload changes to avoid the harm of overprovision or under-provision of resources. On the one hand, resources under-provision leads to violating the Service level agreement (SLA), thus less quality of services and customer dissatisfaction. On the other hand, the over-provision of cloud resources causes energy and cost waste. In this case, the cloud service costs a lot of energy and resources instead of being cost-effective.

The HPC research community has been giving special attention to the cost and computational performance of different cloud services (Egwutuoha et al., 2014). The available VM scheduling mechanisms depend on user inputs and static partitioning of clusters into zones of availability for various application types (such as HPC and non-HPC).

However, this process is problematic because a large-scale HPC application would ideally require a dedicated cloud resource, dramatically increasing operating costs. To overcome this problem, we aim to identify suitable application combinations whose execution profiles complement each other well and can be consolidated on the same hardware resources without harming the overall HPC performance. This way, the solution's business value is maximized by utilizing the hardware, lowering the cost for HPC applications while maintaining performance and profitability (Parashar et al., 2013).

Dynamic consolidation of Virtual Machines (Goscinski et al., 2012) is considered one of the best ways to improve resource utilization while optimizing energy efficiency in cloud data centres. This methodology influences the dynamism of Cloud workloads helping to reduce the quantity of active physical hosts needed to manage the workload while turning the idle hosts to low-power modes to reduce the total energy consumption. The hosts are active only when resources are needed. However, while reducing energy consumption is advantageous, it is essential to satisfy Quality of Service (QoS) requests.

In addition, massive amounts of data are being transferred between the infrastructure and consumers in Cloud applications; therefore, it is essential to view the computing and network aspects of energy efficiency carefully. Migrating VMs based on CPU utilization alone is not accurate enough because they will not be accurate for the network, Input and Output (I/O) and memory-intensive applications (Fernandes et al., 2016). Therefore, developing a solution to enhance energy efficiency based on hybrid factors (CPU, Memory, Bandwidth) is essential.

Many challenges emerge when running HPC applications in the cloud, such as changeable workload, assignment of nodes, random layout, unstable network flows, the

shortage of low-latency interconnects, virtualization overhead optimization and the application-agnostic cloud schedulers (such as MOAB or LoadLeveler) (Skovira, 2000). For an efficient deployment of the HPC application, sharing resources is required. Currently, user inputs and static partitioning of clusters into availability zones for HPC and non-HPC applications are the base of VM scheduling mechanisms, making scheduling the main challenge to efficiently utilising resources. Fixed scheduling modules and strategies are now embedded in the structure of cloud platforms, which sometimes lead to underutilization of cloud resources harming the HPC performance. In the cloud, Software as a service (SaaS) workloads rely on request-level parallelism, which makes them different from HPC workloads that rely on a thread or task-level parallelism. As a result, HPC workloads are considered more communication-intensive and more sensitive to communication latency (Sigh & Bal, 2017). For this reason, more advanced scheduling techniques are required to use a shared pool of resources, providing a reconfigurable system.

1.3 Problem Statement

As discussed in the problem background, modern data centres are getting more complicated, which complicates the problem of energy usage, power efficiency, maximizing the performance of the workload and efficient utilization of resources. Thereby, there is a crucial need for dynamic state-of-the-art techniques to schedule a fusion of HPC and non-HPC applications smartly to maximize resource utilization and energy efficiency. Nevertheless, existing HPC cloud providers still rely on a separate dedicated pool of resources for HPC and non-HPC to maintain the high performance for HPC applications. Furthermore, depending on the variation in the input flux of HPC workload results in either under-utilized or over-utilized provision of resources in a cloud data

centre. The trade-off between improved HPC performance vs better resource usage makes scheduling for HPC and non-HPC applications challenging. Considering these issues, the problem statement of the research is mentioned as follows:

How to develop dynamic scheduling techniques to classify suitable co-allocated applications (HPC and non-HPC) that can be executed on the same pool of resources while guaranteeing optimum utilization of resources, thus avoiding cross-interference, concurrently ensuring the high performance of applications and at the same time energy efficient?

In this study, it is hypothesized that the placement of VMs to physical machines, while considering the dynamic nature of applications and energy, will efficiently utilise virtualized cloud data centres.

Accordingly, several research questions can be pulled out from the problem statements are stated as follows:

- i. How the suggested proactive, dynamic VM scheduling technique aids in overcoming the trade-off between optimum utilization while boosting the performance of HPC and non-HPC applications using a single pool of cloud data centre resources?
- ii. How can the suggested proactive dynamic VM scheduling technique help with energy efficiency issues so that cloud data centre resources can be used to their maximum capacity while satisfying SLA?

1.4 Objectives

Accordingly, the primary goal of the proposed work is to propose a proactive dynamic VM scheduling method with the ability to intelligently co-locate diverse applications (both HPC and non-HPC) according to their needs for cloud data centre resources to achieve improved resource and energy usage

This goal of the study will be attained by achieving the following research objectives:

- i. To propose an intelligent, proactive dynamic VM scheduling technique to optimize resource utilization in a cloud data centre;
- ii. To test proactive dynamic VM scheduling technique for improving the efficiency of the cloud data centre; and
- iii. To evaluate the energy and efficiency of the proposed proactive dynamic VM scheduling technique.

The proposed study encourages the cloud providers to use the same cloud data centre for all the applications simultaneously rather than using two Cloud data centres, one for HPC applications and one for non-HPC applications. Cloud providers can cut costs by preventing underutilization of resources during off-peak hours since HPC applications do not always employ all the data centre resources if they dynamically use the same data centre with the aid of a scheduler.

1.5 Scope

Cloud computing is a recent operational model that combines a group of existing technologies in a significant way, such as grid computing, Peer-to-peer (P2P), and Web applications (Zhang et al., 2010). High-Performance Computing (HPC) refers to using supercomputers, grid environments, or clusters of computers to solve computation-intensive problems (Kuo-Bin, 2003). The two domains are broad and have been the subject of many studies. This study focuses on HPC in the cloud, which refers to using cloud resources to run HPC applications. Typically, HPC applications need dedicated infrastructure to be executed based on locally-owned resources, which are expensive to obtain and operate. Therefore, HPC users seek a more cost-effective and flexible solution, which is a cloud. However, the effective combination of the two domains: HPC applications and Cloud Computing, is facing some challenges. For example, the underlying cloud infrastructure typically limits the tuning of HPC applications. In addition, HPC large-scale tightly coupled applications need faster networks than most cloud networks. Also, stacking several workloads on the same physical resources is not a proper approach for HPC applications, meaning they need a dedicated cloud that costs many resources. As a result, resources are used inefficiently. This study aims to make the most of cloud data centre resources while minimizing application performance and energy usage. This research focuses on mixing HPC and non-HPC applications on the same physical system to complement each other. As a result, they can share the same pool of resources without sacrificing performance. This research seeks to optimize the performance of cloud resources to execute HPC application-focused either on the infrastructure or resource management. There have been severe approaches to formulating scheduling policies based

on resource usage and performance predictions. This study focuses on the opportunities and challenges of dynamic VM consolidation for HPC in the cloud.

1.6 Significance of the Study

By achieving the objectives of this study, a significant gap in the field of HPC cloud will be filled. Cloud computing has many advantages, including better service and cost reductions. It also prevents the job queuing of the traditional HPC systems as the Cloud computing pay-as-you-go pricing model is available for lease as Infrastructure as a Service (IaaS) and Hardware as a Service (Haas) for computation-intensive applications (Egwutuoha et al., 2014). Because of these advantages, more computation-intensive applications will run in HPC systems in the cloud. However, for small-scale applications, public clouds are the optimum solution because of being cost-effective, but it is not so for supercomputers (i.e., HPC (Parashar et al., 2013)). Significant problems hinder the great potential offered by the cloud for HPC applications. Among the problems concerning cloud computing are two strongly related to the HPC applications: reduced performance stability and limited effective energy-efficient solutions. The energy consumption problem is even more complicated because the HPC applications are typically implemented in big data centres, increasing the power load (Leite, 2015). Although developing an effective and integrated solution to the problem of energy and cost overconsumption has been a concern of many studies, it is yet to be realized.

The goal of the proposed study is to offer a way to efficiently arrange a combination of HPC and non-HPC applications in the cloud-based data centre in order to maximize resource usage and enhance energy efficiency. Developing energy and HPC-aware scheduling algorithm for proactive dynamic VM consolidation will achieve better

resource utilization and energy efficiency and limit cross-application interference through careful co-location. This way, the HPC cloud will be cost-effective and ensure the optimum execution of the applications at all times.

1.7 Organization of Thesis

This study investigates a significant aspect of the HPC cloud by attempting to present reliable energy and HPC-aware scheduling algorithm for dynamic VM consolidation in the cloud. The complete research is organized into four chapters. Chapter 1 is introductory and starts with an overview of the topic under study, followed by problem background, problem statement, research objectives, the research scope, significance, thesis organization and summary.

Chapter 2 includes the literature review on the topic of the study with an in-depth analysis of the related studies that cover cloud, HPC applications, problems regarding the combination of the HPC and Cloud domains, Service Level Agreement (SLAs), and models existing in the literature. Chapter 2 also explains the limitations of the existing models and hinders that might be faced while attempting to design and develop an enhanced model. Chapter 3 elaborates on the research methodology adopted to achieve the present study's objectives and aim. Chapter 3 also presents the description and design of the energy and HPC-aware scheduling algorithm for dynamic VM consolidation. Chapter 4 includes the application of the model, the discussion of the research and research conclusion, limitations and future direction, which are critically discussed and justified in Chapter 5.

CHAPTER 2

LITERATURE REVIEW

2.1 Overview

This chapter introduces the literature review conducted to lay the ground knowledge necessary to meet the study's objectives stated earlier. The chapter includes definitions of High-Performance Computing (HPC) and Cloud Computing. Explanations of central concepts and elements related to the cloud are explained, such as cloud workload, types of cloud service, virtualization, and more. Then an overview of studies conducted on the HPC cloud is presented, categorized into three main parts: viability studies, studies on facilitating the use of HPC in the cloud and studies on optimizing the performance and energy effects of HPC applications in the cloud. The relevant studies are presented with a significant focus on Hybrid host overload detection algorithms presented by other researchers as they are the most relevant to the proposed work.

2.2 Definitions of HPC and Cloud

In the past twenty years, the need for computation power has surged explosively, showing an exponential growth pattern. Supercomputers originated in the 1960s to resolve problems like intensive calculations, brain patterns, weather forecasting, simulations spanning various fields, and many more (Tomašević et al., 2020).

2.2.1 High-Performance Computing (HPC)

High-Performance Computing (HPC) uses supercomputing and parallelism to solve complex problems (Morales-Hernández et al., 2020). High-performance computing (HPC) can also be defined as several supercomputers, computer clusters, or computing

systems linked through commercially available interconnects to handle large-scale science and engineering problems. The majority of these applications are developed with Message Passing Interface (MPI) (The Message Passing Interface (MPI) standard, 1991) or OpenMP (The OpenMP API specification for parallel programming, 1997) to make use of the parallel processing power of these systems. HPC environments focus on achieving the best performance possible using the available infrastructure.

2.2.2 Cloud

Cloud computing is considered a game changer in the IT industry because it has transformed the business model by modifying how things can be accessed, computed, managed, stored and delivered. Cloud technologies are often considered newcomers despite being developed in the early 2000s and are a crucial player in numerous, widely-used technologies like networking, grid computing, distributed computing, parallel computing, etc. (Atlam et al., 2018). It is an innovative, on-demand service that uses the pay-as-you-go model, like power, water, and light (Zhang et al., 2018). Cloud computing is among the main reasons for the rapid spread of data centres worldwide (Sun et al., 2019). Cloud computing is defined by the National Institute of Standards and Technology (NIST) as “a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service interaction” (Atlam et al., 2018). Cloud can also be defined as “a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.” (Archana, 2018).

Figure 2.1 illustrates cloud computing variations and related technologies from 2004 – 2014. This is based on the views of Gartner in their report on the Strategic Technology List. Figure 2.1 shows the drivers of changes.

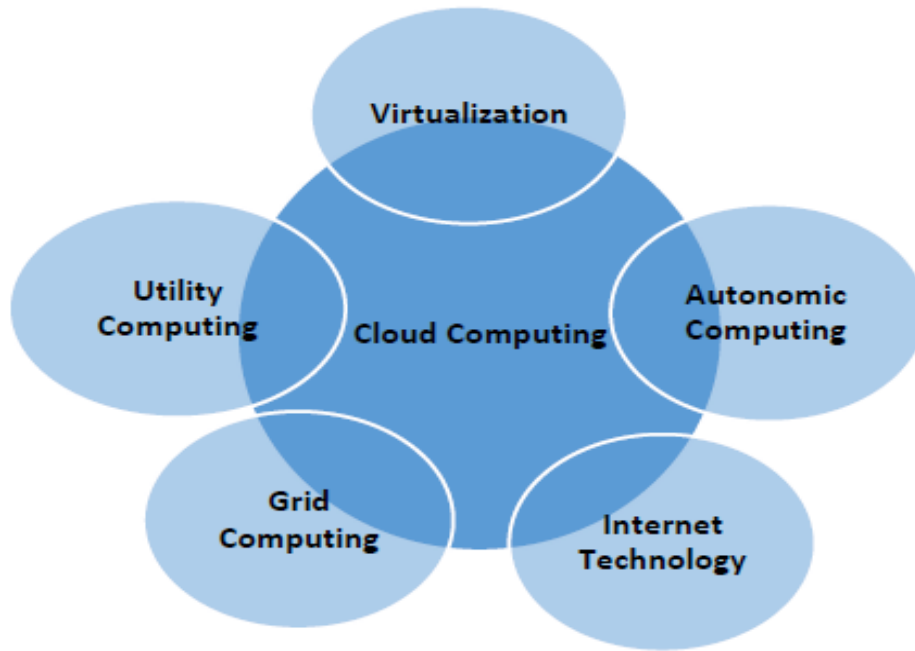


Figure 2.1: Convergence of Various Advances Leading to the Advent of Cloud Computing (Adamuthe & Thampi, 2014)

As revealed in Figure 2.1, Cloud is regarded as a more advanced technology than computer networks, but it works to facilitate workload management and interaction with remote resources (Netto et al., 2018). Clouds are better than other technologies because cloud increases profits, reduce the workforce, provide variable solutions for any scientific or business problem, decrease the complexity of resource connectivity and provide better productivity (Zhang et al., 2018). Gartner stated five attributes of the cloud: first, service-based means that resources in the cloud system become available through well-defined services. Second, clients consume the resources they need (scalability and elasticity).

Thirdly, all resources are shared because the assets are used until their maximum capacity but are not owned by the clients. Fourthly, clients pay for the service they get. Finally, cloud services are a form of internet technology (Mvelase et al., 2018).

2.3 Cloud Workload

A cloud workload can be described as a service, a specific application, a capability or a specific amount of work that can be run on a cloud resource. Typically, cloud workload is received by the cloud data centre as jobs sent by the users. Each job contains self-defining characteristics, including the submission time, user identity and resource requirements related to CPU, disk space, memory, and more. One job could include more than one task scheduled to be executed in the Cloud. Cloud workloads perform differently in different server architectures and vary more based on CPU cores than the RAM capacity of the machines at the server level (Daradkeh et al., 2020). Significantly, cloud computing is advantageous for its unique proprieties which as “self-scaling” and “elasticity”, which indicate that the system can adapt available resources to the changeable workload over time while maintaining the efficient utilization of these resources. Self-scaling needs a manual installation and provision of resources and depends on the early detection of problems (Zhang et al., 2018). Cloud computing differs from traditional data centres because the former can become more energy efficient by utilising techniques such as virtualization and consolidation workload (Renugadevi et al., 2020). The workloads with high demand are not considered more suitable candidates for cloud infrastructure. Figure 2.2, extricate from IBM presentations on cloud computing, demonstrates the compatibility of diverse variations of cloud computing workloads (Mateo, 2011). In the new era of cloud, with the high demand and proliferation of cloud services, there is a vital need to enhance the current cloud architecture to fill the gaps that help in the deployment of various natures

of workload in a more efficient manner that supports in better utilization of cloud data centre resources. Furthermore, until now, little consideration has been paid to characterizing cloud workloads.

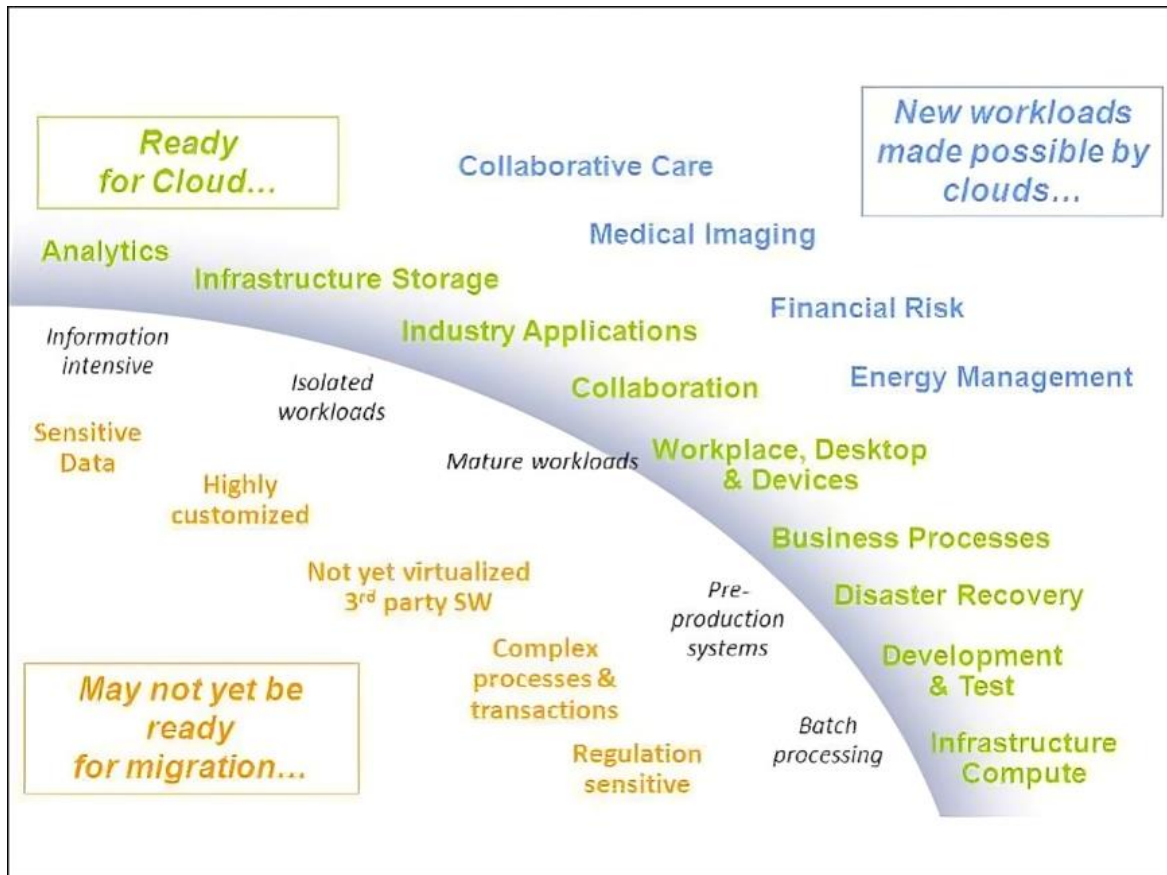


Figure 2.2: IBM Presentations on Diverse Variations of Cloud Computing Workloads (Mateo, 2011)

2.4 Types of Cloud Services

As shown in Figure 2.3, Cloud services offered to users include Software as a Service (SaaS), Platform as a Service (PaaS), Infrastructure as a Service (IaaS), Hardware as a Service (HaaS), Network as a Service (NaaS) and Storage as a Service (STaaS).

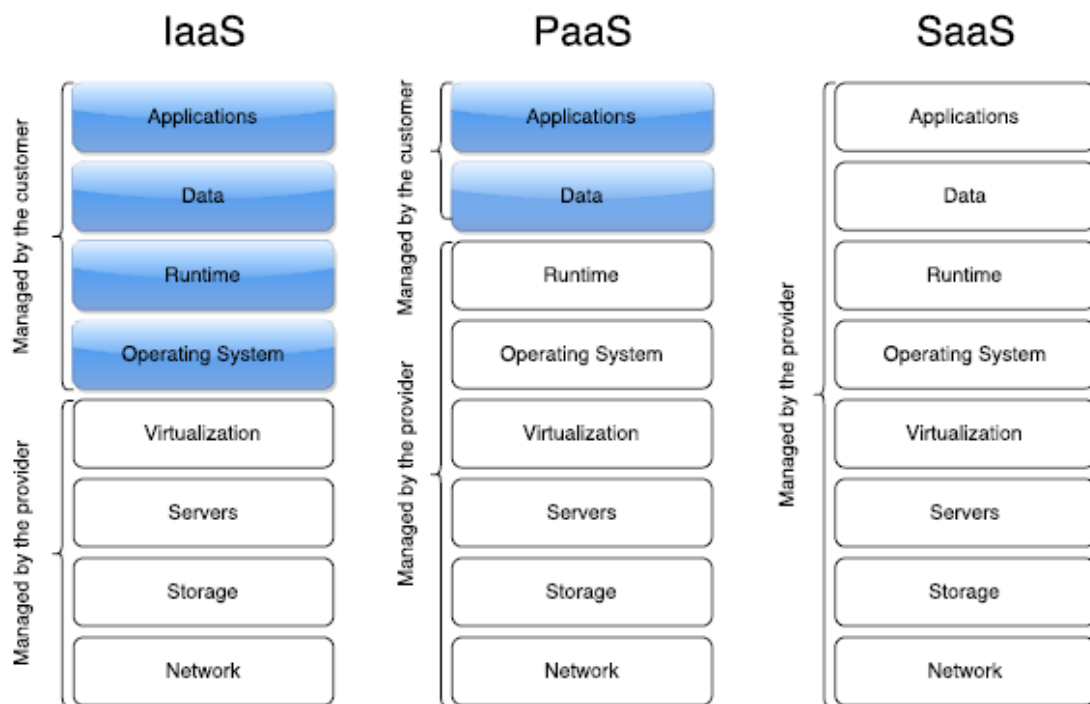


Figure 2.3: Cloud Computing Service Models: IaaS, PaaS and SaaS (Muratore et al., 2018)

There are two main types of clouds: private clouds and public clouds. The provider and the customer are the same entity in the private cloud, such as big companies and universities. Here, there are no monetary charges for using resources and the resources are not shared between different organizations. The public cloud is thus different as it is owned by one provider and used by several customers who pay per hour to rent machines. The resources used by users in a public cloud are shared among several users giving rise to the concept of multitenancy, where users are referred to as tenants (Gandhi, 2020). Amazon EC2 (Amazon EC2. <http://aws.amazon.com/ec2/>, 2018), Google Compute Engine (Google compute engine. <https://cloud.google.com/products/>, 2018), HP Cloud (HP cloud services. <http://hpccloud.com/>, 2018), Rackspace (Rackspace. <http://rackspace.com>, 2018), and

Windows Azure (Windows Azure. <http://windowsazure.com>, 2018) are examples of public clouds.

Typically, one significant entity like Amazon or Google manages and owns a cloud. However, several customers use the cloud for various purposes, such as students who run code on virtual machines and giant companies such as Netflix or NASA, who use thousands of cloud machines to deploy their services (Alkayal, 2018; Mvelase et al., 2018). Cloud is advantageous for users because it saves them the cost of purchasing their physical resources by depending on virtualization. The following are some notable and well-known modern cloud systems:

Amazon Web services (AWS): AWS is a complete IaaS system that provides web services with a few services that make AWS a PaaS and SaaS provider (Marozzo, 2018). It contains two main components: Amazon Elastic Compute Cloud (EC2) and Amazon Simple Storage Service (S3) (Amazon simple storage service., n.d.). EC2, an IaaS cloud, provides customers with computing resources such as computer power, memory, bandwidth, and more to create VMs. EC2 enables customers to amend their resource requirements and adjust the settings of the complete software stack, including platform, application, and more (Mvelase et al., 2018). The Simple Storage Service S3 offers scalable storage with a REST/SOAP interface to host applications' data (Amazon Web Services, 2021).

Google App Engine: Google App Engine provides PaaS infrastructure for customers to develop and run applications. The simple web application framework called “web app” enables customers to develop, maintain and upload their applications. The

Python programming language is the base for Google App Engine's present runtime environment (Mahesh et al., 2019).

Microsoft Windows Azure: Microsoft Windows Azure provides PaaS infrastructure that customers can use to develop, host, scale and manages web applications (Mahesh et al., 2019). Currently, Windows Azure provides only the .NET framework, REST (71), HTTP, and XML-based APIs. IaaS services are also provided that enable customers to create and maintain VMs. Moreover, Azure also offers data storage services accessed by ADO.NET or LINQ (Tulloch, 2013).

Salesforce: Salesforce offers SaaS cloud products and services (Vo, 2019). It dominates CRM and enables enterprises to add applications to its Customer Relation Management (CRM) product (Vo, 2019). Salesforce also offers PaaS services to developers who want to create applications to run on the Salesforce cloud platform. It currently provides the Eclipse integrated development environment (IDE) and APEX (Gupta et al., 2018).

Eucalyptus: Eucalyptus is an open-source cloud computing system that enables researchers to conduct experiments and improve the software infrastructure. Initially, it was a research project at the University of California, Santa Barbara. Eucalyptus cloud comprises a node, cluster, storage, and cloud controller. Node controllers help create VMs based on the available resources on a node. The cloud controller is used for task scheduling and allocation. The cluster controller is responsible for interconnectivity among variable nodes in a cluster and VMs. The controller maintains Eucalyptus's storage feature (Sharma & Rai, 2019).

OpenNebula: OpenNebula is an open-source cloud platform. The European Union sponsored the development of OpenNebula as a private cloud characterized by incredible customizability, and it can also be used as a hybrid cloud. It is perfect for organizations that want to quickly establish a cloud using VM technology with a few machines. VMware, KVM, and Xen hypervisors are among the virtualization tools supported by OpenNebula (Sharma & Rai, 2019; Vashistha & Dwivedi, 2018).

AbiCloud: AbiCloud offers a standard, open-source and highly interoperable cloud platform. It enables the moving of applications among different clouds. AbiCloud is unique because users can duplicate and implement a real cloud in another place. AbiCloud consists mainly of three elements: AbiCloud server, AbiCloud Web services and Virtual System Monitor. AbiCloud server facilitates the managing of the computer centre and user interface. AbiCloud Web Services is responsible for virtual applications. Finally, Virtual System Monitor is a plug-in monitor manager. Interestingly, AbiCloud includes a web-based management tool that is very user-friendly (AMD IO-virtualization technology IOMMU, n.d.).

Nimbus: Nimbus is an open-source cloud system that provides its resources as IaaS. Nimbus is very customizable and consists of four components: workspace service, workspace control, workspace resource management and workspace pilot. Web services are maintained with GSI (Grid Security Infrastructure) authentication by workspace service. Workspace control develops OS/system images and connects with hypervisors. VMs are managed through Workspace resource management. Finally, virtualization in the cloud's clusters is possible via a workspace pilot. Nimbus has three prominent roles; (1) To allow users or businesses to build private or community IaaS clouds, (2) To ensure users

can use the IaaS cloud and (3) To support developers by allowing them to experiment with the IaaS cloud (Vashistha & Dwivedi, 2018).

OpenStack: OpenStack is an open-source and open standard cloud infrastructure. It is a community comprised of many registered developers and contributing companies. However, it initially provides IaaS services, including three main components: OpenStack Compute, OpenStack Object Storage, and OpenStack Image. OpenStack Compute distributes VM instances. OpenStack Object Storage allows storage, and OpenStack Image conducts the operations required for managing virtual disk images (Sharma & Rai, 2019).

Reviewing a comparison of cloud resources in the fast-changing cloud landscape is a challenging job. There are thousands of cloud services, hundreds of cloud service providers and dozens of IaaS service providers that provide pay-as-you-go model pricing, each of which regularly changes and updates its portfolios. However, Table 2.1 demonstrate the strength and weakness of the top-3 cloud vendors:

Table 2.1: Comparisons of the Top-3 Cloud Vendors Based on Strength and Weakness

Vendor	Strengths	Weaknesses
AWS	AWS owns an Influential market position on the global scale by offering Intensive and mature assistance as well as adequate training for numerous corporations to boost cloud adoption.	AWS's prime drawback is the price reduction, and the cost structure of AWS for running high data and workloads is a crucial concern for industries.

Table 2.1: continued

Microsoft Azure	Even though Microsoft Azure was still late on the cloud market, they are the second-largest cloud service provider. The key reason for the Azure cloud platform's competitive success is its existing Windows users, which can simply integrate cloud solutions without any hurdles.	The only downside observed is that some Azure business-grade applications are poor and less ready for business than anticipated.
Google	Google is a reliable and suitable cloud platform since it can handle huge workloads, dynamic machine learning and quick response time.	The shortcoming is an alternative solution rather than a strategic platform provider. In addition, more open-source centring organizations prefer using Google cloud.

2.5 Virtualization

Cloud providers depend mainly on virtualization technology, enabling them to consolidate several users' workloads simultaneously in the same physical space (Liu et al., 2020). The concept of virtual machines (VMs) is based on the nature of a cloud that enables the running of diverse instances of operating systems on a single physical machine (host) (Zhang et al., 2018). In this sense, multiple VMs hosts many operating systems, thus generating flexible cloud applications. Applications in cloud computing are executed over several computers connected by a network, usually a data centre network. It indicates high line rates, low latencies and a tree-like topology. Also, every physical machine in the network includes variable virtual machines, as shown in Figure 2.4 below (Gandhi, 2020):

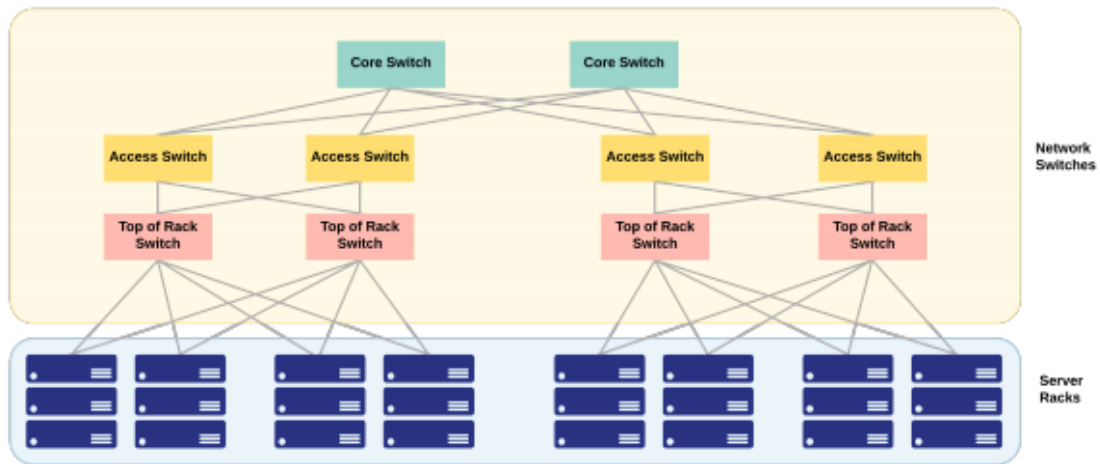


Figure 2.4: A Typical Data Centres Network

Virtualization is an essential component of cloud computing. It is a software form of physical resources. Virtualization allows users to create several virtual machines in one physical machine. It is a means by which the cloud provides resources securely, quickly and efficiently. Virtualization includes three main techniques: full-virtualization, para-virtualization and hardware-assisted virtualization. Also, the hypervisor is the main component of cloud virtualization. It is called a hypervisor or Virtual Machine Monitor (VMM) that manages the hosted VMs. The hypervisor is the mediator between VMs and the hardware resources in a physical machine. Each host machine must include minimally one hypervisor to manage its VMs. With a CPU scheduler, hypervisors typically CPU schedule VMs (Rahman, 2016).

2.6 Virtual Machine Allocations

Many tools help plan and implement VM allocation based on CPU, memory and disk usage, such as VMWare Capacity planner (Capacity planner, n.d.), Microsoft Assessment and Planning (MAP), or IBM Workload Deployer (Family 7199+03 IBM workload deployer V3.1, n.d.). Nevertheless, these do not consider network usage or VM

intercommunication. Accordingly, VMs that communicate extensively with each other are often far away from each other. Thus, they communicate long-distance, increasing the load on the network's higher levels, including the most expensive equipment. For example, Facebook has about 1000 times higher traffic usage within the data centres than the traffic coming and going among its users (Jobava et al., 2018). This decreases the data centres' communication performance and power consumption (Vakilinia, 2018).

Efficient VM provisioning allows appropriate execution of applications and load balancing (Rahman, 2016). Significantly, resources can be allocated to VMs in two ways: static provisioning and dynamic resource allocation (Muhammad & Song, 2018; Wang et al., 2017). The dynamic means adapting allocated resources to the work need. The data centres are equipped with a fixed number of resources; thus, VM provisioning needs extreme care. An application with insufficient computing resources can suffer from inefficient performance, lack of availability, or in the worst case scenario, crash entirely. Hence cloud provider needs to allocate appropriate resources per the applications' nature.

2.6.1 Static VM Provisioning

Cloud providers use SLA terms to decide the amount of infrastructure allocated for running VMs and the cost charged to the users. The allocation of infrastructure is usually based on provisioning algorithms. Static provisioning is suitable for applications with a stable workload. In this case, the cost of placing VMs into clouds is paid once before running the user's applications. Also, the required hardware resources remain fixed, contrary to the dynamic VM provisioning, where the allocation of resources to a VM is changeable to adapt to unexpected workload fluctuations. Most static provisioning algorithms aim to place the VMs in the least possible number of physical machines. Static

provisioning comes in two types: online algorithms such as Next Fit, First Fit, Best Fit, and Worst Fit and batch algorithms (Rahman, 2016).

2.6.2 Dynamic VM Consolidation

Dynamic VM consolidation is ideal when applications' workload is variable; thus, migrating VMs to new compute nodes within the cloud is necessary. Significantly, workload variation comes in three types: short-term, medium-term, and long-term, and each cannot be handled using the exact mechanism. Thus, dynamic provisioning is helpful as it helps the system adapt to the changes. However, VMs migration causes runtime overhead and execution delays (Rahman, 2016). Dynamic provisioning means that additional resources are allocated or removed as needed.

Remarkably, dynamic provisioning depends mainly on predicting future workload. Many researchers attempt to provide solutions for efficient dynamic provisioning of resources. Kim et al. (2011) introduced a model for predicting dynamic VM provisioning depending on the previous VM resource usage patterns (Alshammari, 2018). Rolia et al. (2005) also utilized information about previous use to forecast future resource needs for applications (Gachunga, 2018). This approach sometimes causes a decrease in the number of active nodes, which decreases resource consumption and power. Sharma and Maurya (2019) suggested that “ghost VMs”, which are spare and constantly active, would allow the quick assigning of required resources to run VMs (Sharma & Maurya, 2019). Khanna et al. (2006) introduced a model to measure the physical and virtual machines' resources to check whether their use of VMs is more than the predefined limit (Alkayal, 2018). Accordingly, a VM is migrated by the system to another physical host. Dynamic provisioning is challenging as it is difficult to choose the VMs that should be moved and

the time of moving them. VMs migration to a new node is not always the optimum solution, especially in case of a short-time workload peak, as VMs migration, in this case, causes additional management overhead (Rahman, 2016). Dynamic VM consolidation comes in two different types: reactive and proactive. When the initial placement is adjusted as the systems reach a particular undesired condition, it is called reactive VM consolidation.

On the other hand, proactive VM consolidation relies on prediction algorithms focused on historical data. Prediction algorithms allow early changes in the system to endure the change in the workload (Masdari & Zangakani, 2020). (Beloglazov & Buyya, 2015) suggested heuristic algorithms for all four classifications in the VM consolidation technique. The proposed techniques are host overload detection algorithms such as Median Absolute Deviation (MAD), Interquartile Range (IQR), and Regression. Random Choice (RC), Minimum Migration Time (MMT), and Maximum Correlation (MC) are the algorithms proposed for VM selection. Finally, they presented Power Aware Best Fit Decreasing (PABFD) for VM placement based on the bin-packing problem. These algorithms are developed as adaptive heuristics, and this is the only study that examined all four categories of VM consolidation compared to any other study done under the Virtual machine consolidation (VMC) paradigm. (Chowdhury et al., 2015) solve the issue of server consolidation using Modified Worst Fit Decreasing (MWFD), a method that places a VM on the host with the greatest increase in power consumption. MWFD saved more energy than MBFD. In this VM placement technique, VMs are ranked in decreasing order depending on a host's CPU use, and VMs are assigned to hosts with the highest gain in power consumption. This is in contrast to the PABFD algorithm.

2.7 Elasticity

Elasticity means immediate automatic response to fluctuant workload. In this way, the resources increase or decrease automatically. These resources can be scaled by changing the number of a resource, the location of the resource or by increasing or decreasing the size of a resource. Elasticity can also be defined as a technique that helps the system adapt to changes occurring in the workload by making resources available or unavailable to match the user's needs (Runsewe, 2019). Typically, cloud platform users need a variable number of resources, which the provider will offer according to the actual demand. Resource provision is made elastically (Feng et al., 2019). Elasticity is significant for systems that occasionally face peak demands (Feng et al., 2019). Thus, the most efficient operation of the cloud can only be realized if the platform can assign resources promptly, which makes actual demand and resource cap closely aligned (Feng et al., 2019).

Significantly, inaccurate allocation of sources can cause the running of many processes for a single available resource, while at the same time, other types of resources are unutilized. This causes overload and underutilization, increasing wait times, enormous power usage, and SLA violations. Quality of Service (QoS) is a central point of negotiation between the cloud provider and the customer while making the Service Level Agreement (SLAs). The quality agreed upon by the two parties should be provided; otherwise, the provider will be penalized (Alkayal, 2018). In Elastic cloud systems, the usage of the primary sources by CPU, memory, disk space, bandwidth, etc., are the main variables to consider. Thus, they must be carefully examined, forecasted and managed for the system to provide adequate resources (Feng et al., 2019).

2.8 Resource Allocation

Resource allocation is always a challenging issue in all fields as it is responsible for the scale up and down of resources (Nayudu & Sekhar, 2018; Alkayal, 2018). Resource allocation means allocating available resources to cloud services on the internet in cloud computing. More specifically, resource allocation in the cloud is assigning hosts in a cloud provider's infrastructure to execute customers' applications while efficiently using resources according to the predefined goals (Alkayal, 2018).

When virtualization is applied to the cloud's infrastructure in data centres, resource provisioning is done by turning computational resources into virtual machines (Nayudu & Sekhar, 2018). Each host's CPU number determines the number of VMs this host can handle. Significantly, the tasks are usually allocated to virtual machines before they are assigned to hosts. Thus, resource allocation involves two levels (Alkayal, 2018). Based on their requirements, the tasks are first allocated to suitable virtual machines, which are then assigned to proper physical machines (Alkayal, 2018). The allocation of virtual machines is an excellent advantage as it is possible to move a virtual machine from one host to another, called VM migration. This feature maximises resource utilization and performance improvement and reduces power consumption (Zhang et al., 2018). The elasticity, scalability, and increasing resources and costs make the current resource allocation algorithms not efficient enough (Alkayal, 2018). Specifically, traditional algorithms are inaccurate because cloud resources are based on virtualization. The main challenge of resource allocation in the cloud is the discrepancy among hardware capacities, workload prediction, and the quality of service required by the cloud consumer (Alkayal, 2018).

2.9 HPC in the Cloud

The emergence of high processing power computers, fast network connections, and Linux enabled using computer clusters in High-Performance Computing (HPC) areas at the beginning of the 90s (Netto et al., 2018). The process is operated by batch schedulers who receive user requests to run jobs. In cases of high workload, the jobs are queued as Service Level Agreements (SLAs) are not part of this service, so users do not have to worry about the costs of running jobs. The problem with significant clusters is that they are costly and can lead to resource wastage and poor quality of services if not appropriately managed (Netto et al., 2018). With technological advancement, grid computing was used in HPC environments in the early 2000s. Grids are advantageous because they provide users with enormous resources managed by autonomous administrative domains (Netto et al., 2018). However, the service was not always of high quality, which made users depend on best-effort policies to run applications. In the late 2000s, cloud computing started to gain popularity and was the subject of many studies to execute HPC applications on remote cloud resources. Cloud infrastructure presented an excellent solution for running HPC applications as it allowed the utilization of external infrastructure, which means lower cost. Also, in traditional HPC environments such as the HPC system with the batch queue, grids, etc., the job submission time may be quite long, as it may be minutes or even hours. This is different in the cloud, as HPC workload queues and waiting times do not exist (Malla & Christensen, 2019). High-Performance computing clouds are gaining popularity over on-premise clusters for services of scientific applications and business analytics, which indicates their advantages (Netto et al., 2018)

All three levels of cloud computing, namely IaaS, PaaS, and SaaS, can be helpful for HPC. Nonetheless, the most common form adopted by HPC applications is the IaaS

paradigm (Mvelase et al., 2018). IaaS service allows users to execute applications on fast pay-per-use machines without the need to buy, manage or maintain these machines. IaaS is based on virtualization technology.

The main properties of HPC clouds can be summed up in the following points:

First, HPC clouds are web service based as Web APIs are used for software processes and Webform for human operators. Second, HPC clouds are pay as you go model. Thirdly, the HPC cloud is elastic as the resources are adapted to the workload. HPC cloud users can use the elasticity mechanism to adjust their resource pools in the cloud. Fourthly, performance is guaranteed to clients. Fifthly, HPC clouds use virtualization for operation. Besides, HPC clouds are easy to use (Mvelase et al., 2018).

Some studies supported the feasibility of performing HPC in the cloud using standard benchmarks such as High-Performance Linpack (HPL), NAS Parallel benchmarks, NERSC benchmarks, and more (Muhammed et al., 2020). Besides, various disciplines run their HPC applications in the cloud, including the High-energy physics domain, geographic/ seismic domain, electronics design, bioinformatics domain, media and gaming domains, and large-scale engineering simulation studies (Muhammed et al., 2020). Microsoft's Dan Reed stated that "fortunately, the emergence of cloud computing, coupled with powerful software on clients, offers a solution to this conundrum. The net effect will be the democratization of research capabilities now available only to the most elite scientists" (Mvelase et al., 2018).

2.10 Viability Studies on HPC Cloud

Many viability studies were conducted to investigate the metrics utilized to examine how viable it is to use the HPC cloud, resources utilized in the experiments, computational infrastructure and software. These studies focus on CPU, memory, storage and disk performance by conducting analyses based on HPC benchmarks, microbenchmarks and HPC user applications (Netto et al., 2018). Most studies comparing HPC cloud and on-premise clusters used Amazon Web Services (AWS) because Amazon was the first provider of a simplified use of cloud resources for individuals and small organizations (Amazon, 2017). One of these studies was made by Gupta et al. (2012), who concluded that the public cloud saves costs for small-scale applications (Netto et al., 2018; Gupta et al., 2012). However, clouds can complement supercomputers for large-scale applications. Besides, they indicated that the scalability of applications in the cloud is limited by network latency. Other studies concluded that the cloud could be used efficiently for a subset of HPC applications (Netto et al., 2018; Gupta & Milojicic, 2011). Significantly, Gupta et al. (2012) compared the performance of HPC benchmarks of clusters, grids and a private cloud and concluded that communication-intensive HPC applications suffer from degraded performance in the cloud while other HPC applications perform well (Netto et al., 2018; Gupta et al., 2012; Muhammed et al., 2020). Also, the cloud is cost-effective only for HPC application that does not require high-performance networks. In this regard, Belgacem and Chopard (2015) examined running multi-scale large parallel applications on hybrid HPC clouds and concluded that HPC hybrid clouds are suitable for large applications (Netto et al., 2018). Another study indicated that the turnaround time is four times longer in HPC on-premise than in the cloud (Netto et al., 2018). However, they also stated that deciding between the use of cloud and local HPC

clusters is complicated and should be based on application scalability and the goals of optimizing cost or turnaround time. Berriman et al. (2010) examined the benefit of a cluster at the National Centre for Supercomputing Applications (NCSA) versus Amazon EC2 cloud offers (Netto et al., 2018). They concluded that for CPU and memory-intensive applications, the cloud offers more power and is cheaper than their on-premise cluster. For other applications that do not consume a large volume of data, clusters offer better performance. They also shed light on the attempts to create an academic cloud, which they believe will offer different levels of services than those provided by commercial clouds. Also, it is stated in the Magellan Report sponsored by the US Department of Energy (DoE) that some workloads witness slowdown that amounts to 50 times when executed in clouds (Netto et al., 2018). However, they also highlighted that the cloud benefits parallel applications as the high-performance cluster queuing systems do not provide proper support to this class of applications.

One of the advances in HPC cloud technology is the emergence of HPC-optimized cloud resources. One of the most comprehensive studies on this topic is the Magellan report, which included various architectures, workloads, a metrics (Netto et al., 2018). Most viability studies investigating the public cloud as large-scale workloads are generally not published in academic papers.

Netto et al. (2018) summarized the findings of some viability studies in Table 2.2:

Table 2.2: Summarized Findings of Some Feasibility Studies (Netto et al., 2018)

Aspect	Findings
Cost	Cloud is more advantageous than on-premise clusters when running small applications

Table 2.2: continued

Throughput	Because of the lack of queueing systems in on-premise clusters, the cloud provides higher throughput for running single applications with low communication requirements
Resources	The leading causes of low HPC performance in the cloud are network virtualization and hardware heterogeneity; thus, improving infrastructure positively influences HPC application performance.
Network	Considering the effect of network speeds regarding variable communication models of parallel applications, the cloud is preferable when loosely-coupled or embarrassingly-parallel applications are used.

2.11 Studies on Facilitating the Use of HPC Application in the Cloud

The fundamental reasons for moving HPC workloads to the cloud are the cost-benefit achieved and the optimization of the execution of these workloads. However, these benefits will be compromised if the HPC cloud is difficult to use. The aim behind abstracting away the infrastructure from HPC users is to create user-friendly HPC platforms where users are not required to be aware of the underlying cloud infrastructure to use the HPC applications. They need to submit the applications, relevant parameters, and QoS expectations through a web portal, and resource provisioning, application scheduling, deployment, and execution will be handled by middleware (Netto et al., 2018). The purpose of studies on the usability of the HPC cloud is to promote the adoption of HPC by new users who do not understand the system administration and configuration of complex computing environments. Belgacem and Chopard (2015) indicated that a hybrid infrastructure consisting of clusters and clouds enhances execution time only if the application is adjusted to adapt to the difference in CPU speeds between clusters and

clouds and that between local networks and Wide area network (WAN) connections (Netto et al., 2018). Cost management is related to HPC cloud usability.

In Table 2.3, Netto et al. (2018) summarized the findings of many studies on the usability of the HPC cloud as follows:

Table 2.3: Summarized the Findings of Studies on the Usability of HPC Cloud (Netto et al., 2018)

Aspect	Findings
Web portals	It is hard for users to choose cloud resources; thus, user productivity can be increased by developing portals that abstract details.
Execution Steering	Attempts to transform job creation into an automatic process facilitate the HPC cloud usage for non-IT specialists.
Workflow management	Models of variable cloud pricing decrease the time and costs for deploying and exposing HPC applications in the cloud.
Application deployers	HPC applications are different from traditional cloud applications because the former has a different software stack; thus, systems that consider HPC application requirements decrease the time to solve in clouds.
Legacy-to-SaaS	Legacy applications run in cloud environments instead of traditional computing platforms suffer from inefficiencies because of the different infrastructure assumptions. However, principled methodologies can prevent such inefficiencies.

2.12 Optimizing the Performance of HPC Clouds

The studies on optimising the performance of HPC clouds examine the infrastructure or resource management. Regarding the infrastructure, the studies focus on networking because it is the leading cause of the failure faced while executing HPC

workloads in the cloud. However, studies on resource management focus on proposing scheduling policies. There have been strenuous efforts to examine the optimization of resource allocation for HPC clouds, such as spot instances and elasticity. In this sense, resource usage and performance predictions are helpful for optimization. These studies focus mainly on the challenges faced when running HPC applications in the cloud.

2.13 Challenges Facing HPC Cloud

Performing HPC in the cloud is faced with some criticism from people experts in the field. For example, Dr Iordache was against the idea belief that performance is central for HPC while more latency is the expected result of virtualization in clouds (Mvelase et al., 2018). However, this statement is inaccurate because virtualization is not about latency but only provides a needed platform independent of the currently used physical platform. However, one of the drawbacks of the cloud is the lack of best practices or standards. The absence of best practices or standards in the cloud is a significant drawback as it impedes the development of the system (Netto et al., 2018). Thus, HPC cloud users do not have standards to refer to, so they redundantly write their applications (Netto et al., 2018). Besides, Dr Kranzmuller, chair of the HITS Scientific Advisory Board (SAB), expressed his rejection regarding running HPC in the cloud, saying that “there are only minimal uses for the public cloud in HPC for applications with larger storage requirements or closely coupled parallel processes with high I/O requirements, clouds are often useless” (Mvelase et al., 2018).

Moreover, the director of NERSC, Dr Yelick, stated that “there is a part of the workload in scientific computing that’s well-suited to the cloud, but it is not the HPC end, it is the bulk aggregate serial workload that often comes up in scientific computing, but

that is not the traditional arena of high-performance computing” (Mvelase et al., 2018). One of the main problems of the HPC cloud is that it is an open field, which enables cloud providers to use the same physical resources for several workloads to achieve economies of scale. This approach harms the HPC application's performance (Netto et al., 2018). Furthermore, among the main problems of the cloud are that applications sometimes run slower than customers expect, and customers cannot determine the number of machines to request for running their applications. Besides, customers’ traffic affects the bandwidth available for each other to use, which might affect the speed of cloud networks (Gandhi, 2020). These problems become even worse for network-intensive applications such as HPC.

Also, the efficient use of the cloud for executing High-Performance Computing (HPC) applications is sometimes hindered because of the difference in the properties of the HPC applications compared to the traditional standard enterprise and Web applications (Netto et al., 2018). The main problems of running HPC applications in the cloud are memory management issues, data locality, data mobility, data storage, legal aspects and security vulnerability concerns. Besides, HPC application issues appear when allocating jobs to cloud resources, mostly virtual machines (Benkhelifa et al., 2019). Ye et al. (2012) indicated that the most prominent factors impeding the optimum performance of clouds are the disk I/O, process latencies, and overhead of network communication in multiple virtual machine systems (Benkhelifa et al., 2019). Moreover, adequate resource provisioning is challenging because of the dynamism of cloud resources, inaccurate resource prediction mechanisms, poor memory management in VMs, and poor response time (Witte et al., 2019). Many HPC applications need accurate handling of both virtualized and physical resources because they need processing resources like CPUs and GPUs and produce a

significant amount of data (Webscope datasets, n.d.). Data moves in HPC Cloud within and geographically distant data centres. Among the significant challenges in HPC Cloud is thus load balancing. All these challenges can be summed up in the following points.

2.13.1 Cost vs Performance

The main benefit of the HPC cloud is the unlimited availability of resources, which prevents queues in a cluster when resources are overloaded. However, HPC hardware is costly compared to traditional clouds. Thus, placing users in such hardware would be wasteful for cloud providers. The challenge, then, is to provide sustainable business for cloud providers via economies of scale while offering high-quality performance for users (Netto et al., 2018). Besides, cloud clients sign a Service-Level Agreement (SLA) with the service provider as computing service users, ensuring they get a defined quality of service. Maintaining the specified level of quality at low costs is the ultimate goal of cloud providers. If the workload is underestimated, the user pays lower costs because of the use of less hardware but will get substandard performance during peak demand time. This substandard performance will cause SLA violations, negatively affecting the Cloud provider's reputation. Contrariwise, when the workload is overestimated to maintain performance at the high workload, the resources will usually be underused, and the user will incur the high cost of unutilized infrastructure (over-provisioning). The two cases are the most prevalent challenges in cloud workload prediction (Zeng et al., 2017).

Many efforts have been made to balance cost and performance, such as in Kocoloski et al. (2012). They presented a system named dual-stack virtualization, which comprises a VM manager hosting HPC and allocates VMs based on the level of isolation required by the users (Netto et al., 2018). Also, there have been some efforts to introduce

flexible resource rental models for the HPC cloud. For example, Zhao and Li (2012) introduced a model based on strategies for planning depending on on-demand and spot instances (Netto et al., 2018).

Moreover, running HPC in the cloud is different from other traditional cloud applications, and the cost advisors available for such users are useless for HPC. Thus, many efforts have focused on introducing a tool by which users can predict costs for running HPC applications in the cloud. For example, Rak et al. (2015) presented a framework called mOSAIC to predict the performance and costs of running HPC in the cloud (Netto et al., 2018). With this framework, the users can depend on simulation and benchmarks to forecast the future performance of the applications during the development stage.

2.13.2 Service Level Agreement

Service Level Agreements (SLAs) are the base upon which VM placement decisions are made. The use of the cloud typically includes three parties: infrastructure resource providers (IaaS), business service providers and end-users. Business service providers pay IaaS providers for renting computing resources, and end-users pay business service providers for implementing their requests. SLA states the level of performance and related costs, which the parties agree to through negotiations. SLA should always be maintained to avoid violations leading to penalties. CloudWatch, CloudKick, Monitis, and LogicMonitor are commercial SLA monitoring tools (Rahman, 2016). In addition, some researchers developed models that can monitor, detect and cope with SLA violations automatically (Prasad & Bhavsar, 2020). To determine whether VM SLAs are met, VM resource usage information, including CPU, memory, disk and network usage data, is

necessary. Automatic management is necessary for a cloud system to make decisions about resource allocation, detect SLA violations and suggest proper solutions for resource bottlenecks and SLA violations. For example, Padala et al. (2009) introduced automatic resource control through the "Auto-control" system. Auto-control can allocate and reallocate resources based on the needs of the applications running at the time (Qu et al., 2018; Singh et al., 2019; Moreno-Vozmediano et al., 2019).

2.13.3 Energy Consumption in HPC Cloud Data centres

The main objective of Cloud providers is to provide optimal application performance while reducing energy consumption, which is a significant roadblock (Muhammed et al., 2020). However, cloud data centres consume energy and transmit equal carbon footprints. A data centre is where an organisation's servers, computer systems and IT equipment are placed and managed. It is the facility where the organization keeps its data. According to Gartner's IT Glossary:

"The data centre is the department in an enterprise that houses and maintains back-end information technology (IT) systems and data stores—its mainframes, servers and databases. In the days of large, centralized IT operations, this department and all the systems resided in one physical place, hence the name data centres" (Data centre, n.d.). Data centres have rapidly developed since the dot-com bubble of the 1990s' when companies realized the importance of their internet presence and searched for efficient methods to establish their IT system to have readily available fast internet connectivity. In 2007, the amount of energy consumed by data centres became significant as one average data centre needed as much energy as a small town (Khan et al., 2020). The problem is even aggravated as in 2016, data centres consumed about 3% of all electricity in the United

States, in 2018, data centres consumed about 1% of the electricity consumed worldwide (Ayanoglu, 2019).

In addition, energy consumption is considered the costliest element in any HPC platform because the average energy consumed depends on the server's workload throughout the day. Barroso and Hölzle (2007) researched and concluded that 90% of the time, servers are running at a capacity below 100%. This means that the servers consume maximum energy regardless of the workload (Jouppi et al., 2018). Cloud providers are committed to applying the Service Level Agreement (SLA) (Nayudu & Sekhar, 2018). The SLA states the conditions of the Quality of service (QoS) and the Quality of expectations (QoE) for the end-user. Typically, client demands are met in the Cloud by provisioning resources more than the amount needed to complete the prospective job because if resource forecasting goes wrong, the SLA will be violated, affecting the QoS and QoE (Panneerselvam, 2018). Users' requests arrive at the data centres as job submissions and are scheduled onto Virtual Machines (VMs) installed on physical servers. The intensity of the demand of the users determines the resource levels allocated by providers for task execution. Thus, the levels of resources allocated are not usually wholly utilized, which makes the physical servers work below their standard capacities. This makes a significant percentage of the server resources unutilized. Specifically, around 46% of machines do not use their maximum capacity for long periods. When these machines work for tasks lower than their capacity, they use much energy with no significant benefit. Servers, in this case, are called “comatose” or “zombies” servers (Daradkeh et al., 2020). Studies indicate that unutilized servers use about two-thirds of the power consumed by the same server work at full load (Renugadevi et al., 2020).

Decreasing the active resources at the data centres reduces energy consumption, which may lead to the temporal loss of services, thus breaching the agreed SLA and harming the provider's reputation (Daradkeh et al., 2020). The unavailability of services is not tolerable for Cloud users because they have bought a contract stating all the agreed terms and requirements. The inaccurate estimation and understanding of the attributes of the data centres and the Cloud workloads are the main reasons for the low efficiency of the available methodology of energy allocation (Renugadevi et al., 2020). Thus, quantifying the workload, data centres' performance, and corresponding energy implications for the entire system will improve the execution of HPC applications in the cloud.

For determining the causes of energy-wasting, studies indicated that long tails, task failures, rollback, performance interference, latency sensitiveness of the workloads, and different run-time tasks are among the reasons for overconsumption of energy (Panneerselvam et al., 2018). Analyzing workload characteristics has many benefits, including understanding the internal distribution of resource consumption and task length, which allows the exposure of job behaviour heterogeneity in resource consumption. Also, determining the percentage of the requested concerning the utilized resources helps determine the idle resource time. Significantly, energy efficiency has witnessed some improvement by power consumption management (Lin et al., 2018). Efficient resource allocation algorithms are needed to decrease the system's total power consumption (Alkayal, 2018). Reducing power consumption is the request of all cloud providers as it means spending less cost on cloud infrastructure (Zhang et al., 2018). Energy-saving can be reached by providing techniques to allocate VMs to a fewer number of hosts. In this case, the unutilized resources can be shut down to save power (Alkayal, 2018).

Overconsumption of energy is witnessed at different levels in a large-scale data centre's environment. Several studies have been conducted to examine the impact of energy over-consumption. Some researchers indicated that the shared kill and evict cases, failure and repair times and termination in Cloud data centres cause energy waste (Panneerselvam, 2018; Gao et al., 2019; Daradkeh et al., 2020). In addition, the network, dispatch and in-house computing latency of tasks also contribute to excessive energy consumption (Panneerselvam, 2018). Garraghan et al. (2015) examined long tails and referred to their effect as lengthening the time needed to complete the entire task, which typically consumes more energy (Panneerselvam et al., 2018). Yang et al. (2013) investigated the impact of performance interference on energy consumption, which was reduced using the workload placement model (Panneerselvam, 2018). In addition, some studies stated that communications connecting the users to the cloud consume a lot of energy (Panneerselvam, 2018).

Reducing energy consumption in a data centre is very challenging because of the complexity of systems utilized in a cloud data centre. Hardware efficiency is not the only element contributing to energy consumption. Resource management systems and the application's efficiency play a significant role in the energy consumed (Chaurasia et al., 2018). Proactive dynamic VMs provisioning is an effective way to use resources appropriately and reduce energy consumption. Proactive dynamic VMs provisioning is only achieved based on historical data and workload prediction. VM requirements prediction is the base to decide whether or not to use additional physical machine resources (PM), reorder query execution, and accept or reject a new incoming query (Ismaeel et al., 2018).

Many researchers proposed different methods to save energy wasted in cloud data centres. In Muhammad and Song (2018) and Wang et al. (2017), they have suggested energy-saving algorithms. They first aimed to distribute the VMs in a way that saves power. Typically, the VMs are allocated evenly to all the servers, which constantly work regardless of how much data they have to run. This causes power waste as all the servers are running all the time. The other algorithm, thus, ensures that the servers already on receive more loads so that they get fully utilized, and extra servers are not needed. Hamzaoui et al. (2020) suggest auto-scale allocation using a technique they developed based on future resource usage prediction by feeding the past usage data to the Bayesian network. First, training data is fed to the Bayesian network to prepare it for future patterns. Then, the network receives the actual usage data to predict future usage. Liu et al. (2018) compared several prediction models based on past data. All of them are learning models that must be trained first using training data to ensure prediction models are minimized.

Chen et al. (2020) suggested a new mechanism, including an algorithm for VM placement and a method for frequency scaling. Virtualization effectively limits energy waste because it enables an individual physical machine to host many virtual machines. Similarly, Dynamic Voltage and Frequency Scaling (DVFS) is a new technology for saving energy where cloud managers could reduce performance interference by making careful decisions about VM placements. Cloud managers have to choose whether to limit performance interference by combining VMs with different characteristics or to make better use of DVFS by combining VMs with similar features. The experiments resulted in a 60% reduction in the total energy-delay product. Alves et al. (2018) analyzed the Interference-aware Virtual Machine Placement Problem (IVMPP) in Small-scale HPC Applications running in clouds. They explain that the purpose of this problem is to

minimize the interference faced by small-scale HPC applications and the number of physical machines. Cross-interference usually occurs when applications run on a mutual physical machine, which damages the performances of the applications in HPC running in clouds; this problem often happens. As a solution to eliminating the interference-aware Virtual Machine Placement Problem, the iterated local search framework is suggested. Alves et al. (2018), minimize the interference in HPC applications when standard physical machines are shared. Indeed, more than a 40% reduction in interference was achieved by this method, according to the results. Significantly, the consolidation effect and energy were not considered. Souza et al. (2019) proposed a hybrid resource management system for HPC and Data-Intensive (DI) systems to combine both on the same platform. The most critical feature between HPC and DI systems is that a fixed set of resources is assigned entirely to an application in HPC. They proposed a hybrid system allowing dual-level scheduling for DI jobs in HPC infrastructure. Slurm and Mesos resource managers are the base for HPC and DI jobs. Resource consumption is increased by the hybrid architecture by 20% in a specific cluster with a 12% reduction in queue makespan while meeting all the deadlines for HPC jobs. However, they ignored the interference, co-location effects, energy waste, and future resource utilization.

2.14 Solutions to the Challenges Faced by HPC Cloud

For businesses in all sectors, high performance computing (HPC) is essential because it enables them to accelerate innovation and profit. Although cloud-based HPC systems are becoming more widely used, there are still some misconceptions about their cost, security, and performance. To avoid holding back enterprises of all sizes due to outdated, incorrect information, it is critical to dispel these myths and remove typical obstacles to cloud-based HPC. The proposed study revealed these obstacles and shows

how cloud-based HPC addresses each concern, paving the way for business to start reaping the rewards of cloud-based HPC.

2.14.1 HPC-Aware Schedulers

Gupta et al. (2013) presented an HPC-aware scheduler for cloud platforms that considers the HPC application's topology requirements (Netto et al., 2018; Gupta et al., 2013). This scheduler uses benchmarking information and determines the kind of network requirement of the applications and how its performance is influenced when resources are shared with other applications. The scheduler shows 45% performance improvement when compared to an HPC-agnostic scheduler. As for the high cost of running HPC in the cloud, Marathe et al. (2014) designed and developed techniques to decrease such costs (Netto et al., 2018). They implemented strategies for determining bid prices for Amazon EC2 spot instances and scheduling checkpoints for user applications.

Somasundaram and Govindaraan developed a framework to schedule HPC applications on a cloud platform to meet users' deadlines and reduce the costs of executing their applications (Netto et al., 2018; Alkayal, 2018). This scheduler is based on Particle Swarm Optimization. Marshall et al. (2013) explored elasticity as a cloud functionality for HPC workloads (Netto et al., 2018). Using Torque and Amazon EC2, they developed a prototype in which jobs would be received by machines in the cloud in case of the unavailability of enough resources at the job submission moment. Intending to make use of the advantages of clouds, grids, and clusters, Mateescu et al. (2011) introduced a concept called "Elastic Cluster" (Netto et al., 2018). This concept offers intelligent infrastructure and workload management, considering the different types and locations of resources and application performance requirements. Cloud providers prefer consolidating applications

into the same hardware to save costs. To ensure the optimum performance of a cloud environment, continuous efforts have been exerted to increase inter-VM isolation and reduce VM overhead (Netto et al., 2018). In addition, Scheduling and fixing VMs processes for physical servers have become popular in the cloud context. This is because an efficient scheduler allows significant energy waste reduction.

2.14.2 VM Consolidation

The problem of energy over-consumption in the HPC cloud can be approached by VM consolidation. Energy consumption could be reduced by switching idle nodes to low-power modes (i.e., sleep), thus preventing idle power consumption. In addition, live migration can dynamically consolidate VMs to the least number of physical nodes based on their resource requirements. The strong positive relationship between a server's total energy consumption and CPU utilization has encouraged many researchers to develop models that provide an accurate prediction for CPU- intensive applications, and they ignored network, I/O and memory-intensive applications (Dhiman et al., 2010).

Dynamic Virtual Machine (VM) consolidation is promising in reducing energy consumption by dynamically fixing the number of active machines based on demand. Different algorithms are proposed to achieve VM consolidation, including adaptive utilization-based algorithms and regression-based algorithms. However, both suffer from certain limitations: adaptive utilization-based algorithms generate a poor prediction of host overloading, while regression-based algorithms are complex because it depends on offloading the overloaded host by selecting VMs to prevent performance degradation.

The process of offloading the overloaded host in regression-based algorithms is repeated until the host is not overloaded. The VM selection during the migration is based

on three policies. Minimum migration time (MMT), Random Choice (RC), and Maximum correlation (MC). The minimum migration time policy focuses on migrating VMs that need the least time to complete migration compared to the other VMs specified for the same host. The random choice depends on selecting a VM to be moved from the host based on a uniformly distributed discrete random variable. Regarding the maximum correlation, the VMs selected for migration have the highest CPU utilisation correlation with other VMs (Beloglazov & Buyya, 2012).

2.14.3 Hybrid Local Regression Host Overload Detection Method (HLRHOD)

Local regression (Cleveland & Loader, 1996) is useful in modelling a relation between a predictor variable and a response variable, as shown in Equation 2.1:

$$Y_i = f(x_i) + \epsilon_i \quad \text{Equation 2.1}$$

Where

$f(x_i)$ is an unknown function, ϵ_i is an error term, representing random errors in the observations or variability from sources not included in the x_i .

Local regression depends on fitting simple models to localized subsets of data to construct a curve that resembles the original data. This method relied on Loess's method developed by Cleveland (Cleveland, 1979). The formula (Chen et al., 2012) will help calculate the host utilization based on hybrid factors with a metric indicating the CPU-network-memory combines a load of virtual and physical servers. The host utilization will be calculated then the local regression LR host overload detection algorithm will be used (Beloglazov & Buyya, 2012).

Algorithm: Hybrid Local Regression Host Overload Detection (HLRHOD)

	Input: CPU utilization, RAM utilization and BW utilization
	Output: A decision on whether the host is overloaded, so VM is migrated
1	foreach host in the host List do
2	foreach sample of data do
3	$Y \leftarrow \frac{w1}{1-cpu} \times \frac{w2}{1-memory} \times \frac{w3}{1-net_{node}};$
4	end
5	Use LoessParameterEstimates(Y);
6	predicatedUtilization $\leftarrow$ estimates[0] + estimates[1] * (length + migrationIntervals); predicatedUtilization *= getSafetyParameter();
7	if predicatedutilization < 1 then
8	Repeat step 2 to step 7;
9	end
10	end
11	return predicatedutilization $\geq$ 1 and host is overloaded;

2.14.4 HPC Cloud Workload Prediction

Auto-scaling technology is an excellent way to make use of the scalability nature of cloud systems. This way, cloud providers will automatically manage VM instances (Ghobaei-Arani et al., 2018). Thus, it is necessary to predict the workload to apply resources pre-distribution. Workload prediction is essential for efficient resource pre-distribution and accurate distribution in service clouds.

Wang et al. (2017) categorized the workloads based on the load change speed into slow and fast time-scale data. They explained the effect of the workload on the value of dynamic resizing (Muhammad & Song, 2018; Wang et al., 2017).

Currently, there are two ways to handle the problem of workload fluctuation in the cloud. One way is reactive, which means to react based on the current state of the workload in the system. The second way is predictive, which means workload forecasting. The predictive solutions use ML or mathematical, analytical techniques to predict the workload of making scaling decisions (Runsewe, 2019).

Traditional regression methods and Machine Learning (ML) algorithms have been used to predict input workload. The available predictive systems work on ad-hoc bases and apply several methods that do not depend on state-of-the-art statistical evaluation of the problem from a time-series perspective. A comprehensive forecasting process is proven beneficial in many fields, including healthcare, industry, science, and business (Hyndman & Athanasopoulos, 2018).

Many researchers attempted to provide workload predictors for cloud users and providers to avoid many problems because of workload fluctuation. These are classified by Kim et al. (2016) into five types (Alshammari, 2018):

The first type is the naïve workload predictors, which can be divided into two types mean based and recent. The mean-based predictor predicts the next job's arrival time with the mean arrival time of all earlier executed jobs. The recent mean-based predictor forecasts the arrival time of the coming job depending on the means arrival time of a few recent jobs.

The second type includes regression-based workload predictors, which can be categorized into global and local regressions. There are six regression-based predictors: global and local regression with linear, quadratic, and cubic models (Kim et al., 2016). The global regression models predict the arrival time of the following job based on a linear or polynomial regression model using attributes of all previous job arrival times. These models are single variable regression as job arrival time is considered the primary variable. The local regression models are used in two steps: selecting job arrival samples by applying a Kernel function and using the samples to generate linear or polynomial regression (Alshammari, 2018). The global and local regressions differ in that the job arrival samples used as the base for generating a regression model differ in size and similarity. Local regression utilizes a few samples more similar to the prediction target (Alshammari, 2018). Some researchers use this method. For example, a mixed model comprised of a reactive model for under-provisioning and a predictive model for over-provisioning was introduced by Iqbal et al. (2011) (Ilushkin et al., 2018). They depended on polynomial regression to forecast the number of web and database server instances for the specified workload.

The third type includes temporal-based workload predictors. Time-series predictors are popular means for forecasting workload patterns. Several temporal methods for resource provisioning in cloud computing include ARMA, ES, AR and ARIMA (Alshammari, 2018).

Hybrid local regression host overload detection method (HLRHOD): Because of its simplicity, Local Regression (LR) is considered the most attractive host overload detection algorithm. However, a Multi-factor local regression is more accurate as it considers

multiple factors of Central Processing Unit (CPU), Random Access Memory (RAM), and Bandwidth (BW).

Local regression (Urul, 2018) is used to represent a relationship between a predictor factor and response as in the following Equation 2.2:

$$Y_i = f(x_i) + \epsilon_i \quad \text{Equation 2.2}$$

Where

$f(x)$ is an unknown function

ϵ_i is an error term that shows random errors in the observations or variability from sources that do not exist in the x_i .

Local regression means using simple models representing concentrated subsets of data to formulate a curve that calculates the original data. The LOESS model was initially proposed by Cleveland (1979) and further developed by Cleveland and Devlin (1988), who proposed their locally-weighted polynomial regression. A low-degree polynomial is placed in a subset of data at each point in the dataset. By using weighted least squares, the polynomial is placed to add high weight to the points below the points whose response is being estimated. On the other hand, it provides low weight to the faraway points. For the point being computed, the value of the regression function is gained through running the local polynomial using the explanatory variable values for this data point.

Localized subsets of data: the subgroups of the data used for each weighted least-squares fit in Loess are calculated by the nearest neighbours techniques. The input provided by the client is called the smoothing parameter. It indicates how much data is

used to absorb each local polynomial. A smoothing parameter is a number between $(\lambda + 1)/n$ and 1, with λ representing the local polynomial degree. The value of α is the proportion of data utilized in every fit (Abdelsamea et al., 2017). α denoted the smoothing parameter since it manages the flexibility of the LOESS regression function. High values of α result in a smoother function that fluctuates less with the data fluctuations.

Similarly, low α means that the regression function will conform better to the data. The minimal value of the smoothing parameter is not favourable as the regression function might eventually start to get the random error in the data. For many LOESS applications, estimation of the smoothing parameter usually lies in the 0.25 to 0.5 (Abdelsamea et al., 2017).

Degree of local polynomials: the local polynomials suitable to each data subcategory are usually of the first or the second degree, meaning they are either locally linear or locally quadratic. This basic local model can function properly in certain circumstances. LOESS relies on the notion that any function will likely be adjusted in a small neighbourhood by a low-order polynomial, which can easily fit the data. High-degree polynomials could attempt to overfit the data in each subcategory and are numerically unstable, making computations challenging (Abdelsamea et al., 2017).

Weight function: the weight function allocates the least weight to the faraway data points, and the most weight to the data points nearest to the estimation point. The notion of the weight function indicates that points close to one another in the explanatory variable space are most likely to be related to each other than faraway points. The application of this notion means that points following the local model have a significant effect on the local model parameter estimates. In any localized data subcategory, the weight is

determined by calculating the weight function at a distance around that point and the estimation point. The maximum absolute distance for all of the points in the data subcategory is precisely one. The usual weight function used for LOESS is the tricube weight function shown below in Equation 2.3:

$$W(x) = \begin{cases} (1 - |x|^3)^3 & \text{for } |x| < 1 \\ 0 & \text{for } |x| \geq 1 \end{cases} \quad \text{Equation 2.3}$$

For example, if the degree of the polynomial fitted by the method is 1, then the parametric family of functions is $y = a + bx$. The line is fitted to the data utilizing the weighted-least-squares technique with weight $w_i(x)$ at (x_i, y_i) . The values of a and b are determined by minimizing the function presented in Equation 2.4:

$$\sum_{i=1}^n w_i(x) (y_i - a - bx_i)^2 \quad \text{Equation 2.4}$$

Loess' (3) Local regression host overload detection algorithm (Mason et al., 2018) is used. For each new observation, there is a new trend line $f(y) = a + by$ where $f(y)$ is the predicted Utilization and a and b are estimated (0) and (1), respectively. This trend line predicts the next observation $f(x_{k+1})$. The technique shows that the host is overloaded, and several VMs could be migrated from it if the inequalities in Equation 2.5 are satisfied.

$$sf(x_{k+1}) \geq 1 \quad \text{Equation 2.5}$$

Where s is the safety parameter.

Moreover, Magalhaes et al. (2015) presented a workload model based on the CloudSim simulator using generalized extreme value/lambda distributions (Kecskemeti et al., 2018). They indicated that users' behaviour should be considered in workload modelling. Furthermore, CloudSim is a cloud simulator based on Java. It was developed by the Cloud Computing and Distributed Systems (CLOUDS) Laboratory of the University of Melbourne. Its main aim is to act as a generalized, extensible simulation framework for modelling, simulating, and experimenting with new Cloud computing infrastructure and application services. CloudSim enables researchers and developers to test specific design issues they want to examine. It is helpful in many ways. For example, it can be used as a model and simulator for large-scale data centres, server hosts, energy-aware computational resources, and virtual machine allocation policies (CloudSim: A framework for modelling and simulation of cloud computing infrastructures and services, n.d.). Simulators help examine variable algorithms, applications, and policies in a controllable environment before implementation. For example, simulators are especially significant for comparing provisioning policies with the assigned workload. It is cost-effective because obtaining a testing setup is very expensive in the natural environment. In addition, it saves time as simulators run way faster than in real-time. For example, a whole year can be simulated in minutes. CloudSim Java library offers a perfect simulation facility, including all the components of an actual cloud, such as data centres, hosts, VM, policies, and workloads. It even provides researchers in the energy efficiency domain to model the scenarios they want to test to determine energy and cost waste and consumption. Moreover, CloudSim is a constructive simulator useful for analyzing concepts and predicting possible outcomes in various conditions (Modeling & simulation, n.d.). It is advantageous because it generates the output that would be generated in a real cloud from a given input.

2.15 Precip and Analysis of Literature Review

Numerous research has attempted to investigate the present cloud service architecture for running HPC, as shown in Table 2.4. However, there were relatively few strategies that concentrated on offering a way to efficiently schedule a mixture of HPC and non-HPC workloads in the cloud to maximize resource usage while maintaining the performance of applications as well as enhance the energy efficiency of the cloud-based datacentres. The proposed work attempts to close this gap by creating a novel technique for executing HPC and non-HPC applications on a single resource pool in the cloud data centre. To improve applications performance and minimize the trade-off between enhanced performance and optimal resource use.

Table 2.4: Literature Review for Gap Analysis

Researchers	Type of Platform (Dedicated V/S Combined)	Application type (HPC or Non-HPC)	Parameter evaluated	VM scheduling Technique	Study outcome
V. Mauch, M. Kunze and M. Hillenbrand	Dedicated	HPC	Performance of HPC (virtualization and underutilized or overloaded resources)	InfiniBand has QoS mechanisms (based on weighted round-robin scheduling)	Allocation of virtual clusters (isolated from other tenants) using InfiniBand but sophisticated scheduling algorithms are required to support live migration and to use limited QoS resources efficiently
M. Gienger	Dedicated	HPC	Performance and flexibility	Meta scheduling mechanism	The trade-off between user requirements and application performance showed a drop in performance by 14%. Cloud cannot wholly replace HPC
A. El-Moursy, A. Abdelsamea, R. Kamran and M. Saad	Combined	HPC	Performance of algorithms based on energy consumption (energy efficiency, operating cost, CO ₂ emission) and CPU utilization (CPU, memory and network BW utilization)	CloudSim toolkit for time-shared (i.e. Round-Robin) and space-shared (i.e. First Come, First Serve) provisioning techniques	Algorithms resulted in an improvement of 12% in energy consumption and 80% improvement in a metric that constitutes energy consumption, service level agreement violation and number of VM migrations

Table 2.4: continued

Researchers	Type of Platform (Dedicated V/S Combined)	Application type (HPC or Non-HPC)	Parameter evaluated	VM scheduling Technique	Study outcome
Z. Li, X. Yu, L. Yu, S. Guo and V. Chang	Combined	Non-HPC	Energy consumption and quality of service (QoS) with the help of algorithms for host overloading detection, VM selection, and under-loaded host detection	VM consolidation performs VM migration where heuristic or meta-heuristic algorithms are used	The proposed method reduces energy consumption, host overloading risk and improves QoS
H. Jin, H. Qin, S. Wu and X. Guo	Combined	HPC	Performance degradation and interference due to co-scheduling and Shared Last Level cache (SLLC) bottleneck	Cache Contention-Aware virtual machine Placement approach (CCAP) co-scheduled with cache pollution programs to dispatch VMs on the Xen hypervisor	CCAP decreases the SLLC bottleneck and improves the performance of cache-sensitive applications and cache pollution programs when they are co-scheduled
M. M. Alves, L. Teylo, Y. Frota and Drummond	Combined and Dedicated	HPC	Interference and number of physical machines for allocation	Interference-aware virtual machine placement problem (IVMPP) based on Iterated Local Search (ILS)	Interference was reduced by more than 40%

Table 2.4: continued

Researchers	Type of Platform (Dedicated V/S Combined)	Application type (HPC or Non- HPC)	Parameter evaluated	VM scheduling Technique	Study outcome
A. Souza, M. Rezaei, E. Laure and J. Tordsson	Combined (Data Intensive applications or DI and HPC)	HPC	Sharing of resources between DI jobs and HPC and identifying spare resources	Co-scheduling HPC (using the standard HPC scheduler) and DI applications (using Mesos)	Improved resource utilization by 20% and 12% decrease in queuing
A. Shahidinejad, M. Ghobaei-Arani and M. Masdari	Dedicated	Non-HPC	Cloud resource management	Hybrid Imperialist Competitive / k-means-based clustering algorithm to cluster workloads and decision tree algorithm to provision resources	It has reduced cost and response time. Increased CPU utilization and elasticity
Q. Zheng, R. Li, X. Li, N. Shah, J. Zhang, F. Tian, K.-M. Chao and J. Li	Dedicated	Non-HPC	Power consumption and resource wastage (solving VMcP or Virtual Machine consolidation Placement problem)	VMPMBBO (Virtual Machine Physical Machine Biogeography Based Optimization)	VMPMBBO is more efficient, adaptable and extensible
A. Mosa and R. Sakellariou	Dedicated	Non-HPC	Resource utilization (mainly CPU and memory) and SLAVs	Dynamic VM placement solution using genetic algorithm (GA)	There was a 20% improvement in CPU utilization, 40% improvement in memory utilization and 70% decrease in AOSLAVs

Table 2.4: continued

Researchers	Type of Platform (Dedicated V/S Combined)	Application type (HPC or Non- HPC)	Parameter evaluated	VM scheduling Technique	Study outcome
A. Kraemer, C. Maziero, O. Richard and D. Trystram	Dedicated	HPC (migrating cloud jobs to HPC partition)	Number of response time violations without interfering with HPC jobs execution	The convergence area scheduler	Reduced number of response time violations by setting slack factor (priority)
V. Antonenko, A. Chupakhin, I. Petrov and R. Smeliansky	Dedicated	HPC	Sharing the same resources simultaneously to decrease wait and execution times	The researchers need to develop a scheduler so that resources can be shared according to unique metrics for MPI programs	Underutilized resources can be used to run MPI programs. Also, a 20% increase in execution time was observed, which improves CPU utilization and HPC effectiveness
A. Gupta, L. V. Kale, D. Milojicic, P. Faraboschi and S. M. Balle	Dedicated	HPC	Resource utilization, cross-application interference, cost and Service Level Agreements (SLA)	HPC-aware OpenStack Nova scheduler	Enhanced VM placement, 45% improvement in HPC performance, 32% increase in throughput and jitter (noise) was limited to 8%
S. Di, D. Kondo and F. Cappello	Dedicated	Non-HPC	Resource utilization per application is based on how their jobs are executed	The study characterises Google apps and does not discuss VMs or schedulers. It classifies apps using the k-means clustering algorithm	(a) Number of jobs and resource utilization per application follows the Pareto principle (b) All apps can be divided into four types based on batch-task execution

Table 2.4: continued

Researchers	Type of Platform (Dedicated V/S Combined)	Application type (HPC or Non-HPC)	Parameter evaluated	VM scheduling Technique	Study outcome
I. S. Moreno, P. Garraghan, P. Townend and J. Xu	Dedicated	Non-HPC	Resource management, operational conditions, Quality of Service (QoS) and use and task behaviour diversity	The study analyzes the diversity of user and task behaviour using the k-means clustering algorithm	The workload is driven by the characteristics of tasks and behavioural patterns of users. User patterns are more diverse
A. Abdelsamea, A. A. El-Moursy, E. E. Hemayed and H. Eldeeb	Dedicated	Non-HPC	Inefficient usage of resources and CPU, memory and bandwidth utilization	Host overload detection algorithms (MRHOD or Multiple Regression Host Overload Detection and HLRHOD or Hybrid Local Regression Host Overload Detection algorithms) and 3 VM selection algorithms (MMT, RC and MC)	Power consumption was reduced by six times, and the ESV metric was improved by 24%

Table 2.4: continued

Researchers	Type of Platform (Dedicated V/S Combined)	Application type (HPC or Non- HPC)	Parameter evaluated	VM scheduling Technique	Study outcome
A. Beloglazov and R. Buyya	Dedicated (IaaS)	Non-HPC	Resource usage, energy consumption and VM placement	Time-shared scheduling algorithm and novel techniques for the auto- adjustment of utilization thresholds	The results showed reduced energy consumption and adherence to the Service Level Agreement (SLA) which proved the efficiency of the algorithm
R. N. Calheiros, R. Ranjan, A. Beloglazov, B. De Rose and Rajkumar	Dedicated	Non-HPC	Scalability, overhead and performance of CloudSim's (simulator) components	Cloudlet scheduler (space-shared or time- shared)	The response times showed that the policies and components were correctly implemented. The turnaround time was reduced by more than 50%, and the makespan improved by 20%

Table 2.4: continued

Researchers	Type of Platform (Dedicated V/S Combined)	Application type (HPC or Non- HPC)	Parameter evaluated	VM scheduling Technique	Study outcome
M. R. Chowdhury, M. R. Mahmud and R. M. Rahman	Dedicated	Non-HPC	Energy and performance	Multiple redesigned VM placement algorithms and clustered VMs	Lower power consumption, SLA violation and performance degradation

2.16 Summary

This chapter delved into the literature found on HPC cloud workload. It starts with an overview of the main concepts related to HPC and cloud while highlighting the importance of providing a method to enable accurate workload prediction using dynamic proactive VM consolidation techniques. The literature reviewed in Chapter 2 revealed that many studies have attempted to examine the current cloud service architecture for operating HPC. However, very few approaches focused on providing a method to effectively schedule a mix of HPC and non-HPC applications in the cloud to optimize resource utilization. The present study tries to fill this gap by developing a new technique for running the HPC and non-HPC applications using a single pool of resources in the cloud datacentre. The Hybrid Local Regression Host Overload Detection algorithm (HLRHOD) based proactive dynamic VM consolidation technique merged with an innovative and intelligent VM scheduling technique to co-locate HPC and non- HPC applications to avoid cross-application interferences are used. The proposed technique can help maximise the applications' performance and reduce the tradeoff between better performance and optimum resource utilization.

CHAPTER 3

RESEARCH METHODOLOGY

3.1 Overview

The essential purpose of research is to generate new knowledge or get a better grasp of a subject (Guimaraes, 2003). A research methodology outlines the precise strategy and organized steps to pursue this new knowledge. Based on the research paradigm used, the standard research methodologies in the field of computer science can be divided into two main categories: Theoretical or Analytical approaches and Experimental techniques (Maringanti, 2022). Theoretical or Analytical approaches are based on formal theories, wherein systems under study are first translated into mathematical or statistical models. Then they are examined with the aid of already-developed mathematics and analytic techniques to learn more about the field (Niedbala, 2022). On the other hand, experimental approaches rely on examining current or proposed concepts and solutions through experiment design, observations, data gathering, and validation (Ameta, 2021). In computer science, experimental approaches are a handy tool for understanding phenomena, drawing conclusions, and proposing new algorithms.

This thesis addresses the RQ1, RQ2, and RQ3 using experimental approaches, as shown in Figure 3.1. Two experimental research methodologies are typically used in VM scheduling studies: empirical research and simulations. To understand an existing system's characteristics, empirical research is widely used.

Exploratory and explanatory approaches are used in empirical research. Explanatory research is a term used to describe studies that provide reasons for

phenomena. The researcher, in this context, attempts to pinpoint the causes and effects of the situation they are examining. Explanatory research methodologies include case studies, literature reviews, observational studies, focus groups, and pilot projects. Contrarily, explanatory research quantifies the relationship between various configurations and parameters through controlled experiments.

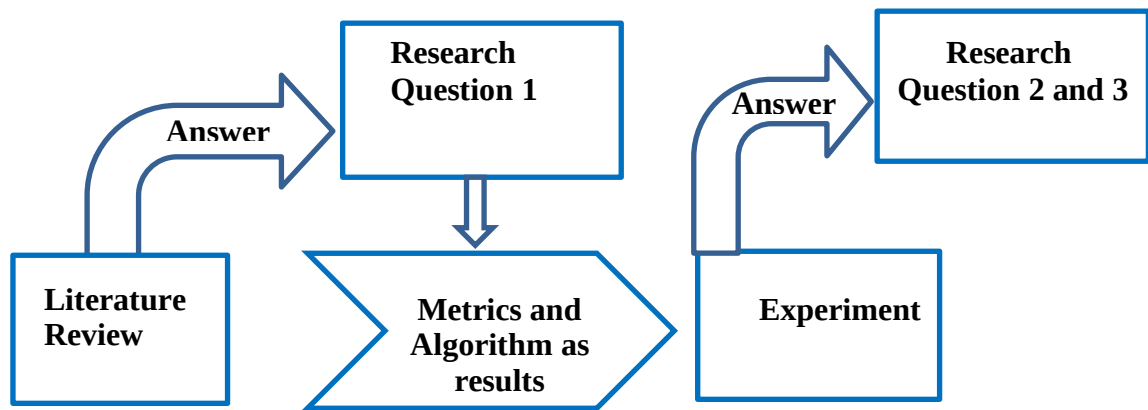


Figure 3.1: Research Methodology and Outcomes

In order to learn more about the details of the current models and conduct tests to find the best solution, we have conducted explanatory research using a Literature review. Based on the literature review results, the researcher identified the HPC and cloud limits to develop the research questions included in the proposed work. Additionally, simulations—one of the primary experimental research approaches—are employed to answer the research questions.

Implementing HPC on the cloud is accompanied by problems such as energy waste, performance, and poor utilization of resources. Figure 3.2 presents the action of each methodology phase. The literature review and analysis focus on critically examining

the available research to identify the gap targeted by this study. The design phase includes determining the proposed model's structure and components. In contrast, the Hybrid Local Regression Host Overload Detection algorithm (HLRHOD) is used as proactive dynamic VM consolidation and is presented as the base for the suggested scheduler and the application's nature-aware VM placement technique. The evaluation phase includes applying the suggested model in CloudSim to ensure it meets its aims.

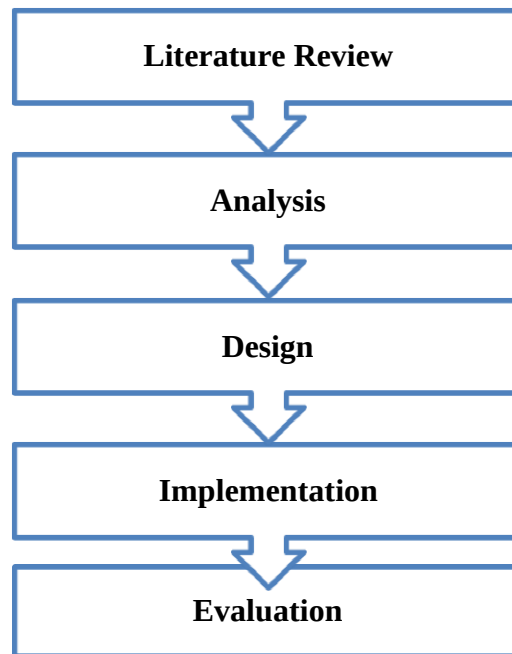


Figure 3.2: Empirical Research Methodology

3.2 The Literature Review

An in-depth systematic literature review is conducted while focusing on running HPC in the cloud and the accompanying problems. The target was to identify, evaluate and interpret the literature about the research questions and the subject matter of the study. The purpose of the literature review was to summarize the HPC cloud's complex scenario, investigate gaps, and find a solution to the problems. The literature review is a critical

element of research to lay ground knowledge and achieve a deep understanding of the subject under study. The literature reviewed in Chapter 2 revealed that many studies have attempted to examine the current cloud service architecture for operating HPC. However, very few approaches focused on providing a method to effectively schedule a mix of HPC and non-HPC applications in the cloud to optimize resource utilization. The proposed study tries to fill this gap by developing a new dynamic scheduler for running the HPC and non-HPC applications using a single pool of resources in the cloud data centre. After reviewing several recent research studies on the subject, it was determined that the phases of the design, implementation and evaluation of the proposed work are based on the most relevant studies on HPC in the cloud, precisely (Gupta et al., 2013) and (Abdelsamea et al., 2017).

3.3 Analysis

The baseline research paper chosen for study based on a literature review was crucial in creating the technique for the proposed work.

In Gupta (2014), they investigated a shared pool of resources that can be used to implement very diverse classes of applications in the cloud efficiently. They claim they provide a solution for the challenges of collocated allocation of various VM instances of a single-job request to physical hosts. They stated that they achieved this goal while maintaining Service Level Agreement (SLA) requirements. They developed scheduling algorithms to customize resource allocation for only HPC applications by applying Multi-dimensional Online Bin Packing (MDOBP) heuristics. They also made sure that cross-application interference was limited as necessary. However, besides all the achievements mentioned above in Gupta (2014), they ignored the energy problem and SLA violations. In

addition, they used the static VM consolidation techniques and limited the discussion about the optimum HPC performance and optimum utilization of resources by focusing only on throughput; therefore, other dimensions such as energy utilization of resources (i.e. RAM, CPU and bandwidth) are yet to be examined.

Furthermore, to validate the study, the researcher started to examine the energy efficiency of the proposed algorithm MDOBP (Gupta, 2014) and the default scheduler of the CloudSim. Figure 3.3 below shows a comparison between MDOBP (Gupta, 2014) and the default scheduler of the CloudSim regarding energy consumption. MDOBP consumes more energy than the default due to the likely limits of the technique of the static VM consolidation applied in the research. Therefore, this study proposes a proactive dynamic VM consolidation technique for HPC.

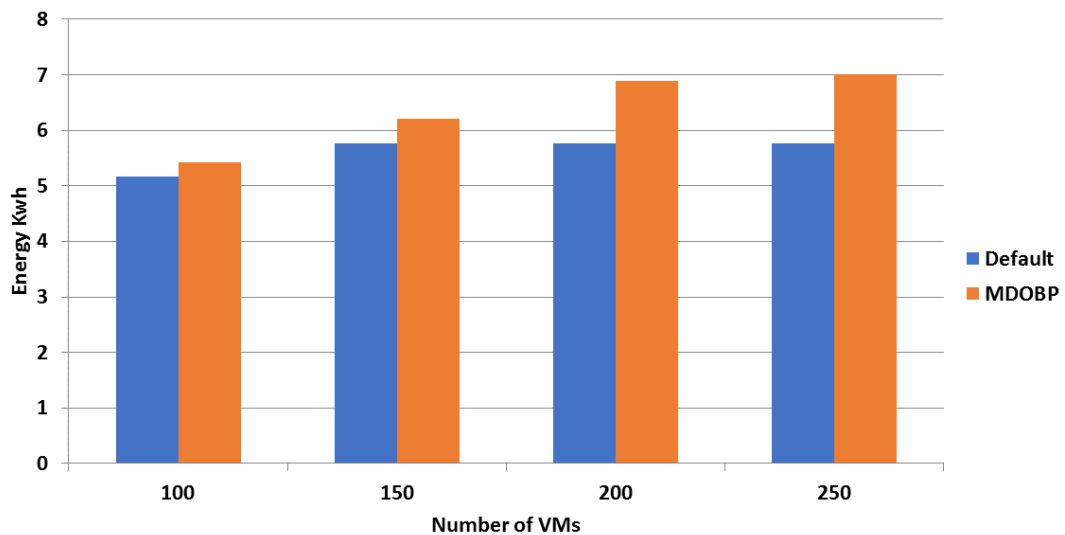


Figure 3.3: Energy Comparison of MDOBP vs Default Scheduler

In another research by Abdelsamea et al. (2017), they tried to achieve an optimum VM consolidation, transferring VMs between physical servers with no downtime. Thus, resource and energy consumption are rationalized. VM consolidation includes overload/underload detection, VM selection and VM placement. Their approach does not apply only heuristic-based techniques or periodic adaptation of the VM allocation as most current VM consolidation approaches do. Abdelsamea et al. (2017) do not rely on CPU utilization for host overload detection only as in most current studies; rather, they developed hybrid factors to improve VM consolidation. They introduce a multiple regression algorithm based on CPU, memory, and bandwidth utilization for detecting host overload. Their algorithm, the Hybrid Local Regression Host Overload Detection algorithm (HLRHOD), is significant because they claim that it achieves energy consumption reduction while maintaining the SLA requirements and provides an actual indication of host utilization according to three parameters utilization (CPU, Memory, Bandwidth) and not the CPU parameter only. They used CloudSim to prove that the proposed algorithm achieves a six times reduction in energy consumption compared to single-factor regression algorithms that use random workload. The proposed work takes a similar approach by developing a model based on the Hybrid Local Regression Host Overload Detection algorithm (HLRHOD) proposed by Abdelsamea et al. (2017).

3.4 Design of the Proposed Energy-Aware Multi-Dimensional Online Bin Packing (EAMDOBP) Technique

The design section of the methodology explains the development of the proposed Energy-Aware Multi-Dimensional Online Bin Packing (EAMDOBP) algorithms based on proactive dynamic VM consolidation in cloud computing.

3.4.1 Energy-efficient and HPC-Aware Proactive Dynamic VM Consolidation in Cloud Computing

Static VM consolidation and dynamic VM consolidation are two different techniques to achieve VM consolidation. Figure 3.4 shows how static consolidation works where the VM placement on PMs is pre-determined. There is a significant similarity between the N-dimensional bin packing problem and static VM placement as the bins represent PMs, and the packing items are similar to the VMs. Also, the volume of the resource natures determines the bin's size. Static VM consolidation is suitable for jobs that are short and run for only a couple of hours where PMs resources are predetermined for different types of VMs (Shim, 2016). Energy consumption is defined based on heuristics or historical VMs demand patterns.

Nonetheless, application providers' cost usually increases when resource demand is low. Also, there may be a lack of resources in periods when resource utilisation is high. (Zheng et al., 2016)

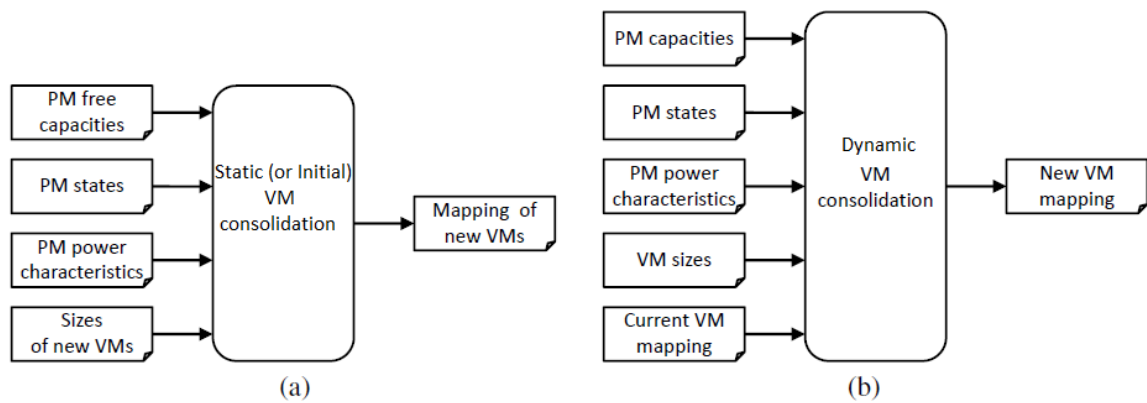


Figure 3.4: Inputs and Outputs of (a) Static VM Consolidation and (b) Dynamic VM Consolidation

Figure 3.4 shows the dynamic VM consolidation. It starts with a current mapping of VM-to-PM and tries to get the best possible mapping. In Dynamic VM consolidation, compatible PMs are sought to allocate the chosen VMs from cold and hot-cold PMs. The effectiveness of the whole VM consolidation process is incredibly reliant on PM for VM migration. The dynamic reallocation of the VMs is a systematic process for determining the best allocation to achieve the cloud provider's objectives. To balance resource utilization and PM overload, VM allocation should be based on an application-centric and energy-aware strategy. The ultimate purpose is to limit resource and energy waste while satisfying application QoS. Through dynamic VM consolidation, energy consumption will be minimized by reducing the number of operating servers. Dynamic VM consolidation will minimize PMs utilized through re-allocation or live migration of VMs between PMs while interference is minimized. Accordingly, dynamic VM consolidation will maintain the performance and the predefined QoS in the SLA agreed by the tenant and the service provider. Therefore, dynamic consolidation is the most efficient method to improve resource utilization. Dynamic VM consolidation exists in two types: reactive and proactive. When the initial placement is adjusted when the system becomes in a specific unneeded state, it is called reactive provisioning, where the adjustment happens because of issues related to performance, maintenance, power, load, or SLA violations. However, in proactive monitoring, the VM's initial placement changes before the system become in a particular state based on monitoring, historical data, and prediction algorithms (Masdari et al., 2016). In proactive VM consolidation, changes in the workload and system usage are prepared in advance through prediction.

One of the most severe issues in cloud computing is the real-time prediction of resource usage. Some studies focused on server load prediction (Song et al., 2018). Others

focused on VM load prediction (Jheng et al., 2014). Also, VM utilization and host utilization prediction (Mason et al., 2018) have been the topic of some studies. The proposed work attempts to predict the level of CPU, memory and bandwidth utilization of a host.

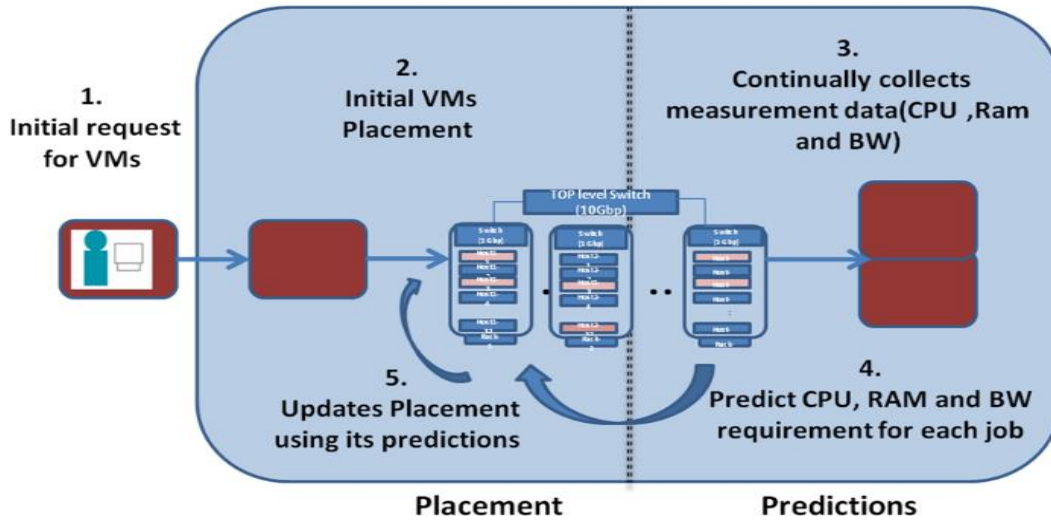


Figure 3.5: The Proposed Energy-Aware Multi-Dimensional Online Bin Packing (EAMDOBP) Architecture

3.4.2 Energy-Aware Multi-Dimensional Online Bin Packing (EAMDOBP) Architecture

Energy-Aware Multi-Dimensional Online Bin Packing (EAMDOBP) algorithm's high-level architecture is shown in Figure 3.5. It is divided into two sections: prediction and placement.

i. Prediction

The Hybrid Local Regression Host Overload detection method (HLRHOD) predicts resource consumption. In an environment where dynamic and changeable workloads take place, fixed values of resource utilization are improper. Local regression (Cleveland & Loader, 1996) is beneficial to model a correlation between the predictor variable and the response variable as follows:

$$Y_i = f(x_i) + \epsilon_i \quad \text{Equation 3.1}$$

Where

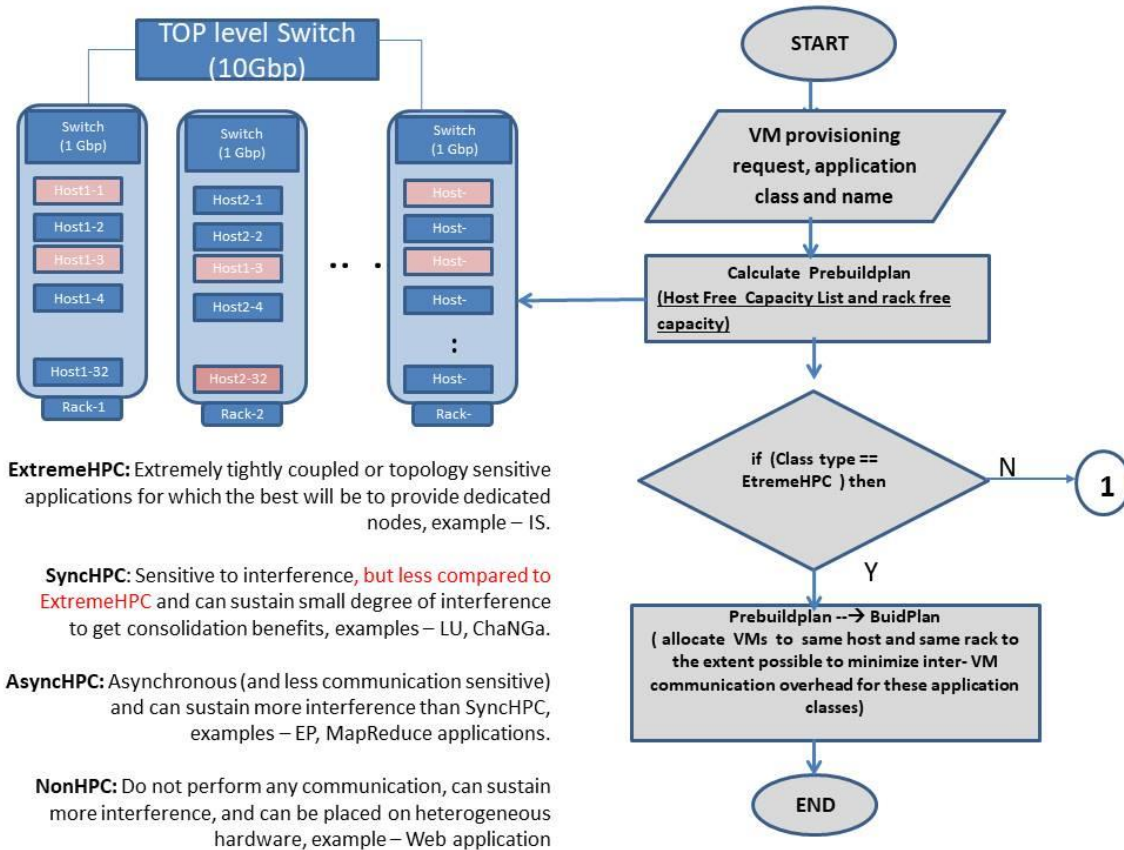
$f(x)$ is an unknown function, ϵ_i is an error term, representing random errors in the observations or variability from sources not included in the x_i . Local regression depends on placing simple models into localized data sub-sets to create a curve similar to the original details. Local regression is based on Cleveland's Loess method (Cleveland, 1979). The formula developed by Chen et al. (2012) will be supportive for calculating the host utilization using hybrid factors using a CPU-network-memory metric combining a load of virtual and physical servers. The host utilization will be computed, and the local regression host overload algorithm will be used (Beloglazov & Buyya, 2012).

ii. Placement

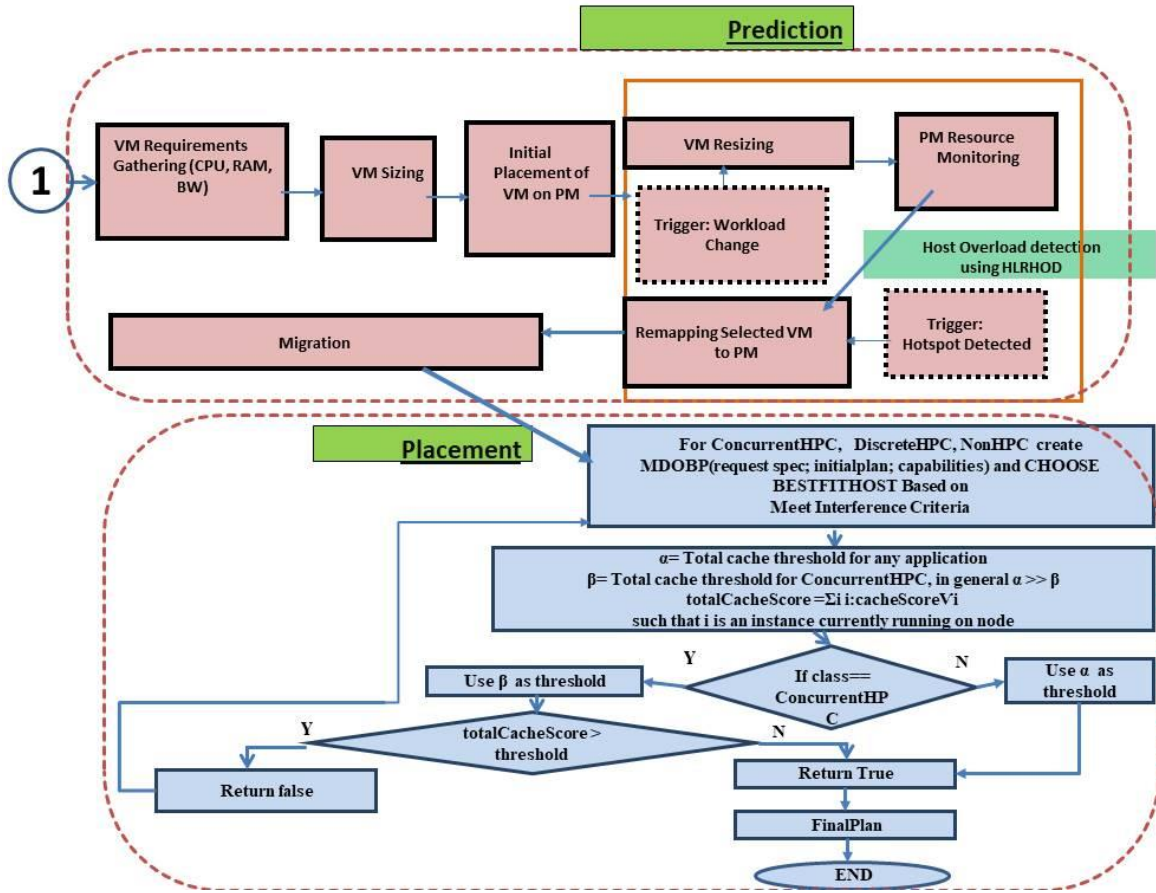
In this study, MDOBP (Multi-dimensional online bin packing) (Gupta et al., 2013) is used for VM placement to minimize the interference between co-located HPC and non-HPC applications. Each HPC application is first categorized according to its communication pattern. In this way, the virtual machine is elected by this algorithm to host an application. For example, tightly-coupled synchronous HPC applications should

continuously operate on a dedicated physical machine. For other HPC applications, the VM placement strategy focuses on reducing the number of physical machines while maintaining an acceptable level of interference. Therefore, MDOBP minimises interference and the number of operating physical machines possible.

Figure 3.6 (a) and (b) shows proposed proactive VM consolidation flowcharts. As a start, lists of the host's free and rack-free capacity are made, as shown in Figure 3.6 (a). Then, the data of the request specification is taken as input. Cloudlets are created based on the application type, capacity (i.e. Number of CPU, RAM and bandwidth) and instance type VM. Then, the cloudlet is dedicated to VMs, which are allocated to a host. The status is checked in each scheduled interval using the proposed proactive dynamic consolidation technique. With the default underload detection algorithm in scheduled intervals, the underutilized hosts are detected and deactivated after migrating all VM to active VMs, as shown in Figure 3.6 (b). HLRHOD identifies the overloaded hosts, and the surplus VM is moved to available hosts. Sometimes, turning on an additional host is necessary. A log is created after each iteration to evaluate energy consumption and QoS.



(a)



(b)

Figure 3.6: Energy-Aware Multi-Dimensional Online Bin Packing (EAMDOBP)

3.5 Implementation

Cloud computing has faced backlashes since entering the IT field due to its high operation cost and dependence on constant Internet accessibility. Furthermore, repeatedly testing the new algorithm on an extensive infrastructure can make the task challenging, making the project's development economically impractical (Beloglazov & Buyya, 2012). Simulation tools provide evidence of the suggested algorithms to prevent the project from overhead costs and unnecessary complications. Simulation tools are commonly used

throughout the development industry as they are feasible in all aspects. They help lower the overall cost by depicting the real-world scenario and allowing an unlimited number of repeated testing (Pandey & Gonnade, 2014). Also, simulation tools are user-friendly, and all computer-literate users can use them without difficulty, regardless of their prior programming skills (Chakravorty et al., 2006).

The CloudSim Toolkit (Rodrigo & Rajiv, 2011) is a cloud systems simulator that supports the modelling of the system and behaviour of cloud system resources like VMs, data centres, and resource management policies. VMs can be provisioned at the host and VM levels using the CloudSim Toolkit by using space-sharing and time-sharing techniques. The space-sharing approach CloudSim is similar to the First Come, First Serve (FCFS) scheduling algorithm where specific CPU cores are distributed to particular VMs. On the other hand, the Round-Robin (RR) scheduling technique and time-sharing approach are alike as both allocate the capacity of one core to variable VMs. CloudSim makes virtualization framework models applicable based on the on-demand provision of resources and resource management.

Moreover, it can simulate service applications with varying workloads and quickly extend commonly used application delivery techniques with the least possible effort. The algorithm suggested in this thesis has been experimented with the CloudSim 3.0 simulation toolkit (Calheiros, 2011) using four VM types and two PM types. CloudSim allows energy awareness simulation; however, energy-aware simulation was added to the framework externally as it was not initially present (Calheiros, 2011).

3.6 Evaluation Methodology and Experimental Setup

The implementation of the proposed EAMDOBP technique is explained by detailing the experimental setup, power model, simulation software used, performance metrics, and workload.

Like every piece of software, CloudSim 3.0 also has its limitations that must be addressed. We identified these while implementing our scheduling and migration techniques on CloudSim 3.0. Firstly, designs of CloudSim 3.0 have been developed for tasks that execute on the cloud and only require one processor, while HPC machines require multiple processors to function correctly. To overcome this hurdle, CloudSim 3.0 was extended to make the environment suitable for simulating HPC in the cloud. This was achieved by enhancing the implementation of multi-core jobs in HPC; this consequently extended the PowerDatacenter class responsible for authorizing the simulation of power-aware data centres in the cloud environment. Additionally, DatacenterBroker class was extended to DatacenterBrokerEAMDOBP, with two additional features:

- i. Allocation of lengthy cloudlet to efficient VM: The Virtual Machine and Cloudlet attributes are determined, and the most efficient VMs are assigned lengthy jobs. This is done to prevent the VM from entering the idle state and to reduce the execution time. The utilisation of VMs will be improved, increasing efficiency since the VMs will not stay idle, and the default drawback of CloudSim 3.0, called the DatacenterBroker policy, will also be eliminated.

ii. Cache-awareness for addressing cross-interference

The execution of the high-performance application in the cloud gives rise to cross-interference. Accurately forecasting the slowdown in each application can provide many benefits like it can prevent shared resource utilization from unexplainable application slowdowns. The extended DatacenterBroker policy will ensure that the VMs do not exceed the interference criteria before assigning them cloudlets. Just like Gupta's approach, fixed criteria were used to calculate the interference. First, the total value of the cache requested does not surpass the threshold for any VM operating on the host. Careful and thorough experimentation was done to decide the threshold value. It was observed in Gupta's approach that a value larger than 60 caused a decrease in efficiency because a single node was packaged with multiple aggressive cache-intense applications (Gupta, 2014).

On the contrary, a value smaller than 60 resulted in a waste of CPU cores since only a few applications had significant cache scores. So, the value of 60 was set as the threshold. Secondly, interference indices were maintained to record interference between applications that had received high-performance penance when the hosts were shared. The recorded interference indices were used to dissuade co-locations that employ more resources so that the suitable cloudlet (job) was assigned to the VM by the DatacenterBrokerEAMDOBP.

When the simulation begins, several VMs of several types will be generated, and cloudlets (jobs) will be directed to the DatacenterBrokerEAMDOBP broker for mapping jobs to VMs. The class of cloudlets has been modified, so now it contains the application class and name for VM provisioning along with the parameters it already had previously. The default CloudSim class called VmAllocationPolicySimple has been extended to

PowerVmAllocationPolicyMigrationEAMDOBP for managing the user requests belonging to various types of VM. Two methods, findHostForVm (for initial VM allocation) and optimizeAllocation (for VM consolidation) in the PowerVmAllocationPolicyMigrationEAMDOBP class, are responsible for conducting the proposed Energy-aware multi-dimensional online bin packing scheduling. Additionally, the class has approximately 30+ methods like (PowerVm and PowerHost), primarily called from the Optimize Allocation, which were modified to implement the proposed algorithm. Table 3.1 lists the names of the default classes and their extended versions.

Table 3.1: The Original and Extended Classes of CloudSim

CloudSim	Extended
org.cloudbus.cloudsim.power.PowerData centre,	org.cloudbus.cloudsim.power.HPCPower Data centre,
org.cloudbus.cloudsim.power.PowerData center Broker.	org.cloudbus.cloudsim.power.HPCPower Data center Broker
org.cloudbus.cloudsim.power.PowerVmAllocationPolicyMigrationAbstract.	org.cloudbus.cloudsim.power.PowerVmAllocationPolicyMigrationAbstractHPC

Once the HLRHOD detects host overload (Abdelsamea et al., 2017), some specific VMs will be selected and moved from the overloaded host to another host. Also, the Minimum Migration Time (MMT) algorithm is used (Beloglazov & Buyya, 2012) for VM selection, and Energy-Aware Multi-Dimensional Online Bin Packing (EAMDOBP) algorithm is used for VM migration.

To prove the efficiency and competency of the Energy-Aware Multi-Dimensional Online Bin Packing (EAMDOBP) algorithm, the proposed algorithm is substantiated with the following heuristic algorithms from the literature:

- The Power-Aware-Best-Fit Decreasing algorithm (PABFD) (Beloglazov & Buyya, 2012).
- The Modified- Worst-Fit Decreasing algorithm (MWFD) algorithm (Chowdhury et al., 2015).

3.6.1 Power Model

Computational resource energy consumption in the cloud data centre is determined by power supplies, cooling devices, disk storage, memory, and CPU (Minas & Ellison, 2009). However, recent studies have indicated that energy consumption increases linearly with the increase of CPU. Moreover, providing hosts with a great memory has become possible and is a key factor in the energy consumption of a host. The complexity of modelling the energy consumption of modern multicore CPUs makes establishing analytical models a difficult research problem. Accordingly, SPECpower benchmark results are used instead of an analytic formula of node energy consumption. A data centre including 50 hosts and 50 VMs is simulated, and planet lab and random workload traces are utilized. A scheduling interval set to 300 sec frequently evaluates the host overload (Nithya & John, 2013). The host types are: HP ProLiant ML110 G4 (Intel Xeon 3040, 2 cores 1860 MHz, 4 GB), and HP ProLiant ML110G5 (Intel Xeon 3075, 2 cores 2660 MHz, 4 GB). The power consumption features of the chosen hosts are presented in Table 3.2.

The purpose is to simulate many hosts to examine the influence of virtual machine consolidation. That is why using servers with more cores is not preferred. Also, simulating less powerful CPUs is more suitable because only a few workloads can overload the host in this case. Dual-core CPUs can adequately evaluate the resource management algorithms of the multicore CPUs.

Table 3.2: Power Consumed by the Chosen Hosts at Various Load Levels in Watts

Server	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
HP Proliant G4	86	89.4	92.6	96	99.5	102	106	108	112	114	117
HP Proliant G5	93.7	97	101	105	110	116	121	125	129	133	135

3.6.2 Performance Metrics

Specific metrics are utilized to examine the performance of the algorithms as follows:

- **Simulation Time (ST):** Also known as makespan, it is the duration of the experiment in seconds. The duration of the simulation is calculated while taking into consideration the number of applications processed. The purpose of Simulation time (MakeSpan) is to calculate algorithm efficiency.
- **Throughput:** It is the work a system accomplishes in a specific time, so increasing throughput is the ultimate goal of IT managers and scientists. In this study, throughput indicates the performance of a system.

- **Efficient resource utilization (ERU):** the optimum consumption of CPU, memory and bandwidth by HPC and non- HPC applications in the data centres
- **Power usage effectiveness (PUE)** indicates the data centre's performance and improves it. PUE is regarded as the most common data centre metric. The PUE is calculated using Equation 3.2, where PCS is the cooling power, and PC is the computing power of the host. This calculation is iterated over each host.

$$PUE = \frac{P_{cs}+P_c}{P_c}; \quad \text{Equation 3.2}$$

Table 3.3 shows the PUE values based on several experiments conducted in a small data centre (Uddin et al., 2013). Low PUE indicates higher efficiency as a significant amount of power has been consumed by computing power (Yuventi & Mehdizadeh, 2013).

Table 3.3: PUE Efficiency Values

PUE	Level of efficiency
3.0	Very inefficient
2.5	Inefficient
2.0	Average
1.5	Efficient
1.2	Very efficient
1.1	Standard

- **Number of VM migrations:** The VMs are selected for migration when the overloaded or under-loaded hosts are determined for achieving dynamic VM consolidation. Reducing the VM migration time is a crucial constraint in the migration process because it necessitates reducing the total number of VM migrations.

- **Energy and SLA violations (ESV):** There is an inverse relationship between the energy consumption of physical hosts and SLAV because energy consumption can be reduced by increasing SLAV, as shown in Equation 3.3. However, the primary purpose of host managing frameworks should be the reduction of energy consumption and SLAV. Therefore, a parameter combining energy and SLA violations is proposed by Beloglazov and Buyya (2012). Low ESV is preferred because it indicates that SLA violations are less than the energy saved.

$$ESV = E \times SLAV \quad \text{Equation 3.3}$$

3.6.3 Workload

Accounting records of workload for several months are obtained from the national grid of the Czech Republic called MetaCentrum (Parallel workloads archive: MetaCentrum, 2022). This grid consists of 14 clusters and nodes, each including multiprocessor machines (806 processors). The workload data includes CPU and memory usage. The standard workload format (SWF) contains the log. METACENTRUM-2013-1.swf. It is used while relying on accounting data collected by the scheduler. Three utilization models were built to check the CPU and RAM, calculate the Bandwidth (BW) from the workload, and send it to the cloudlets.

3.6.4 CloudSim Architecture and Modelling

This thesis uses CloudSim for cloud simulation, interpretation of resource algorithms, and modelling due to its efficiency, applicability, and flexibility. It will help fulfil the proposed work's purpose by stimulating the activities of large-scale cloud data centres. Figure 3.7 below shows the CloudSim architecture.

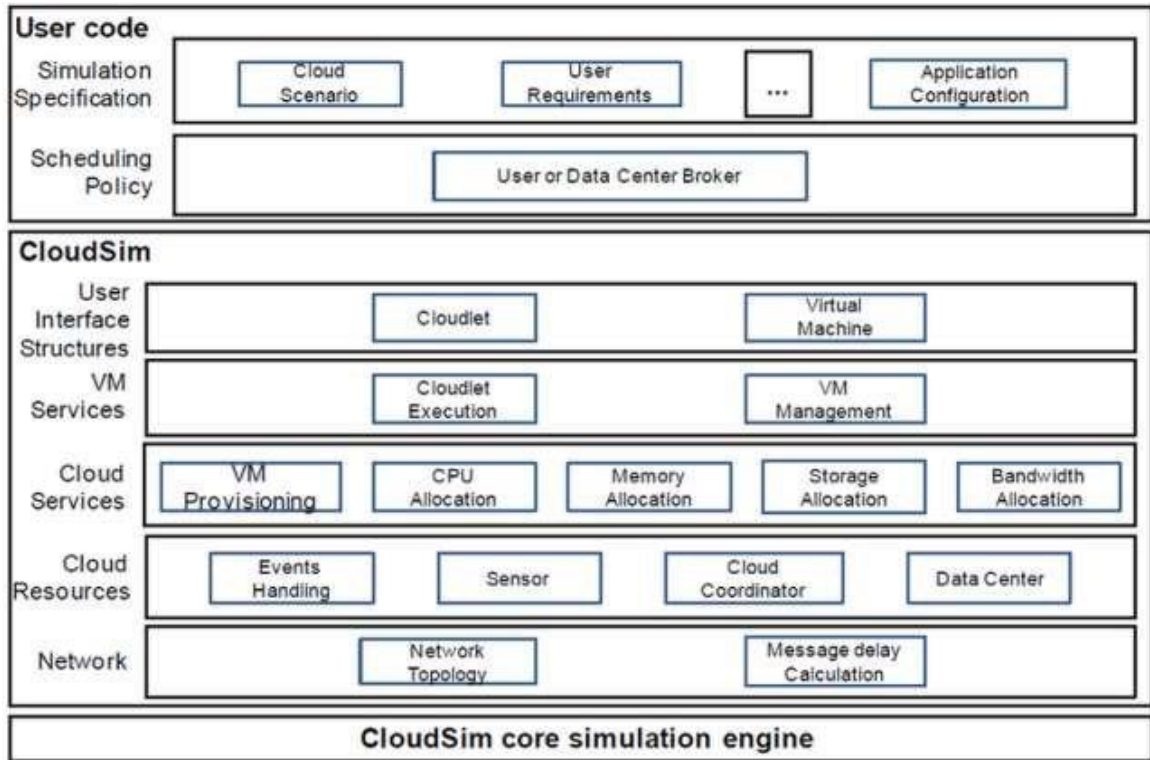


Figure 3.7: CloudSim Architecture (Redkar & Guidici, 2011)

CloudSim core simulation engine: this is the first version issued of CloudSim. It is helpful for some prosperities, including the host, data centre, data centre broker, VMs, etc.

CloudSim simulation: this layer is helpful for resource creation such as the host, VMs, and data centre. It helps manage application execution, interface management, storage, bandwidth, and memory.

User code is topping the CloudSim architecture and is essential for specific applications' requirements such as user type, data centre broker policies, and tasks.

The middle layer, CloudSim simulation, is used in the proposed work. CloudSim can simulate IaaS clouds after extending data centre, hosts, and VMs entities. Both multi-core and single functionalities are allowed for the hosts by CloudSim. The assignment of

all these entities will be completed according to the necessary characteristics. CloudSim supports Cloud Data centres (DC) simulation, providing various hardware configurations, such as storage, memory, bandwidth, etc. It facilitates the creation of Cloud entities, such as a server, broker, Cloud Info Service (CIS) and VM, as illustrated in Figure 3.8. It offers versatility in transfers to various virtualized services between space-share and time-share allocation of CPU cores. It also supports various VM policies CSP applied, allowing the data centre to allocate hosts to several VMs (Calheiros, 2011).

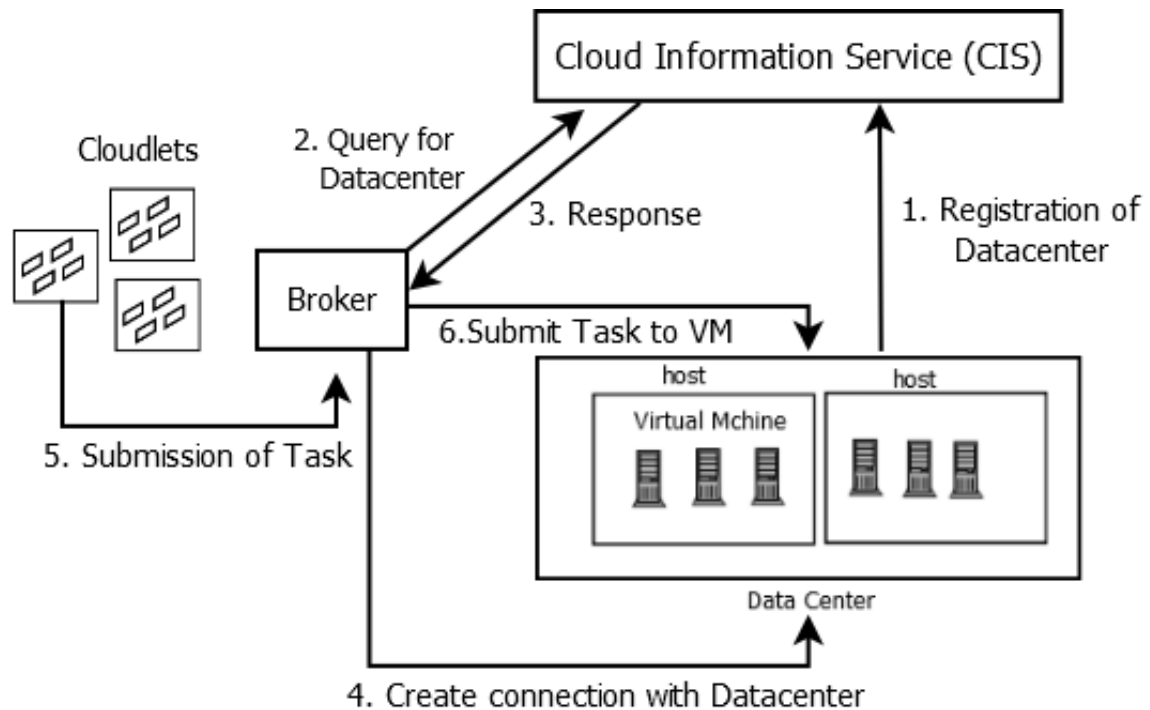


Figure 3.8: Communication Flow between Various Entities in CloudSim

3.7 Summary

This chapter explains the methodology applied in this study. The main aim is to develop a model for the various collocating combination of applications (i.e., HPC and non-HPC) workload in the cloud data centre to save energy and maintain the quality performance of the applications while simultaneously maximising the efficient usage of

resources. The components that will be used for the design, implementation and evaluation of the proposed model are explained in this chapter. Combining the Hybrid Local Regression Host Overload Detection algorithm (HLRHOD) and proactive dynamic VM consolidation is the base for the suggested technique. The evaluation phase includes applying the suggested technique in CloudSim to ensure it meets its aims.

CHAPTER 4

EVALUATION AND RESULTS

4.1 Overview

In this chapter, the researcher discussed the evaluation and results of the proposed technique that provisions resources (hosts) and VMs to a data centre to optimize energy consumption under application performance constraints, thus resulting in better and more efficient utilization of cloud data centres. The nature of the application needs to be adequately examined as naive co-scheduling of various nature (i.e. HPC and non-HPC) applications on shared resources will undoubtedly decrease the efficiency due to cross-interference. Cross-interference problem is often encountered while executing HPC applications in the cloud data centre. As shown in Chapter 2, the literature review, some researcher considers cross-interference challenge while proposing virtual machine placement strategies. However, they neither studied the effects of energy nor considered the proactive dynamic VM consolidation technique to dynamically reduce the number of used physical machines to optimize the usage of resources. This chapter includes simulation results and evaluation, as well as a comparison of techniques used in earlier studies.

4.2 Simulation Results and Analysis

This section explains the simulation outcomes of the proposed EAMDOBP technique compared with the other heuristic scheduling algorithms.

4.2.1 Number of VMs

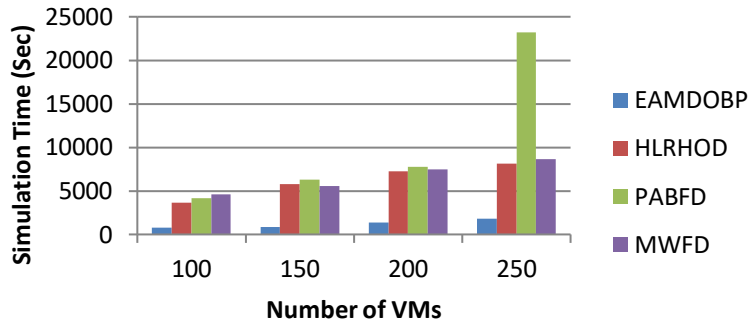
The sensitivity analysis conducted in this study relies on fixing the number of hosts to 800 and changing the number of virtual machines to 100, 150, 200 and 250 while adjusting the scheduling interval to 300 sec. the impact of changing the number of VMs is examined using Metacentrum workload traces on the energy consumption and other system and performance metrics for different algorithms. CPU, RAM, and bandwidth utilization were compared using the Metacentrum workload traces.

4.2.2 Algorithms Comparative Analysis

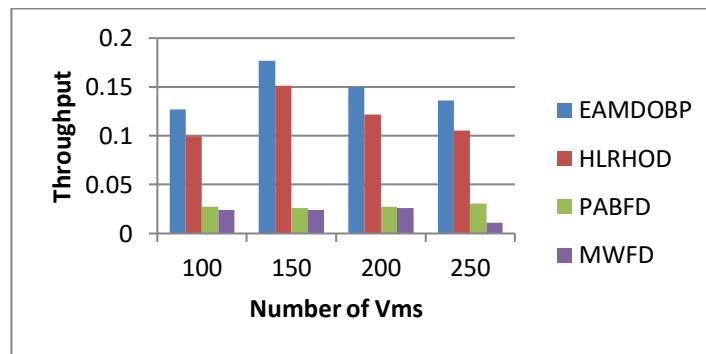
Simulation time is considered the most critical metric for evaluating performance in a dynamic system. The x-axis represents the number of VM, while the y-axis represents the Simulation time. Figure 4.1 (a) shows how the Energy-Aware Multi-Dimensional Online Bin Packing (EAMDOBP) algorithm outperforms the other algorithms in terms of Simulation time by reflecting the relative increase of 60% compared to HLRHOD, PABFD and MWFD, the novel Energy-Aware Multi-Dimensional Online Bin Packing (EAMDOBP) technique delivers the fastest simulation time independent of the number of VM. Furthermore, when the number of VM is large, the simulation time using PABFD is the highest. In contrast, the simulation duration is consistently less when HLRHOD is used instead of MWFD and PABFD.

The traditional HPC framework co-schedules tasks of the exact nature to achieve synchronous development. Following the proposed EAMDOBP technique, VMs are assigned to the host while keeping the nature of the HPC workload. The EAMDOBP algorithm produces a relative increase in throughput rate by 73%, as illustrated in Figure

4.1(b). On the other hand, the HLRHOD, PABFD, and MWFD algorithms produce the same throughput rates.



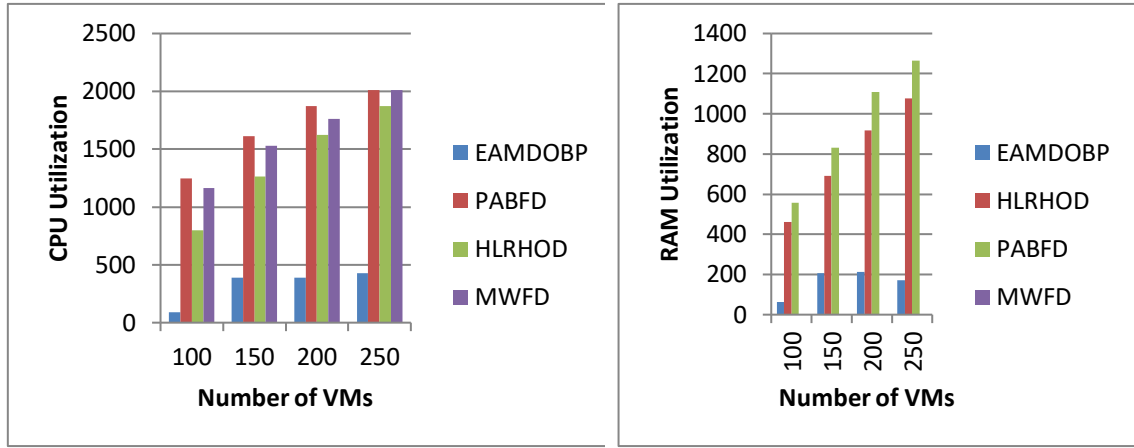
(a) Simulation Time



(b) Throughput

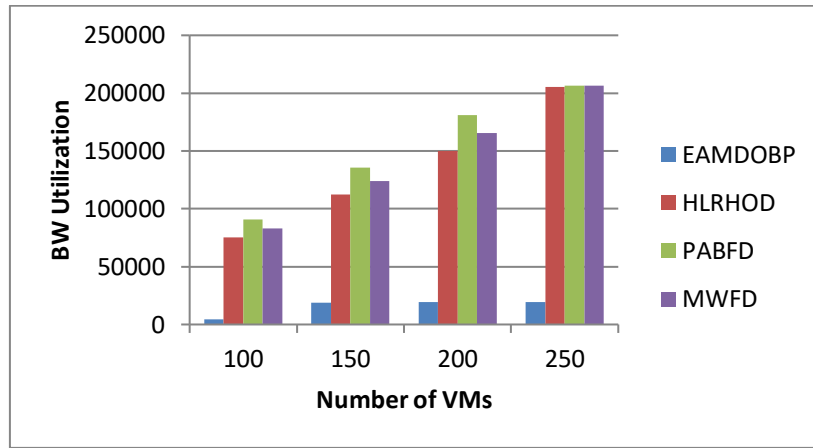
Figure 4.1: Algorithms Comparison for Metacentrum HPC Workload, where the Simulation time and Throughput metrics are depicted in Log-Scale

The EAMDOBP algorithm is used to predict resource use and assign the VM as per the nature of the workload to avoid cross-interference. Hence, it results in better CPU, RAM and Bandwidth utilization by a relative improvement of 77%, 84% and 70%, respectively, as shown in Figure 4.2. HLRHOD improves HPC application execution efficiency while increasing CPU, RAM and Bandwidth utilization.



(a) CPU utilization

(b) RAM Utilization



(c) Bandwidth Utilization

Figure 4.2: Algorithms Comparison for Metacentrum HPC Workload, where the CPU, RAM and Bandwidth Utilization metrics are depicted in Log-Scale

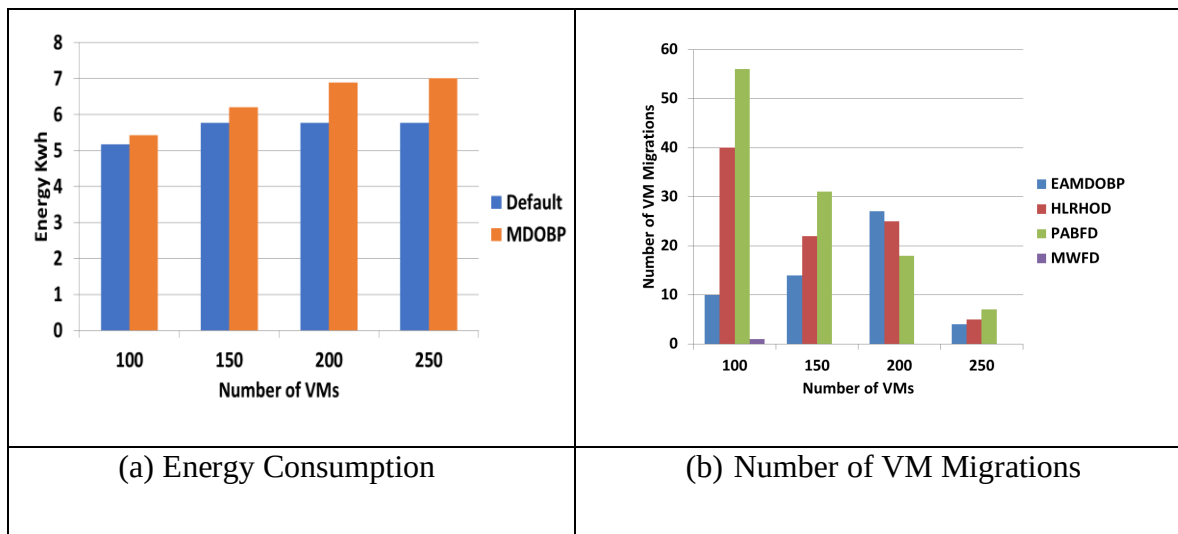
The EAMDOBP algorithm can predict the best node for VM allocation while maintaining rational energy consumption. The co-location of the VMs is conducted while considering energy and interference rate in all the workload allocations in the system nodes. Therefore, VM migration is reduced, which minimizes energy waste. Figure 4.3(a) shows that when consolidation is absent, energy consumption increases with the number of

VM increases. Also, when consolidation occurs, energy consumption remains the same even with the increase in VMs. This may be interpreted as VM consolidation achieving a reduction in energy consumption.

As shown in Figure 4.3(b), VM migrations are at the lowest level with the EAMDOBP algorithm compared to PABFD, MWFD, and HLRHOD. With the MWFD algorithm, VM migration is at its maximum levels. Each host's cooling, processing power, and PUE were measured after each migration.

Figure 4.3(c) shows that PUE is the least by the EAMDOBP algorithm compared to PABFD, MWFD, and HLRHOD. Low PUE means more efficiency because, in this case, the computing power composes the most significant part of consumed power. In the case of the PABFD algorithm, the PUE is at the maximum.

Figure 4.3(d) shows that EAMDOBP results in the lowest ESV compared to PABFD, MWFD, and HLRHOD. With the increase in the quantity of VM, ESV levels increase significantly with PABFD and MWFD algorithms.



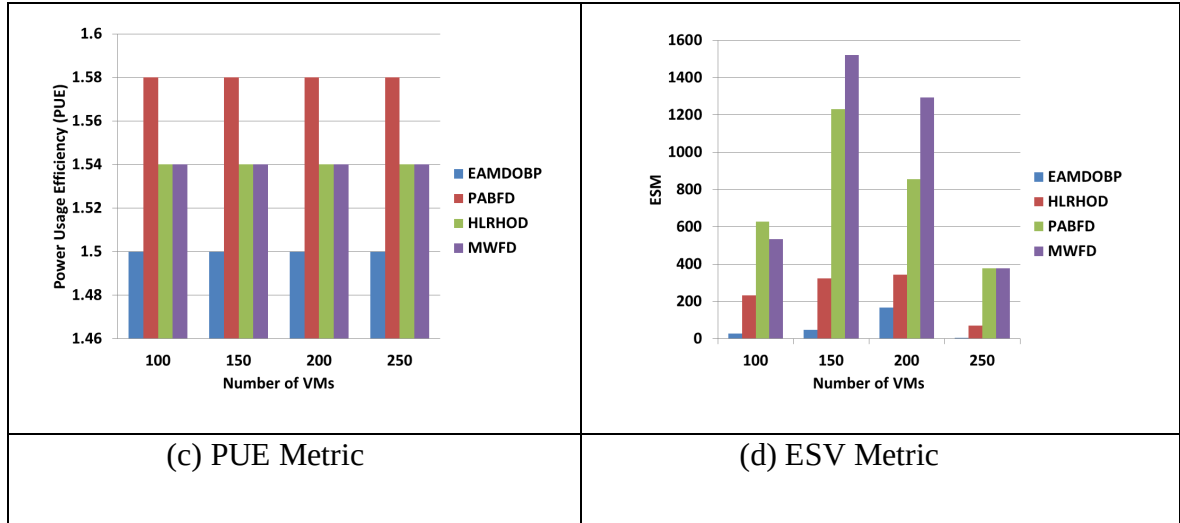


Figure 4.3: EAMDOBP Results

4.3 Contribution and Significance

This study presents a new approach to the optimum operation of HPC in the cloud. Cloud computing Infrastructure-as-a-service has gained tremendous popularity as a promising alternative to supercomputers. Small and medium companies with emerging HPC demands find the pay-as-you-go model of cloud very useful. However, HPC requirements and current cloud environments are not fully compatible. The main objective of this study is to provide a solution to achieve effective and efficient HPC in the cloud. In addition to addressing the challenges of running HPC applications in the cloud, the study highlights the great potential of running HPC in the cloud. The combination of HPC and cloud is examined in this thesis from the aspects of the HPC users and cloud providers (achieving tangible benefits and running a profitable business). Therefore, our approach is focused on enhancing HPC performance, resource utilization and cost in the cloud.

This study examined how VM consolidation can optimize HPC running in the cloud. VMs consolidation using smart co-location can achieve the desired trade-off

between the optimum performance of HPC in the cloud and resource utilization. The purpose of the algorithms proposed in the proposed work is to maintain the optimum performance of HPC in the cloud while rationalizing resources and cost. This will be translated into broader adoption of HPC in the cloud.

In the literature review, the related studies were examined to pinpoint the current limitations of HPC in the cloud and found that developing an improved proactive Dynamic VM consolidation would be a practical solution. Considering the information obtained from the literature review, this research designed (Energy Aware Multi-Dimensional Online Bin Packing) EAMDOBP.

CloudSim is used in this study as a simulator to test the proposed model (EAMDOBP). The experiments were conducted with the CloudSim 3.0 simulation toolkit using four VM types and two PM types. Furthermore, CloudSim was extended with energy-aware simulations, initially not present in the core framework.

The acceptance of the developed model was verified using Heuristics PABFD (power-aware best fit decreasing algorithm), MWFD (Modified Worst Fit decreasing algorithm) and Hybrid Local Regression Host Overload Detection (HLRHOD). The results were analysed, proving that CSPs can employ EAMDOBP to save energy while improving the overall efficiency of cloud data centres without sacrificing heterogeneous workload performance.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Introduction

In this thesis, efforts are made to leverage the potential of cloud computing to expand the reach of HPC. Recent developments have shown that cloud computing with Infrastructure-as-a-Service options is a promising addition to or replacement for conventional supercomputers. Since they can benefit from the pay-as-you-go model (renting vs buying) and elasticity-on-demand provisioning in clouds, small and medium-sized companies are particularly drawn to them. This is especially true for those with emerging or inconsistent HPC demands. The extensive performance research in the thesis demonstrates that, despite this promise, there is a mismatch between present cloud systems and typical HPC requirements. The proposed technique offers effective and efficient HPC strategies on the cloud to close that gap.

5.2 Conclusion

The researcher developed three main objectives for the study; in the next paragraph, the researcher provides the study's conclusion in alignment with the study objectives.

- i. To propose an intelligent, proactive dynamic VM scheduling technique to optimize resource utilization in a cloud data centre:

Virtual machines serve as the sole mode of resource provision in cloud-based datacenters (VM). VM allocation needs to be improved for effective resource use. A novel proactive dynamic resource allocation for virtual machine distribution is suggested in the proposed study. Impact of virtual machine allocations on co-located virtual machines are

thoroughly considered by combining the present status and future forecasted data of each application load as well as the nature of the workload. Through intelligent HPC-aware VM placement, particularly application awareness, we optimize the performance for HPC in the cloud. Experiments show that the Energy Aware Multi-Dimensional Online Bin Packing (EAMDOBP) has achieved better CPU, RAM and Bandwidth utilization by a relative improvement of 77%, 84% and 70%, respectively compared to PABFD, MWFD and HLRHOD.

- ii. To test proactive dynamic VM scheduling technique for improving the efficiency of the cloud data centre; and

We explore the benefits and drawbacks of VM consolidation for HPC in the cloud. Furthermore, we create scheduling algorithms that optimize resource allocation while being HPC-aware. Researchers accomplish this by employing Energy Aware Multi-Dimensional Online Bin Packing (EAMDOBP) heuristics while keeping cross-application interference within a range. The suggested application-aware VM scheduling algorithm in CloudSim is presented, along with its techniques, implementation, and assessment. We also enhance CloudSim to make it acceptable for HPC cloud simulation and present simulation results.

- iii. To evaluate the energy and efficiency of the proposed proactive dynamic VM scheduling technique.

Experiments show that the Energy Aware Multi-Dimensional Online Bin Packing (EAMDOBP) rationalize energy consumption, enhances CPU utilization and reduces PUE, which means better energy efficiency. Moreover, reduced simulation time, higher throughput, minimized VM migration, and reduced ESV are achieved by EAMDOBP

compared to PABFD, MWFD and HLRHOD. The analysis of the results indicated that the EAMDOBP algorithm generates better results in all the parameters used compared to PABFD, MWFD and HLRHOD.

5.3 Future Research Recommendations

In this thesis, the researcher emphasised bridging the HPC-cloud divide by improving application performance and energy and resource efficiency. Another major challenge that discourages HPC users from moving to the cloud is security. Cloud security challenges include lack of trust in the cloud provider, loss of control and multi-tenancy. Future research is needed to address these issues to enable secure HPCs in the cloud. Furthermore, Current tools for performance analysis indicate only how the application has performed. They offer no detail on the other tasks that the shared RAM, CPU and bandwidth could have consumed. There is a strong need for tools to highlight regions with or without interference and recommend the proper combination of jobs to be scheduled using the same resource pools. Moreover, there is no standard tool or unit for calculating performance-energy trade-offs when consolidating servers. There is a need for a metric to convey the performance-energy trade-offs when consolidating servers and a tool to depict the metric values to assist system administrators in the server selection process.

5.4 Closing Remarks

Datacenters are becoming a prominent role in the development of our society. Managing these data centres, especially the servers that make them up, has become crucial as more individuals utilize the services provided. The management of both performance and energy usage is necessary. The data centre industry now contributes more than 7% of all global CO2 emissions.

In this thesis, tradeoff options for performance and energy efficiency were investigated. The proposed technique will help to promote effective server administration in data centres. This thesis argues that intelligent platform-to-application mapping and cloud scheduling considering HPC, can close the gap between present clouds and high-performance computing. We are confident that our study will be very beneficial in maximizing the full HPC cloud computing potential.

REFERENCES

- Abdelsamea, A., El-Moursy, A. A., Hemayed, E. E., & Eldeeb, H. (2016). Multiple regression host overload detection. *Sensors to Cloud Architectures Workshop (SCAW2016) held in conjunction with HPCA-22*.
- Abdelsamea, A., El-Moursy, A. A., Hemayed, E. E., & Eldeeb, H. (2017). Virtual machine consolidation enhancement using hybrid regression algorithms. *Egyptians Informatics Journal*, 18, 161-170.
- Abdelsamea, A., Hemayed, E. E., Eldeeb, H., & Elazhary, H. (2014). Virtual machine consolidation challenges: A review. *International Journal of Innovation and Applied Studies*, 8, 1504–1516.
- Abdi, H. (2007). Multiple correlation coefficient. *Encyclopedia of Measurement and Statistics (Edited by Neil J. Salkind)*, 648-651.
- Aboulnaga, A., Salem, K., Soror, A. A., Minhas, U. F., Kokosielis, P., & Kamath, S. (2009). Deploying database appliances in the cloud. *Data Engineering, IEEE Computing Society*.
- Adamuthe, A. C., & Thampi, G. T. (2014). Analyzing trend of cloud computing and it's enablers using gartner strategic technology.
- Adas, A. (1997). Traffic models in broadband networks. *IEEE Communications Magazine*, 35(7), 82-89.
- Ahmad, R. W., Gani, A., Hamid, H. A., Shiraz, M., Yousafzai, A., & Xia, F. (2015). A survey on virtual machine migration and server consolidation frameworks for cloud data centers. *Journal of Network and Computer Applications*, 52, 11-25.

- Alkayal, E. (2018). Optimizing resource allocation using multi-objective particle swarm optimization in cloud computing systems. England: University of Southampton.
- Alshammari, D. (2018). Evaluation of cloud computing modelling tools: Simulators and predictive models. *Thesis*. Glasgow, Scotland: University of Glasgow.
- Alves, M. M., Teylo, L., Frota, Y., & Drummond, L. M. (2018). An interference-aware virtual machine placement strategy for high performance computing applications in clouds. *2018 Symposium on High Performance Computing Systems (WSCAD)*, (pp. 94–100).
- Amazon. (2017). *Amazon Web Services*. Retrieved 11 December, 2018, from <https://aws.amazon.com>
- Amazon EC2. <http://aws.amazon.com/ec2/> (2018).
- Amazon simple storage service*. (n.d.). (Amazon) Retrieved from <http://docs.amazonwebservices.com/AmazonS3/latest/dev/>
- Amazon Web Services. (2021). *User Guide*. Retrieved 2021, from AWS: <https://docs.aws.amazon.com/AmazonS3/latest/userguide/Welcome.html>
- AMD IO-virtualization technology IOMMU*. (n.d.). (Advanced Micro Devices Inc.) Retrieved 12 12, 2018, from <http://support.amd.com/>
- Ameta, D. &. (2021). Why does the experiment fail with students? An Experience. *Association for Computing Machinery (ACM)*, (pp. 1-5).
- Amiri, M., Feizi-Derakhshi, M. R., & Mohammad-Khanli, L. (2016). IDS fitted Q improvement using fuzzy approach for resource provisioning in cloud. *Journal of Intelligent & Fuzzy Systems*, 32(1), 229-240.

- Amooee, G., Minaei-Bidgoli, B., & Bagheri-Dehnavi, M. A. (2011). A comparison between data mining prediction algorithms for fault detection (Case study: Ahanpishegan Co.). *International Journal of Computer Science Issues*, 8(6), 425-431.
- Anandharajan, B. D., & Bhagyaveni, A. M. (2013). VM consolidation techniques in cloud data center. *Journal of Theoretical and Applied Information Technology*, 53, 267–273.
- Anderson, M., & Braak, C. T. (2003). Permutation tests for multi-factorial analysis of variance. *Journal of Statistical Computation and Simulation*, 85-113.
- Anton, J., & Petouhoff, N. (1996). *Customer relationship management*. Prentice Hall.
- Antonenko, V., Chupakhin, A., Petrov, I., & Smeliansky, R. (2019). Improving resource usage in HPC clouds. *In CEUR Workshop Proceedings*, 180-184.
- Anuradha, V. P., & Sumathi, D. (2014). A survey on resource allocation strategies in cloud computing. *Information Communication and Embedded Systems (ICICES) in International Conference on (pp. 1-7). IEEE*, 2(4), 305-312.
- Archana, R. (2018). A survey of fault tolerance in cloud computing. *Shanlax International Journal of Arts, Science and Humanities*, 6(1), 99-104.
- Arianyan, E., Taheri, H., & Sharifian, S. (2016). Multi target dynamic VM consolidation in cloud data centers using genetic algorithm. *Journal of Information Science and Engineering*, 32, 1575-1593.
- Assunao, M. D., Gelas, J. P., Lefevre, L., & Orgerie, A. C. (2012). The Green Grid'5000: Instrumenting and using a grid with energy sensors. *In Remote Instrumentation for eScience and Related Aspects.*, 25–42.

- Atlam, H. F., Alenezi, A., Alharthi, A., Walters, R. J., & Wills, G. B. (2018). Integration of cloud computing with Internet of Things: Challenges and open issues. *IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, 670-675.
- Ayad, A., & Dippel, U. (2010). Agent-based monitoring of virtual machines. *Information Technology (ITSim), International Symposium, 1*, 1-6.
- Ayanoglu, E. (2019). Energy efficiency in data centers. *IEEE ComSoc Technical Committees Newsletter*.
- Bald'an, F. J., Gallego, S. R., Bergmeir, C., Herrera, F., & Ben'itez, J. M. (2016). A forecasting methodology for workload forecasting in cloud systems. *IEEE Transactions on Cloud Computing*, 6(4), 929-941.
- Bali, M. S., & Khurana, S. (2013). Effect of latency on network and end user domains in cloud computing. *International Conference on Green Computing, Communication and Conservation of Energy (ICGCE 2013)*.
- Barham, P., Dragovic, B., Fraser, K., Hand, S., Harris, T., Ho, A., . . . Warfield, A. (2003). Xen and the art of virtualization. *Proceedings of the 19th ACM Symposium on Operating Systems Principles (SOSP)*, 164-177.
- Barroso, L. A., & Hölzle, U. (2007). The case for energy-proportional computing. *Computer*, 40(12), 33–37.
- Barroso, L. A., & Holzle, U. H. (2009). The datacenter as a computer: An introduction to the design of warehouse-scale machines. *Synthesis Lectures on Computer Architecture*, 4, 100–108.

- Baset, S., Wang, L., & Tang, C. (2012). Towards an understanding of oversubscription in cloud. *USENIX Hot-ICE'12*.
- Bashar, A. (2013). Autonomic scaling of cloud computing resources using BN-based prediction models. *IEEE 2nd International Conference on Cloud Networking (CloudNet)*.
- Belgacem, M., & Chopard, B. (2015). A hybrid HPC/cloud distributed infrastructure: Coupling EC2 cloud resources with HPC clusters to run large tightly coupled multiscale applications. *Future Generation Computer Systems*, 42, 11-12.
- Beloglazov, A. B. (2011). A taxonomy and survey of energy-efficient data centers and cloud computing. *Journal of Advances in Computers*, 82, 47–111.
- Beloglazov, A., & Buyya, R. (2010). Adaptive threshold-based approach for energy-efficient consolidation of virtual machines in cloud data centers. *MGC*.
- Beloglazov, A., & Buyya, R. (2012). Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers. *Concurrency and Computation: Practice and Experience (CCPE)*, 24(13), 1397- 1420.
- Beloglazov, A., & Buyya, R. (2013). Managing overloaded hosts for dynamic consolidation of virtual machines in cloud data centers under quality of service constraints. *IEEE Transactions on Parallel and Distributed Systems (TPDS)*, 24, 1366–1379.
- Beloglazov, A., & Buyya, R. (2015). OpenStack Neat: A framework for dynamic and energy-efficient consolidation of virtual machines in OpenStack clouds. *Concurrency and Computation: Practice and Experience (CCPE)*, 27, 1310–1333.

- Beloglazov, A., Abawajy, J., & Buyya, R. (2012). Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. *Future Generation Computer Systems*, 28, 755–768.
- Benedict, S. (2013). Performance issues and performance analysis tools for HPC cloud applications: a survey. *Computing*, 95(2), 89-108.
- Benkhelifa, E., Hani, A. B., Welsh, T., Mithunzi, S., & Guegan, C. G. (2019). Virtual environments testing as a cloud service: A methodology for protecting and securing virtual infrastructures. *IEEE Access: Special Section on Advanced Software and Data Engineering for Secure Societies*, 7, 108660-108676.
- Berriman, B., Juve, G., Deelman, E., Regelson, M., & Plavchan, P. (2010). The application of cloud computing to astronomy: A study of cost and performance. *In Proceedings of the IEEE International Conference on e-Science. IEEE*.
- Bianchini, R., & Rajamony, R. (2004). Power and energy management for server systems. *IEEE Computer*, 37, 68-74.
- Bianchini, R., & Rajamony, R. (2010). Minimizing electricity cost: Optimization of distributed Internet data centers in a multi-electricity-market environment. *IEEE INFOCOM*, 1-9.
- Bircher, W. L., & John, L. K. (2007). Complete system power estimation: A trickle-down approach based on performance events. *International Symposium on Performance Analysis Systems and Software (ISPASS)*.
- Bohra, A. E., & Chaudhary, V. (2010). VMeter: Power modelling for virtualized clouds. *Parallel & Distributed Processing, Workshops and Phd Forum (IPDPSW), 2010 IEEE International Symposium on IEEE*, 1-8.

- Bohrer, P., Elnozahy, N. E., Keller, T., Kistler, M., Lefurg, C., McDowell, R., & Rajamony, R. (2002). The case for power management in web servers. *In Power Aware Computing, Springer*, 261–289.
- Bouwkamp, J. G., & Bolhom, J. K. (12, 1963). Dynamic response of a two-story steel frame structure. *Bulletin of the Seismological Society of America*, 56, 1289–1303.
- Bowerman, B., O'Connell, R., & Murphree, E. (2013). Business statistics in practice. 7th edition.
- Bradford, R., Kotsovinos, E., Feldmann, A., & Schioberg, H. (2007). Live wide area migration of virtual machine including local persistent state. *Proceeding of the 3rd International Conference on Virtual Execution Environments*, 169–179.
- Burnette, E. (2009). *Eclipse IDE pocket guide*. O'ReillyMedia.
- Butler, R., Welch, V., Engert, D., Foster, I., Tuecke, S., Volmer, J., & Kesselman, C. (2000). A national-scale authentication infrastructure. *Computer*, 33(12), 60-66.
- Buyya, R., Beloglazov, A., & Abawajy, J. (2010). Energy-efficient management of data center resources for cloud computing: A vision, architectural elements, and open challenges. *Proceedings of the 2010 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA 2010)*.
- Buyya, R., Broberg, J., & Goscinski, A. M. (2011). *Cloud computing principles and paradigms*. Wiley Publishing.
- Buyya, R., Yeo, C. S., Venugopal, S., Broberg, J., & Brandic, I. (2009). Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. *Future Generation Computer System*, 25, 599–616.

- Calheiros, N. (2011). CloudSim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. *Software: Practice and Experience*, 41(1), 23-50.
- Capacity planner. (n.d.). (Vmware) Retrieved 13 12, 2018, from <http://www.vmware.com/products/>
- Caravani, P., & Thomson, W. T. (1973). Identification of damping coefficients from system response. *Proceedings of the Fifth World Conference on Earthquake Engineering*. Rome.
- Castroa, P. P., Barreto, V., Corrêaa, S. L., Granville, L. Z., & Cardoso, K. V. (2016). A joint CPU-RAM energy efficient and SLA compliant approach for cloud datacenters. *Computer Networks*, 94, 1-13.
- Chakravorty, S., Mendes, C. L., & Kale, L. V. (2006). Proactive fault tolerance in MPI applications via task migration. *HiPC Ser. Lecture Notes in Computer Science*, 4297, 485-496.
- Chang, Y. C., Chang, R. S., & Chuang, F. W. (2014). *A predictive method for workload forecasting in the cloud environment*. Netherlands: Springer.
- Chase, J. S., Anderson, D. C., Thakar, P. N., Vahdat, A. M., & Doyle, R. P. (2001). Managing energy and server resources in hosting centers. *ACM SIGOPS Operating Systems Review*, 35, 103–116.
- Chaurasia, N., Tapaswi, S., & Dhar, J. (2018). A resource efficient expectation maximization clustering approach for cloud. *The Computer Journal*, 61(1), 95–104.
- Chen, H., Li, Y., & Shi, W. (2012). Fine-grained power management using process-level profiling management. *Sustainable Computing: Informatics and Systems*, 33-42.

- Chen, J., Liu, W., & Song, J. (2012). Network performance-aware virtual machine migration in data centers. *The Third International Conference on Cloud Computing, GRIDs, and Virtualization (CLOUD COMPUTING 2012)*, (pp. 65-78).
- Chen, L., Li, J., Ma, R., Guan, H., & Jacobsen, H.-A. (2020). Balancing power and performance in HPC clouds. *The Computer Journal*.
- Chen, Q., & Deng, Q. N. (2009). Cloud computing and its key techniques. *Journal of Computer Applications*, 9(29), 2562-2567.
- Chowdhury, M. R., Mahmud, M. R., & Rahman, R. M. (2015). Implementation and performance analysis of various VM placement strategies in cloudSim. *Journal of Cloud Computing*, 4, 20.
- Clark, C., Fraser, K., Hand, S., Hansen, J., Jul, E., Limpach, C., . . . Warfield, A. (2005). Live migration of virtual machines. *Proceedings of the 2nd Symposium on Networked Systems Design and Implementation (NSDI), USENIX*.
- Cleveland, W. S. (1979). Robust locally weighted regression and smoothing scatterplots. *Journal of the American Statistical Association*, 74(368).
- Cleveland, W. S., & Loader, C. (1996). Smoothing by local regression: Principles and methods. In Härdle W., Schimek M.G. (eds) *Statistical Theory and Computational Aspects of Smoothing. Contributions to Statistics*. (pp. 10-49).
- Cloud service models (IaaS, PaaS, SaaS) diagram. (n.d.). (dachou) Retrieved 13 12, 2018, from <https://dachou.github.io/2018/09/28/cloud-service-models.html>
- CloudSim: A framework for modeling and simulation of cloud computing infrastructures and services. (n.d.). (Cloudbus) Retrieved from <http://www.cloudbus.org/cloudsim/>

- Coutinho, E. F. (2014). Elasticity in cloud computing: A survey. *Annals of Telecommunications - Annales des Télécommunications*, 70, 7-8.
- Dabbagh, M., Hamdaoui, B., Guizani, M., & Rayes, A. (2016). An anergy-efficient VM prediction and migration framework for overcommitted clouds. *IEEE Transactions on Cloud Computing*, 6, 955–966.
- Daniel, J., & Abadi. (2009). Data management in the cloud: Limitations and opportunities. *Yale University USA*.
- Daradkeh, T., Agarwal, A., Zaman, M., & Goel, N. (2020). Dynamic K-means clustering of workload and cloud resource configuration for cloud elastic model. *IEEE Access*, 8, 219430-219446.
- Data center*. (n.d.). (Gartner) Retrieved 13 12, 2018, from <http://www.gartner.com/it-glossary/data-center/>
- Dean, J., & Ghemawat, S. (12, 2004). MapReduce: Simplified data processing on large clusters. *Proceeding of the 6th USENIX Symposium on Operating Systems Design and Implementation*. San.
- Deng, W., Liu, F., Jin, H., Liao, X., Liu, H., & Chen, L. (2012). Lifetime or energy: Consolidating servers with reliability control in virtualized cloud datacenters. *IEEE 4th International Conference on Cloud Computing Technology and Science*, 18–25.
- Deshpande, U., Wang, X., & Gopalan, K. (2011). Live gang migration of virtual machines. *Proceeding of International Conference on High Performance Distributed Computing*, 135-146.

- Dhanashri, B., & Divya, T. (2017). Implementation of two level scheduler in cloud computing environment. *International Journal of Computer Applications*, 3(3), 166.
- Dhiman, G., Marchetti, G., & Rosing, T. (2010). vGreen: A system for energy-efficient management of virtual machines. *ACM Transactions on Design Automation of Electronic Systems*, 1-27.
- Dhiman, G., Mihic, K., & Rosing, T. (2010). A system for online power prediction in virtualized environments using Gaussian mixture models. In *Proceedings of the 47th ACM/IEEE DAC*, 807–812.
- Egwutuoha, I., Chen, S., Levy, D., & Calvo, R. (2014). Cost-effective cloud services for HPC in the cloud: The IaaS or the HaaS. In *Handbook of Research on End-to-End Cloud Computing Architecture Design* (pp. 455-472). IGI Global.
- Family 7199+03 IBM workload deployer V3.1. (n.d.). (IBM) Retrieved 13 12, 2018, from http://www-01.ibm.com/common/ssi/ShowDoc.wss?docURL=/common/ssi/rep_sm/3/897/ENUS7199-_h03/index.html&lang=en&request_locale=en
- Fan, X., Weber, D. W., & Barroso, A. L. (2007). Power provisioning for a warehouse-sized computer. In *Proceedings of the 34th Annual International symposium on Computer Architecture, ACM*, 13–23.
- Fang, W., Liang, X., Li, S., Chiaraviglio, L., & Xiong, N. (2013). VMPlanner: Optimizing virtual machine placement and traffic flow routing to reduce network power costs in cloud data centers. *Computer Networks* 57.1, 179-196.

- Farahnakian, F., Ashraf, A., Pahikkala, T., Liljeberg, P., Plosila, J., Porres, I., & Tenhunen, H. (2015). Using ant colony system to consolidate VMs for green cloud computing. *IEEE Transactions ON Services Computing*, 8, 187–198.
- Farahnakian, F., Pahikkala, T., Liljeberg, P., Plosila, J., Hieu, N. T., & Tenhunen, H. (2016). Energy-aware VM consolidation in cloud data centers using utilization prediction model. *IEEE Transaction on Cloud Computing*, 7(2), 524-536.
- Feller, E., C. Rohr, D. M., & Morin, C. (6, 2012). Energy management in IaaS clouds: A holistic approach. *The 5th IEEE International Conference on Cloud Computing, CLOUD*, 204–212.
- Feng, D., Wu, Z., Zuo, D., & Zhang, Z. (2019). ERP: An elastic resource provisioning approach for cloud applications. *PLoS ONE*, 14(4).
- Fernandes, F., Beserra, D., Moreno, E. D., Schulze, B., & Pinto, R. C. (2016). A virtual machine scheduler based on CPU and I/O-bound features for energy-aware in high performance computing clouds. *Computers & Electrical Engineering*, 56, 854-870.
- Foster, I., & Kesselman, C. (2003). *The Grid 2: Blueprint for a new computing infrastructure*. San Francisco: Elsevier.
- Foster, I., Zhao, Y., Raicu, I., & Lu., S. (2008). Cloud computing and grid computing 360-degree compared. *Proceeding of IEEE Grid Computing Environments Workshop*, (pp. 1–10). IEEE Press.
- Gachunga, K. M. (2018). Capacity management strategies and operational efficiency in the energy sector in Kenya. Nairobi: University of Nairobi.
- Gandhi, S. (2020). *Vritti: Guarantees for mix flows in inter-datacenter networks*. Raleigh: North Carolina State University.

- Gao, J., Wang, H., & Shen, H. (2019). Task failure prediction in cloud data centers using deep learning. *Proceeding of IEEE Bigdata*.
- Garraghan, P., Moreno, I. S., Townend, P., & Xu, J. (2011). An analysis of failure-related energy waste in a large-scale cloud environment. *IEEE Transactions on Emerging Topics in Computing*, 2(2), 166-180.
- Garraghan, P., Ouyang, X., Townend, P., & Xu, J. (2015). Timely long tail identification through agent based monitoring and analytics. *2015 IEEE 18th International Symposium on Real-Time Distributed Computing*.
- George. (2022). *Uptimeinstitute.com*. Retrieved from Uptimeinstitute.com: <https://uptimeinstitute.com/resources/research-and-reports>
- Ghobaei-Arani, M., Jabbehdari, S., & Pourmina, M. A. (2018). An autonomic resource provisioning approach for service-based cloud applications: A hybrid approach. *Future Generation Computer Systems*, 78, 191-210.
- Ghribi, C., Hadji, M., & Zeghlache, D. (2013). Energy efficient VM scheduling for cloud data centers: Exact allocation and migration algorithms. *13th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid)*, 671-678.
- Gienger, M. (2015). High performance computing in the cloud: A survey on performance and usability. In *Sustained Simulation Performance 2015* (pp. 29–40). Springer.
- Gmach, D., Rolia, J., Cherkasova, L., & Kemper, A. (2009). Resource pool management: Reactive versus proactive or lets be friends. *Computer Networks*, 53, 2905–2922.
- Gmach, D., Rolia, J., Cherkasova, L., Belrose, G., Turicchi, T., & Kemper, A. (2008). An integrated approach to resource pool management: Policies, efficiency and quality

- metrics. In *Proceeding of the 38th IEEE Intl. Conf. on Dependable Systems and Networks (DSN)*, 326–335.
- Google compute engine. <https://cloud.google.com/products/> (2018).
- Goscinski, A., Brock, M., & Church, P. (2012). *High performance computing clouds, in cloud computing: Methodology, systems and applications*. Boca Raton: CRC Press.
- Grance, T., & Mell, P. (2009). The NIST definition of cloud computing. *National Institute of Standards and Technology (NIST)*.
- Guenter, B., Jain, N., & Williams, C. (2011). Managing cost, performance, and reliability tradeoffs for energy-aware server provisioning. *Proceeding of IEEE INFOCOM*, 1332–1340.
- Guimaraes, F. (2003). *Research: Anyone can do it*. Retrieved from Pedia Press.
- Gupta. (2014). Techniques for effieicnt high performance computing in the cloud. *A Doctorial Thesis (University of Illinois at Urbana-Champaign)*.
- Gupta, A., & Milojicic, D. (2011). Evaluation of HPC applications on cloud. In *Proceedings of the Sixth Open Cirrus Summit (OCS)*.
- Gupta, Kale, L. V., Milojicic, D., Faraboschi, P., & Balle, S. M. (2013). HPC-aware VM placement in infrastructure clouds. *2013 IEEE International Conference on Cloud Engineering (IC2E)*, (pp. 11–20).
- Gupta, Kale, L., Gioachin, F., March, V., Suen, C., Lee, B.-S., . . . Milojicic, D. (2012). The who, what, why and how of high performance computing applications in the cloud. In *Proceedings of the 5th IEEE International Conference on Cloud Computing Technology and Science (CloudCom'13)*.

- Gupta, R., Verma, S., & Janjua, K. (2018). Custom application development in cloud environment: Using Salesforce. *2018 4th International Conference on Computing Sciences (ICCS)*, 23-27.
- Guyon, D. O. (2018). Energy-Efficient IaaS-PaaS Co-design for Flexible Cloud Deployment of Scientific Applications. *IEEE*, 69-76.
- Hacking, S., & Hudzia, B. (2009). Improving the live migration process of large enterprise applications. *Proceeding of the 3rd International Workshop on Virtualization Technologies in Distributed Computing*, 51–58.
- Hamzaoui, I., Duthil, B., Courboulay, V., & Medromi, H. (2020). A survey on the current challenges of energy-efficient cloud resources management. *SN Computer Science*, 1(73).
- Han, G., Que, W., Jia, G., & Shu, L. (2016). An efficient virtual machine consolidation scheme for multimedia cloud computing. *Sensors*, 1–17.
- Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning: Data mining, inference, and prediction*. Springer.
- He, Q., Zhou, S., Kobler, B., Duffy, D., & McGlynn, T. (2010). Case study for running HPC applications in public clouds. *Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing*, (pp. 395–401).
- Helali, R., Azzouna, N. B., & Ghedira, K. (2016). QOS prediction in ubiquitous environments: An MLR based service selection approach. *International Wireless Communications and Mobile Computing Conference (IWCMC)*, IEEE, 904-908.

- Hines, M., & Gopalan, K. (2009). Post-copy based live virtual machine migration using adaptive pre-paging and dynamic self ballooning. *Proceeding of International Conference on Virtual Execution Environments*, 51–60.
- Höfer, C., & Karagiannis, G. (2011). Cloud computing services: taxonomy and comparison. *Journal of Internet Services and Applications*, 2(2), 81-94.
- HP cloud services. <http://hpcloud.com/>. (2018).
- Hu, B., Lei, Z., Lei, Y., & Xu, D. (12, 2011). A time-series based precopy approach for live migration of virtual machines. *Proceeding of the IEEE 17th International Conference on Parallel and Distributed Systems (ICPADS)*, 947–952.
- Huang, L., Chen, H. S., & Hu, T. T. (2013). Survey on resource allocation policy and job scheduling algorithms of cloud computing. *Journal of Software*, 8(2), 480-487.
- Hurwitz, J., Bloor, R., Kaufman, M., & Halper, F. (2010). *Cloud computing for dummies*. Chichester: Wiley Publishing.
- Hyndman, R., & Athanasopoulos, G. (April, 2018). *Forecasting: Principles*. Retrieved 11 12, 2018, from <http://otexts.org/fpp/>
- Ibrahim, M. (2012). *A parametric study on* MSc. Thesis, Faculty of Engineering, Cairo University, Giza.
- IEA – International Energy Agency. (n.d.). Retrieved from IEA – International Energy Agency: <https://www.iea.org/>
- Ignatyev, D. I., Shin, H., & Shine, A. T. (2018). Bayesian calibration for multiple source regression model. *Neurocomputing*, 318, 55–64.

- Ilushkin, A., Ali-Eldin, A., Herbst, N., Bauer, A., Papadopoulos, A. V., Epema, D., & Iosup, A. (2018). An experimental performance evaluation of autoscalers for complex workflows. *ACM Transactions on Modeling and Performance Evaluation of Computing Systems*, 3(2), 1-32.
- Indu, G., Pawanesh, A., Pooja, G., Rohit, U., & Sandeep, S. (2011). Cloud computing over cluster, grid computing: A comparative analysis. *Journal of Grid and Distributed Computing*, 01–04.
- INFORMS web site. (1, 2012). *INFORMS web site*.
- Iqbal, W., Dailey, M. N., Carrera, D., & Janecek, P. (2011). Adaptive resource provisioning for read intensive multi-tier applications in the cloud. *Future Generation Computer Systems*, 27(6), 871-879.
- Ismaeel, S., & Miri, A. (2016). Energy-consumption clustering in cloud data centre. *Proceedings of the 3rd IEEE MEC International Conference on Big Data and Smart City ICBDSM. IEEE. Muscut*.
- Ismaeel, S., Karim, R., & Miri, A. (2018). Proactive dynamic virtual-machine consolidation for energy conservation in cloud data centres. *Journal of Cloud Computing*, 7.
- Ismaeel, S., Miri, A., & Chourishi, D. (2015). Using the Extreme Learning Machine (ELM) technique for heart disease diagnosis. *Proceedings of The IEEE International Conference on Humanitarian Technology Conference (IHTC2015)*.
- Jason, W. (2000). Prediction in multiple regression. *Practical Assessment, Research and Evaluation*.

- Jayanthi, S. (2014). Literature review: Dynamic resource allocation mechanism in cloud computing environment. *Electronics, Communication and Computational Engineering (ICECCE), International Conference on* (pp. 279-281). IEEE.
- Jheng, J.-J., Tseng, F.-H., Chao, H.-C., & Chou, L.-D. (2014). A novel VM workload prediction using grey forecasting model in cloud data center. *The International Conference on Information Networking IEEE (ICOIN2014)*, 40-45.
- Jiang, D., Huang, P., Lin, P., & Jiang, J. (2012). Energy efficient VM placement heuristic algorithms comparison for cloud with multidimensional resources. *Information Computing and Applications*, 413-420.
- Jin, H., Deng, L., Wu, S., Shi, X., & Pan, X. (2009). Live virtual machine migration with adaptive memory compression. *IEEE Cluster, New Orleans*.
- Jing, S.-Y., Ali, S., She, K., & Zhong, Y. (2013). State-of-the-art research study for green cloud computing. *The Journal of Super Computing*, 65(1), 445-468.
- Jobava, A. (2015). Intelligent traffic-aware consolidation of virtual machines in a data center. Department of Informatics. Oslo University.
- Jobava, A., Yazidi, A., Oommen, B. J., & Begnum, K. (2018). On achieving intelligent traffic-aware consolidation of virtual machines in a data center using Learning Automata. *Journal of Computational Science*, 24, 290-312.
- Joshiyara, H. (2013). Energy efficient provisioning of virtual machines in cloud system using DVFS. *PARIPEX - Indian Journal of Research*, 3.
- Jouppi, N. P., Young, C., Patil, N., & Patterson, D. (2018). A domain-specific architecture for deep neural networks. *Communications of the ACM*, 61, 50-59.

- Jung, G., Joshi, R. K., Schlichting, D. R., & Pu, C. (2008). Generating adaptation policies for multi-tier applications in consolidated server environments. *Proceedins of the 5th IEEE International Conference on Autonomic Computing (ICAC 2008)*, 23–32.
- Kansal, A., Zhao, F., Kothari, N., & Bhattacharya, A. (2010). Virtual machine power metering and provisioning. *Proceedings of ACM Symposium on Cloud Computing. Indianapolis, IN, USA: ACM Press*, 39–50.
- Kaur, K., & Rai, A. K. (2014). A comparative analysis: Grid, cluster and cloud computing. *International Journal of Advanced Research in Computer and Communication Engineering*, 3, 5730–5734.
- Kaushar, H., Ricchariya, P., & Motwani, A. (2014). Comparison of SLA based energy efficient dynamic virtual machine consolidation algorithms. *International Journal of Computer Applications*, 102, 0975–8887.
- Keahey, K. (2009). Nimbus: Open source Infrastructure-as-a-Service cloud computing software. *Workshop on Adapting Applications and Computing Services to Multi-Core and Virtualization, CERN, Switzerland*.
- Kecskemeti, G., Nemeth, Z., Kertesz, A., & Ranjan, R. (2018). Cloud workload prediction based on workflow execution time discrepancies. *Cluster Computing*, 737-755.
- Khan, A. A., Zakarya, M., Rahman, U. I., Khan, R., & Buyya, R. (2020). HeporCloud: An energy and performance efficient resource orchestrator for hybrid heterogeneous cloud computing environments. *Journal of Network and Computer Applications*.
- Khanna, G., Beaty, K., Kar, G., & Kochut, A. (2006). Application performance management in virtualized server environments. *Network Operations and Management Network Symposium, 2006. NOMS 2006. 10th IEEE/IFIP*, 373-381.

- Kim, K. H., Beloglazov, A., & Buyya, R. (2011). Power-aware provisioning of virtual machines for real-time cloud services. *Concurrency and Computation: Practice and Experience*, 1–19.
- Kim, K., Wang, W., Qi, Y., & Humphrey, M. (2016). Empirical evaluation of workload forecasting techniques for predictive cloud resource scaling. In *2016 IEEE 9th International Conference on Cloud Computing (CLOUD)*, 1-10.
- Kivity, A., Kamay, Y., Laor, D., Lublin, U., & Liguori, A. (2007). KVM: The Linux virtual machine monitor. *Proceedings of the Linux Symposium*, 1, 225-230.
- Kocoloski, B., Ouyang, J., & Lange, J. (2012). A case for dual stack virtualization: Consolidating HPC and commodity applications in the cloud. In *Proceedings of the Third ACM Symposium on Cloud Computing*. ACM, 23.
- Koller, R., Verma, A., & Neogi, A. (2010). Wattapp: An application aware power meter for shared data centers. *Proceedings of International Conference on Autonomic Computing*. Reston, VA, USA: ACM Press, 31-40.
- Konstanteli, K., Cucinotta, T., Psychas, K., & Varvarigou, T. (2014). Elastic admission control for federated cloud services. *IEEE Transactions on Cloud Computing*, 2(3), 348–361.
- Koyano, S., Ata, S., Oka, I., & Inoue, K. (2012). A highgrained traffic prediction for microseconds power control in energy-aware routers. *IEEE/ACM International Conference on Utility and Cloud Computing (UCC)*.
- Kraemer, A., Maziero, C., Richard, O., & Trystram, D. (2018). Reducing the number of response time service level objective violations by a cloud-HPC convergence scheduler. *Concurrency and Computation: Practice and Experience*, 30, e4352.

- Kumar, K. D., & Umamaheswari, E. (2018). Resource provisioning in cloud computing: Using prediction models: A survey. *International Journal of Pure and Applied Mathematics*, 119(9), 1314-3395.
- Kumar, P., & Rai, K. A. (2014). An overview and survey of various cloud simulation tools. *Journal of Global Research in Computer science*, 24–26.
- Kuo-Bin, L. (2003). ClustalW-MPI: ClustalW analysis using distributed and parallel computing. *Bioinformatics*, 19(12), 1585-1586.
- Kusic, D., Kephart, O. J., Hanson, E. J., Kandasamy, N., & Jiang, G. (2009). Power and performance management of virtualized computing environments via look ahead control. *Cluster Computing*, 1–15.
- LaCurts, K. L. (2014). Application workload prediction and placement in cloud computing systems. Massachusetts: Massachusetts Institute of Technology 2014.
- Lee, J., Turner, Y., Lee, M., Popa, L., Banerjee, S., Kang, J.-M., & Sharma, P. (2014). Application-driven bandwidth guarantees in datacenters. *Proceedings of the 2014 ACM Conference on SIGCOMM ACM*.
- Lee, S., Panigrahy, R., Prabhakaran, V., Ramasubramanian, V., Talwar, K., Uyeda, L., & Wieder, U. (2011). Validating heuristics for virtual machines consolidation. *Microsoft Research, MSR-TR-2011-9*, 1–14.
- Leite, A. F. (2015). A user-centered and autonomic multi-cloud architecture for high performance computer applications. Paris: Computer Aided Engineering. Universite Paris.

- Lenk, A., Klems, A., Nimis, J., Tai, S., & Sandholm, T. (2009). *What's inside the cloud? An architectural map of the cloud landscape*. Washington, DC, USA. IEEE Computer Society: IEEE.
- Leverich, J., Monchiero, M., Talwar, V., Ranganathan, P., & Kozyrakis, C. (2009). Power management of datacenter workloads using per-core power gating. *Computer Architecture Letters*, 8, 48–51.
- Li, H., Zhu, G., Cui, C., Tang, H., Dou, Y., & He, C. (2016). Energy-efficient migration and consolidation algorithm of virtual machines in data centers for cloud computing. *Computing*, 98, 303–317.
- Lin, W., Wang, H., Zhang, Y., Qi, D., Wang, J. Z., & Chang, V. (2018). A cloud server energy consumption measurement system for heterogeneous cloud environments. *Information Sciences*, 468, 47-62.
- Liu, C., Shang, Y., Duan, L., Chen, S., & Chen, J. (2015). Optimizing workload category for adaptive workload prediction in service clouds. *Proceedings of the 13th International Conference on Service-Oriented Computing*. Springer-Verlag Berlin Heidelberg. India.
- Liu, G., Sun, H., Bai, W., Li, H., Ren, Z., Zhang, Z., & Yu, L. (2018). A learning-based system for predicting sport injuries. *MATEC Web of Conferences*, 189(1008).
- Liu, X., Wu, J., Sha, G., & Liu, S. (2020). Virtual machine consolidation with minimization of migration thrashing for cloud data centers. *Mathematical Problems in Engineering*, 2020, 1-13.
- Lizhe, W., Laszewski, G. V., Dayal, J., & Wang, F. (2010). Towards energy aware scheduling for precedence constrained parallel tasks in a cluster with DVFS.

Cluster, Cloud and Grid Computing (CCGrid), 2010 10th IEEE/ACM International Conference on IEEE.

Madni, S. H., Latiff, M. S., & Coulibaly, Y. (2016). An appraisal of meta-heuristic resource allocation techniques for IaaS cloud. *Indian Journal of Science and Technology*, 9(4).

Magalhaes, D., Calheiros, R., Buyya, R., & Gomes, D. (2015). Workload modeling for resource usage analysis and simulation in cloud computing. *Comput. Electr. Eng.*, 47(C), 69-81.

Mahadevamangalam, S. (n.d.). Energy-aware adaptation in cloud datacenters. *Faculty of Computing, Blekinge Institute of Technology, 371 79 Karlskrona, Sweden.*

Mahesh, K., Laxmaiah, M., & Sharna, Y. K. (2019). A comparative study on Google app engine Amazon web services and Microsoft Windows Azure. *International Journal of Computer Engineering & Technology (IJCET)*, 10(1), 54-60.

Malla, S., & Christensen, K. (2019). HPC in the cloud: Performance comparison of function as a service (FaaS) vs infrastructure as a service (IaaS). *Internet Technology Letters*.

Malviya, P., & Singh, S. (2009). A study about green computing. *International Journal of Advanced Research in Computer Science and Software Engineering*, 3, 6248–6252.

Marathe, A., Harris, R., Lowenthal, D., Supinski, B., Rountree, B., & Schulz, M. (2014). Exploiting redundancy for cost-effective, time-constrained execution of HPC applications on Amazon EC2. *In Proceedings of the 23rd International Symposium on High-Performance Parallel and Distributed Computing. ACM.*

- Marathe, A., Harris, R., Lowenthal, D., Supinski, B., Rountree, B., Schulz, M., & Yuan, X. (2013). A comparative study of high-performance computing on the cloud. *In Proceedings of the 22nd International Symposium on High-Performance Parallel and Distributed Computing (HPDC'13)*.
- Maringanti, H. (2022). A Decision Making Paradigm for Software Development in Libraries. In Research Anthology on Agile Software, Software Development, and Testing. *IGI Global*, 1444-1457.
- Marozzo, F. (2018). Infrastructures for high-performance computing: Cloud infrastructures.
- Marques, J. A. (2021). Harnessing Cloud Computing to Power Up HPC Applications: The BRICS CloudHPC Project. *In International Conference on Computational Science and Its Applications*,. Springer.
- Marshall, P., Tufo, H., & Keahey, K. (2013). High-performance computing and the cloud: A match made in heaven or hell? *XRDS: Crossroads, The ACM Magazine for Students*, 19(3), 52-57.
- Marzio, M. D., Fensore, S., Panzera, A., & Taylor, C. C. (2019). Local binary regression with spherical predictors. *Statistics and Probability Letters*, 144, 30–36.
- Masdari, M., & Zangakani, M. (2020). Green cloud computing using proactive virtual machine placement: Challenges and issues. *Journal of Grid Computing*, 727-759.
- Masdari, M., Nabavi, S. S., & Ahmadi, V. (2016). An overview of virtual machine placement schemes in cloud computing. *Journal of Network and Computer Applications*, 106(127), 66.

- Mason, K., Duggan, M., Barrett, E., Duggan, J., & Howley, E. (2018). Predicting host CPU utilization in the cloud using evolutionary neural networks. *Future Generation Computer Systems*, 86, 162-173.
- Masoumzadeh, S., & Hlavacs, H. (10, 2013). Integrating VM selection criteria in distributed dynamic VM consolidation using fuzzy Q-learning. *Proceeding of the 9th International Conference on Network and Service Management (CNSM 2013)*, 332-338.
- Mateescu, G., Gentzsch, W., & Ribbens, C. (2011). Hybrid computing-where HPC meets grid and cloud computing. *Future Generation Computer Systems*, 27(5), 440-453.
- Mateo, L. (13 12, 2011). *Migration to cloud: It is all about workloads - Cloud computing news*. Retrieved from Cloud computing news: <https://www.ibm.com/blogs/cloud-computing/2011/12/13/migration-to-cloud-it-is-all-about-workloads/>
- Mauch, V., Kunze, M., & Hillenbrand, M. (2013). High performance cloud computing. *Future Generation Computer Systems*, 29(6), 1408-1416.
- Medina, V., & García, J. (2014). A survey of migration mechanisms of virtual machines. *ACM Computing Surveys (CSUR)*, 46.
- Mell, P., & Grance, T. (2011). *Technical report SP800-145, NIST information*. Maryland: National Institute of Standard and Technology.
- Minas, L., & Ellison, B. (2009). Energy efficiency for Information Technology: How to reduce power consumption in servers and data centers. *Intel Press*.
- Mittal, S. (2014). A survey of techniques for improving energy efficiency in embedded computing systems. *International Journal of Computer Aided Engineering and Technology (IJCAET)*, 6(4), 440-459.

- Mittal, S. (2014). Power management techniques for data centers: A survey. *Technical Report*.
- Mittal, S., & Zhang, Z. (2012). EnCache: Improving cache energy efficiency using a software-controlled profiling cache. *IEEE International Conference On Electro/Information Technology*.
- Modeling & simulation*. (n.d.). (AcqNotes) Retrieved from <http://acqnotes.com/acqnote/tasks/simulator-types>
- Moges, F. F., & Abebe, S. L. (2019). Energy-aware VM placement algorithms for the OpenStack neat consolidation framework. *Journal of Cloud Computing: Advances, Systems and Applications*, 8.
- Mohammadian, V. N. (2020). Comprehensive and systematic study on the fault tolerance architectures in cloud computing. *Journal of Circuits, Systems and Computers*, 29(15).
- Mohan, A., & Shine. (2013). Survey on live VM migration techniques. *International Journal of Advanced Research in Computer Engineering and Technology*, 2, 60–74.
- Monil, M. A., & Rahman, R. M. (2015). VM consolidation approach based on heuristics, fuzzy logic, and migration control. *Journal of Cloud Computing*, 5–18.
- Moore, J., Chase, J., Ranganathan, P., & Sharma, R. (2005). Making scheduling "Cool": Temperature-aware workload placement in data centers. *Proceedings of the Annual conference on USENIX Annual Technical Conference*.

- Morales-Hernández, M., Sharif, M. B., Gangrade, S., Dullo, T. T., Kao, S.-C., Kalyanapu, A., . . . Hodges, B. R. (2020). High-performance computing in water resources hydrodynamics. *Journal of Hydroinformatics*, 1217–1235.
- Moreno-Vozmediano, R., Montero, R. S., Huedo, E., & Llorente, I. M. (2019). Efficient resource provisioning for elastic Cloud services based on machine learning techniques. *Journal of Cloud Computing*, 8(5).
- Mosa, A., & Sakellariou, R. (2019). Dynamic virtual machine placement considering CPU and memory resource requirements. *2019 IEEE 12th International Conference on Cloud Computing (CLOUD)*, (pp. 196–198).
- Muehlen, M. Z., Nickerson, J. V., & Swenson, K. D. (2005). Developing web services choreography standards - The case of REST vs. SOAP. *Decision Support Systems*, 40(1), 9-29.
- Muhammad, A., & Song, W.-C. (2018). A novel framework for resource orchestration in OpenStack cloud platform. *KSII Transactions on Internet and Information Systems*, 12(11), 5404-5424.
- Muhammed, T., Mehmood, R., Albeshri, A., & Alsolami, F. (2020). HPC-smart infrastructures: A review and outlook on performance analysis methods and tools. *IEEE*, 427-451.
- Muratore, L., Lennox, B., & Tsagarakis, N. G. (2018). XBotCloud: A scalable cloud computing infrastructure for XBot powered robots.
- Mvelase, P., Sithole, H., Masoka, S., & Bembe, M. (2018). HPC in the cloud environment: Challenges, and theoretical analysis. *CSREA Press*, 94-101.

- Nagar, N., & Suman, U. (6, 2014). Architectural comparison and implementation of cloud tools and technologies. *International Journal of Future Computer and Communication*, 3, 154–160.
- Nandi, B. B., Banerjee, A., Ghosh, S. C., & Banerjee, N. (6, 2012). Stochastic VM multiplexing for datacenter consolidation. *IEEE Ninth International Conference on Services Computing*, 114–121.
- Napper, J., & Bientinesi, P. (2009). Can cloud computing reach the Top500? *Proceedings of the Combined Workshops on UnConventional High Performance Computing Workshop Plus Memory Access Workshop*, (pp. 17–20).
- Nathuji, R., & Schwan, K. (2007). Virtual power: Coordinated power management in virtualized enterprise systems. *ACM SIGOPS Operating Systems Review*, 265–278.
- Nayudu, P. P., & Sekhar, K. R. (2018). Cloud environment: A review on dynamic resource allocation schemes. *Internation Jorname of Applied Engineering Research*, 13(6), 4568-4575.
- Netto, M., Calheiros, R., Rodregues, E., Cunha, R., & Buyya, R. (2018). HPC cloud for scientific and business applications: Taxonomy, vision, and research challenges. *ACM Computing Surveys*, 51, 8-29.
- Newmark, N. M., & Resenblueth, E. (1971). *Fundamentals of earthquake engineering* (2nd ed., Vol. xx). Prentice–Hall Inc., Englewood cliffs, N.J.
- Nguyen, Q. T., Quang-Hung, N., Tuong, N. H., & Thoai, N. (2013). Virtual machine allocation in cloud computing for minimizing total execution time on each machine. *Computing, Management and Telecommunications (ComManTel), International Conference on* (pp. 241-245). *IEEE*.

- Niedbala, E. F.-W. (2022). Responsible Integration of Behavioral Science in Computer Science Research and Development. *Hawaii International Conference on System(HICSS)* , 1-10.
- Nithya, H., & John, M. R. (12, 2013). A survey on various power measuring techniques for a virtual machine. *International Journal of Advanced Research in Computer Science and Software Engineering*, 374–212.
- Niu, H. X., Li, B., & Zhao, S. (2012). Quality-assured cloud bandwidth auto-scaling for video-on-demand applications. *IEEE International Conference on Computer Communications*.
- Nurmi, D., Wolski, R., Grzegorzcyk, C., Obertelli, G., Soman, S., Youseff, L., & Zagorodnov, D. (2009). The Eucalyptus open-source cloud-computing system. *Proceedings of the 2009 9th IEEE/ACM International Symposium on Cluster Computing and the Grid, CCGRID '09 IEEE Computer Society*.
- Nwobodo, I. (4, 2015). Cloud computing: A detailed relationship to grid and cluster computing. *International Journal of Future Computer and Communication*, 4, 5730–5734.
- Oliker, L., Canning, A., Carter, J., Shalf, & Ethier, S. (2004). Scientific computations on modern parallel vector systems. *Proceedings of SC04 International Conference for High Performance Computing Networking, Storage, and Analysis*. Pittsburgh.
- Orgerie, C.-A., Assuncao, M., & Lefevre, L. (2014). A survey on techniques for improving the energy efficiency of large-scale distributed systems. *ACM Computing Surveys (CSUR)*, 46, 1–31.

- Padala, P., Hou, K.-Y., Shin, K., Zhu, X., Uysal, M., Wang, Z., . . . Merchant, A. (2009). Automated control of multiple virtualized resources. *Proceedings of the 4th ACM European conference on Computer systems, EuroSys '09*. New York.
- Pandey, R., & Gonnade, S. (2014). Comparative study of simulation tools in cloud computing environment. *International Journal of Scientific and Engineering Research*, 5.
- Panneerselvam, J. (2018). A prescriptive analytics approach for energy efficiency in datacenters. *[PhD Thesis]*.
- Panneerselvam, J., Liu, L., & Antonopoulos, N. (2018). An approach to optimise resource provision with energy-awareness in datacentres by combating task heterogeneity. *IEEE Transactions on Emerging Topics in Computing*.
- Panneerselvam, J., Liu, L., Hardy, J., & Nick. (2017). Analysis, modeling, and characterization of Zombie servers in large-scale cloud datacenters. *IEEE Access*, 5, 1508-1514.
- Parallel workloads archive : MetaCentrum. (2022). *Parallel workloads archive : MetaCentrum*.
- Parashar, M., AbdelBaky, M., Roderio, I., & Devarakonda, A. (2013). Cloud paradigms and practices for computational and data-enabled science and engineering. *Computing in Science & Engineering*, 15(4), 10-18.
- Park, J. N., Park, N., & George, M. K. (2003). Multiple Regression with Dependent Dummy Variable (MRDDV) model as an Ad-hoc Risk Management System (ARMS). *IASTED Applied Simulation and Modeling (ASM)*.

- Patel, J., Jindal, V., Yen, I.-L., Bastani, F., Xu, J., & Garraghan, P. (2015). Workload estimation for improving resource management decisions in the cloud. *2015 IEEE Twelfth International Symposium on Autonomous Decentralized Systems*.
- Pepple, K. (2011). *Deploying OpenStack*. O'ReillyMedia.
- Pialorsi, P., & Russo, M. (2007). *Introducing Microsoft LINQ*. Microsoft Press.
- Prakash, K., & Edwin, B. (2013). Survey about power management techniques for high performance data centers in cloud environment. *International Journal of Engineering Research and Technology (IJERT)*, 2.
- Prasad, V. K., & Bhavsar, M. D. (2020). Monitoring and prediction of SLA for IoT based cloud. *Scalable Computing: Practice and Experience*, 21(3), 349-357.
- Qi, Z., Lu, C., & Raouf, B. (4, 2010). Cloud computing: State-of-the-art and research challenges. *Springer*, 7–18.
- Qu, C., Calheiros, R. N., & Buyya, R. (2018). Auto-scaling web applications in clouds: A taxonomy and survey. *ACM Computing Surveys*.
- Rackspace. <http://rackspace.com> (2018).
- Rahman, M. (2016). Improved Virtual Machine (VM) based resource provisioning in cloud computing. A Thesis submitted to the Faculty of Graduate Studies of The University of Manitoba.
- Rajkumar, B., Shin, Y. C., Srikumar, V., James, B., & Ivona, B. (2009). Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. *Future Generation Computer Systems*, 599–616.

- Rak, M., Turtur, M., & Villano, U. (2015). Early prediction of the cost of cloud usage for HPC applications. *Scalable Computing: Practice and Experience*, 16(3), 303–320.
- Ramos, L., & Bianchini, R. (2008). C-Oracle: Predictive thermal management for data centers. In *IEEE 14th International Symposium on High Performance Computer Architecture (HPCA)*, 111–122.
- Ranganathan, P., Leech, P., Irwin, D., & Chase, J. (2006). Ensemble-level power management for dense blade servers. *Proceedings of the 33rd International Symposium on Computer Architecture (ISCA)*, 66–77.
- Redkar, T., & Guidici, T. (2011). *Windows Azure Platform*. Apress.
- Regassa, D. Y. ((2022). Harvesting the Aggregate Computing Power of Commodity Computers for Supercomputing Applications. *Applied Sciences*, 12(10).
- Rehr, J. J., Vila, F. D., Gardner, J. P., Svec, L., & Prange, M. (2010). Scientific computing in the cloud. *Computing in Science & Engineering*, 12, 34–43.
- Renugadevi, T., Geetha, K., Prabakaran, N., & Siano, P. (2020). Carbon-efficient virtual machine placement based on dynamic voltage frequency scaling in geo-distributed cloud data centers. *Applied Sciences*, 0-24.
- Riteau, P., Morin, P., & Priol, T. (2011). Shrinker: Efficient wide area live virtual machine migration using distributed content based addressing. *Proceeding of 17th International Conference on Parallel Processing*, 431–442.
- Robert, W. (5, 2009). Reflections cloud computing. *IEEE Spectrum*.

- Rodrigo, C., & Rajiv, R. (1, 2011). CloudSim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. *Wiley Press, New York, USA*, 23–50.
- Rolia, J., Cherkasova, L., Arlitt, M., & Andrzejak, A. (2005). A capacity management service for resource pools. *Proceedings of the 5th International Workshop on Software and Performance, WOSP '05*.
- Ropars, T., Guermouche, A., Uçar, B., Meneses, E., V. Kalé, L., & Cappello, F. (2011). On the use of cluster-based partial message logging to improve fault tolerance for MPI HPC applications. *Euro-Par 2011 Parallel Processing*.
- Ruiz, P., & Penzien, J. (3, 1969). *Probabilistic study on the behavior of structures during earthquake*. Report, Earthquake Engineering Research Center, University of California, Berkeley, California.
- Runsewe, O. A. (2019). A novel cloud broker-based resource elasticity management and pricing for big data streaming applications. *Thesis*. Ottawa, Canada: University of Ottawa.
- Sadashiv, N., & Kumar, S. D. (2011). Cluster, grid and cloud computing: A detailed comparison. *Proceeding of the 6th International Conference on Computer Science and Education (ICCSE)*, (pp. 477–482).
- Sahni, S., & Varma, V. (2012). A hybrid approach to live migration of virtual machines. *International Conference on Cloud Computing in Emerging Markets, India*.
- Sapuntzakis, C., Chandra, R., Pfaff, B., Chow, J., Lam, M., & Rosenblum, M. (12, 2002). Optimizing the migration of virtual computers. *Proceeding of the 5th Symposium on Operating Systems Design and Implementation (OSDI-02)*.

- Saravanan, S., Venkatachalam, V., & Malligai, S. (2015). Optimization of SLA violation in cloud computing using artificial bee colony. *International Journal of Advances in Engineering*, 410–414.
- Sayedkhan, P. (2014). Virtual machine placement based on disk I/O load in cloud. *International Journal of Computer Science and Information Technologies*, 5, 5477–5479.
- Sefraoui, O., Aissaoui, M., & Eleuldj, M. (10, 2012). OpenStack: toward an open-source solution for cloud computing. *International Journal of Computer Applications*, 55(3), 38-42.
- Sekhar, J., G.Jeba, & Durga, S. (2012). A survey on energy efficient server consolidation through VM live migration. *International Journal of Advances in Engineering and Technology (IJAET)*, 5, 515–525.
- Sempolinski, P., & Thain, D. (11, 2010). A comparison and critique of Eucalyptus, OpenNebula and Nimbus. *IEEE International Conference on Cloud Computing Technology and Science*, 417–426.
- Sharma, G., & Rai, A. (2019). A review on open source solution for cloud computing. *Journal of Business School*, 2(3), 1-6.
- Sharma, N., & Maurya, S. (2019). SLA-cased agile VM management in cloud & datacenter. *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)*, 252-257.
- Sharma, O., & Saini, H. (2016). VM consolidation for cloud data center using median based threshold approach. *Twelfth International Multi-Conference on Information Processing-2016 (IMCIP-2016)*, *Procedia Computer Science*, 89, 27–33.

- Sharma, S., & Chawla, M. (11, 2013). A technical review for efficient virtual machine migration. *International Conference on Cloud and Ubiquitous Computing and Emerging Technologies*, 20-25.
- Shehabi, A., Smith, S., Sartor, D., Brown, R., Herrlin, M., Koomey, J., . . . Lintner, W. (2016). *United States data center energy usage report*. Tech. rep., Lawrence Berkeley National Lab. (LBNL), Berkeley, CA (United States).
- Shi, Y., Jiang, X., & Ye, K. (2011). An energy-efficient scheme for cloud resource provisioning based on CloudSim. *Cluster Computing (CLUSTER), IEEE International Conference*, 26–30.
- Shim, Y.-C. (2016). Performance evaluation of static VM consolidation algorithms for cloud based data centers considering inter-VM performance interference. *International Journal of Applied Engineering Research*, 11(24), 11794-11802.
- Shoaib, Y., & Das, O. (2014). Performance-oriented cloud provisioning: Taxonomy and survey. *arXiv preprint arXiv:1411.5077*.
- Shuja, J., Bilal, K., Madani, S. A., Othman, M., Ranjan, R., Balaji, P., & Khan, S. U. (2014). Survey of techniques and architectures for designing energy-efficient data centers. *IEEE SYSTEMS Journal*, 1–13.
- Sigh, N., & Bal, J. (2017). Energy efficient data center: A systematic literature review. *Advances in Computational Science and Technology, Research India Publications*, 10, 121-128.
- Singh, A., & Kinger, S. (2013). Virtual machine migration policies in clouds. *IJSR*, 2, 364–367.

- Singh, J. Y., & Kirit, M. (2012). Cloud computing concepts, architecture and challenges. *International Conference on Computing, Electronics and Electrical Technologies [ICCEET]*, 877–880.
- Singh, N., & Dhir, V. (2017). A literature review: Efficient VM migration techniques for energy reduction in cloud computing. *International Journal of Advanced Research in Computer Science*, 8, 98-114.
- Singh, P., Gupta, P., Jyoti, K., & Nayyar, A. (2019). Research on auto-scaling of web applications in cloud: Survey, trends and future directions. *Special Issue Papers*, 20(2).
- Singh, S., & Chana, I. (2016). Resource provisioning and scheduling in clouds: QoS perspective. *Supercomputer*, 72(3), 926-960.
- Skovira, J. C. (2000). The easy—loadleveler api project. In *Workshop on Job Scheduling Strategies for Parallel Processing Springer, Berlin, Heidelberg.*, 41-47.
- Somasundaram, T. S., & Govindarajan, K. (2014). CLOUDRB: A framework for scheduling and managing High-Performance Computing (HPC) applications in science cloud. *Future Generation Computer Systems*, 34, 47-65.
- Song, B., Yu, Y., Zhou, Y., Wang, Z., & Du, S. (2018). Host load prediction with long short term memory in cloud computing. *The Journal of Supercomputing*, 74(12), 6554-6568.
- Souza, A., Rezaei, M., Laure, E., & Tordsson, J. (2019). Hybrid resource management for HPC and data intensive workloads. *19th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, CCGrid 2019, May 14-17, 2019, Larnaca, Cyprus*, (pp. 399–409).

- Stantchev, V. (2009). Performance evaluation of cloud computing offerings. *2009 Third International Conference on Advanced Engineering Computing and Applications in Sciences*, (pp. 187–192).
- Sun, Z., Li, L., Xing, X., Lv, Z., & Xiong, N. N. (2019). A novel nodes deployment assignment scheme with data association attributed in wireless sensor networks. *Journal of Internet Technology (JIT)*, 20, 509-520.
- Sundarakani, B. K. (2019). Designing a hybrid cloud for a supply chain network of Industry 4.0: a theoretical framework. *International Journal, Emerald*.
- Takako, P., Estcio, G., Kelner, J., & Sadok, D. (5, 2010). A survey on open-source cloud computing solutions. *WCGA - 8th Workshop on Clouds, Grids and Applications*, 3–16.
- Tamiz, M., Jones, D., & Romero, C. (1998). Goal programming for decision making: An overview of the current state-of-the-art. *European Journal of Operational Research*, 111, 569–581.
- Tang, M., & Pan, S. (2015). A hybrid genetic algorithm for the energy-efficient virtual machine placement problem in data centers. *Neural Process Letters*, 41, 211-221.
- Tang, Q., Mukherjee, T., Gupta, S., & Cayton, P. (2006). Sensor-based fast thermal evaluation model for energy efficient high-performance datacenters. *In Proceedings of the Intelligent Sensing and Information Processing (ICISIP 2006) Fourth International Conference on Bangalore*, 203-208.
- Tang, S., Lee, B., & He, B. (2014). DynamicMR: A dynamic slot allocation optimization framework for Mapreduce clusters. *IEEE Transactions on Cloud Computing*, 2(3), 333- 347.

- Tavares, W. F. (2021). Quantifying and detecting HPC resource wastage in cloud environments. In *2021 International Symposium on Computer Architecture and High Performance Computing Workshops (SBAC-PADW)*, 41-46.
- Thakral, D., & Singh, M. (2014). Virtualization in cloud computing. *International Journal of Computer Science and Mobile Computing*, 3, 1262-1273.
- Thakur, S., Kalia, A., & Thakur, J. (2013). Server consolidation algorithms for cloud computing environment: A review. *International Journal of Advanced Research in Computer Science and Software Engineering*, 3, 379–384.
- The gaia HPC workload. (2014). *The gaia HPC workload*.
- The Green Grid Consortium. (2011). *The Green Grid Consortium*.
- (2011). *The Magellan report on cloud computing for science. Technical report*. U.S. Department of Energy Office of Science Office of Advanced Scientific Computing Research (ASCR).
- The Message Passing Interface (MPI) standard*. (1991). Retrieved from <http://www-unix.mcs.anl>.
- The Metacentrum HPC workload. (2014). *The Metacentrum HPC workload*.
- The OpenMP API specification for parallel programming*. (1997). Retrieved from <https://www.openmp.org/>
- Tian, W., Zhao, Y., Zhong, Y., Xu, M., & Jing, C. (2011). A dynamic and integrated load balancing scheduling algorithm for cloud datacenters. *Proceedings of IEEE CCIS2011*, 311-315.

- Tomašević, M., Lapuh, L., Stević, Ž., Stanujkić, D., & Karabašević, D. (2020). Evaluation of criteria for the implementation of High-Performance Computing (HPC) in Danube region countries using fuzzy PIPRECIA method. *Sustainability*, 1-18.
- Trabelsi, M., & Frigui, H. (2019). Robust fuzzy clustering for multiple instance regression. *Pattern Recognition*, 90, 424–435.
- Tulloch, M. (2013). *Introducing Windows Azure for IT professionals*. Redmond: Microsoft Press.
- Uddin, M., Shah, A., Alsaqour, R., & Memon, J. (2013). Measuring efficiency of tier level data centers to implement green energy efficient data centers. *Middle-East Journal of Scientific Research*, 15, 200–207.
- Urul, G. (2018). Energy efficient dynamic virtual machine allocation with CPU usage prediction in cloud datacenters. The graduate school of engineering and science of Bilkent University.
- Usman, M. J., Ismail, A. S., Abdul-Salaam, G., Chizari, H., Kaiwartya, O., Gital, A. Y., . . . Dishing, S. I. (2019). Energy-efficient nature-inspired techniques in cloud computing datacenters. *Telecommunication Systems*, 71, 275–302.
- Vakilinia, S. (2018). Energy efficient temporal load aware resource allocation in cloud computing datacenters. *Journal of Cloud Computing*, 7(2), 1-24.
- Varghese, B., & Buyya, R. (2018). Next generation cloud computing: New trends and research directions. *Future Generation Computer Systems*, 79, 849-861.
- Vashistha, A., & Dwivedi, N. (2018). A review of cloud simulation tools. *International Journal of Engineering Research & Management Technologies*, 5(2), 28-31.

- Vecchiola, C., Pandey, S., & Buyya, R. (2009). High-performance cloud computing: A view of scientific applications. *2009 10th International Symposium on Pervasive Systems, Algorithms, and Networks, IEEE*, 4–16.
- Veni, T., & Bhanu, S. M. (2013). A survey on dynamic energy management at virtualization level in cloud data centers. *Computer Science and Information Technology*, 3, 107–117.
- Verma, A., Ahuja, P., & Neogi, A. (2008). pMapper: Power and migration cost aware application placement in virtualized systems. *In Proceedings of the ACM/IFIP/LISENIX Ninth International Middleware Conference*, 243–264.
- Vigliotti, F. L., & Batista, M. D. (2014). A green network-aware VMs placement mechanism. *In Proceedings of the IEEE Globecom*.
- VMware infrastructure. *Resource management with VMware DRS*. (2006). VMware Whitepaper.
- Vo, H. (2019). Researching the "Black Art" inside SAAS sales. *Thesis*. Centria University of Applied Sciences.
- Von, G. L., Wang, L., Younge, J. A., & He, X. (2009). Power-aware scheduling of virtual machines in DVFS-enabled clusters. *In Proceedings of the 2009 IEEE International Conference on Cluster Computing (Cluster 2009), New Orleans*.
- Vouk, M. V. (2008). Cloud computing – Issues, research and implementations. *Journal of Computing and Information Technology*, 16, 235–246.
- Wang, H., & Tianfield, H. (2018). Energy-aware dynamic virtual machine consolidation for cloud datacenters. *IEEE Access*, 6, 15259-15273.

- Wang, L., Laszewski, G. V., Dayal, J., He, X., Younge, J. A., & Furlani, R. T. (2009). Towards thermal aware workload scheduling in a data center. *In 10th International Symposium on Pervasive Systems, Algorithms, and Networks (ISPAN), IEEE*, 116–122.
- Wang, L., Ranjan, R., Chen, J., & Benatallah, B. (2011). *Cloud computing: Methodology, systems, and applications*. CRC Press.
- Wang, L., Ranjan, R., Chen, J., & Benatallah, B. (2017). *Cloud Computing - Methodology, Systems, and Applications*. CRC Press.
- Webscope datasets. (n.d.). Retrieved from <https://webscope.sandbox.yahoo.com/>
- Weinhardt, C., Anandasivam, A., Blau, B., Borissov, N., Meinl, T., Michalk, W., & Stößer, J. (2009). Cloud computing – A classification, business models, and research directions. *Business and Information Systems Engineering*, 391–399.
- Weissman, C. D., & Bobrowski, S. (2009). The design of the force.com multitenant Internet application development platform. *Proceedings of the 35th SIGMOD International Conference on Management of Data. ACM*.
- Wen, Y., Li, Z., Jin, S., Lin, C., & Liu, Z. (2017). Energy-efficient virtual resource dynamic integration method in cloud computing. *IEEE Access*, 5, 12214–12223.
- Windows Azure. <http://windowsazure.com> (2018).
- Witte, P. A., Modzelewski, H., Selvage, J., Louboutin, M., Jones, C., & Herrmann, F. J. (2019). An event-driven approach to serverless seismic imaging in the cloud. *IEEE*, 7, 1-27.

- Wood, T., Shenoy, P., Ramakrishnan, K., & Merve, J. (2011). Cloudnet: Dynamic pooling of cloud resources by live WAN migrations of virtual machines. *Proceeding of 7th ACM SIGPLAN/SIGOPS International Conference on Virtual Execution Environments*, 121–132.
- Wooldridge, M. J. (2012). Introductory econometrics: A modern approach, 5th edition. *South Western Cengage Learning*.
- Wu, G., Tang, M., Tian, Y., & Li, W. (2012). Energy-efficient virtual machine placement in data centers by genetic algorithm. *Neural Information Processing*, 315–323.
- Xu, R., Zhu, D., Rusu, C., Melhem, R., & Mosse, D. (2005). Energy-efficient policies for embedded clusters. *In ACM SIGPLAN Notices*, 40, 1–10.
- Yang, J., Liu, C., Shang, Y., Cheng, B., Mao, Z., Liu, C., . . . Chen, J. (2014). A cost-aware auto-scaling approach using the workload prediction in service clouds. *Information Systems Frontiers*, 16(1), 7-18.
- Yang, R., Moreno, I. S., Xu, J., & Wo, T. (2013). An analysis of performance interference effects on energy-efficiency of virtualized cloud environments. *Proceedings: IEEE 5th International Conference on Cloud Computing Technology and Science*.
- Ye, K., Che, J., He, Q., Huang, D., & Jiang, X. (2012). Performance combinative evaluation from single virtual machine to multiple virtual machine systems. *International Journal of Numerical Analysis & Modeling (IJNAM)*, 9(2), 350-371.
- Yousefipour, A., Rahmani, A. M., & Jahanshahi, M. (2018). Energy and cost-aware virtual machine consolidation in cloud computing. *Software Practice Experience*, 48, 1758–1774.

- Yu, Y., & Gao, Y. (2012). Constraint programming-based virtual machines placement algorithm in datacenter. *Intelligent Information Processing*, 295–304.
- Yuventi, J., & Mehdizadeh, R. (2013). A critical analysis of power usage effectiveness and its use in communicating data center energy consumption. *Energy and Buildings*, 64, 90-94.
- Zahariev, A. (2009). Google App Engine. Helsinki University of Technology.
- Zaw, E., & Thein, N. (3, 2012). Improved live VM migration using LRU and Splay Tree algorithm. *International Journal of Computer Science and Telecommunications*, 3.
- Zeng, X., Garg, S. K., Wen, Z., Strazdins, P., Y.Zomaya, A., & Ranjan, R. (2017). Cost efficient scheduling of MapReduce applications on public clouds. *Journal of Computational Science*, 26, 375-388.
- Zhang, F., Liu, G., Fu, X., & Yahyapour, R. (2018). A survey on virtual machine migration: Challenges, techniques, and open issues. *IEEE Communications Surveys & Tutorials*, 20(2), 1206 - 1243.
- Zhang, Q., Cheng, L., & Boutaba, R. (2010). Cloud computing: State-of-the-art and research challenges. *Journal of Internet Services and Applications*, 1, 7-18.
- Zhang, W., Qian, H., Wills, C., & Rabinovich, M. (2010). Agile resource management in a virtualized data center. *Proceedings of the First Joint WOSP/SIPEW International Conference on Performance Engineering ACM*.
- Zhang, X., Huo, Z., J.Ma., & Meng, D. (2010). Exploiting data duplication to accelerate live virtual machine migration. *Proceeding of the IEEE International Conference on Cluster Computing*, 88–96.

- Zhang, X., Wu, T., Chen, M., Wei, T., Zhou, J., Hu, S., & Buyya, R. (2018). Energy-aware virtual machine allocation for cloud with resource reservation. *The Journal of Systems and Software*, 147-161.
- Zhao, H., & Li, X. (2012). Designing flexible resource rental models for implementing HPC-as-a-Service in cloud. *Parallel and Distributed Processing Symposium Workshops & PhD Forum (IPDPSW), 2012 IEEE 26th International*.
- Zheng, Q., Li, R., Li, X., Shah, N., Zhang, J., Tian, F., . . . Li, J. (2016). Virtual machine consolidated placement based on multi-objective biogeography-based optimization. *Future Generation Computer Systems*, 54, 95-122.
- Zhou, A. C., Lao, J., Ke, Z., Wang, Y., & Mao, R. (2021). FarSpot: Optimizing monetary cost for HPC applications in the cloud spot market. *IEEE Transactions on Parallel and Distributed Systems*.
- Zhou, Z., Hu, Z., Song, T., & Yu, J. (2015). A novel virtual machine deployment algorithm with energy efficiency in cloud computing. *Journal of Central South University*, 22, 94–98.
- Zhu, F., Li, H., & Lu, J. (2012). A service level agreement framework of cloud computing based on the cloud bank model. *Computer Science and Automation Engineering (CSAE), IEEE*, 1, 255–259.
- Zhu, X., Young, D., Waston, J., Wang, Z., Rolia, S., Singhal, S., . . . Gardner, R. (2008). 1000 Islands: Integrated capacity and workload management for the next generation data center. *Proceedings of the 5th International Conference on Autonomic Computing (ICAC)*, 172–181.

Zoltners, A. A., Sinha, P., & Lorimer, S. E. (2006). *The complete guide to sales force incentive compensation: How to design and implement plans that work*. AMACOM Div American Mgmt Assn.