

**DESIGN AND EVALUATION OF QoS DRIVEN
ENERGY-EFFICIENT FRAMEWORK FOR
CLOUD COMPUTING**

THESIS

Submitted

in fulfilment of the requirements of the degree of

DOCTOR OF PHILOSOPHY

By

Nagma

1410931002

Supervised by

Dr. Jaiteg Singh
Dean and Professor,
Department of
Computer Applications
Chitkara University,
Punjab

Dr. Jagpreet Sidhu
Assistant Professor, Department
of Computer Science and
Information Technology
Jaypee University of Information
Technology, Himachal Pradesh

January 2020



Department of Computer Science and Engineering

CHITKARA UNIVERSITY

CHANDIGARH-PATIALA NATIONAL HIGHWAY

RAJPURA (PATIALA) PUNJAB-140401 (INDIA)

**DESIGN AND EVALUATION OF QoS DRIVEN
ENERGY-EFFICIENT FRAMEWORK FOR
CLOUD COMPUTING**

THESIS

Submitted

in fulfilment of the requirements of the degree of

DOCTOR OF PHILOSOPHY

By

Nagma

1410931002

Supervised by

Dr. Jaiteg Singh
Dean and Professor,
Department of
Computer Applications
Chitkara University,
Punjab

Dr. Jagpreet Sidhu
Assistant Professor, Department
of Computer Science and
Information Technology
Jaypee University of Information
Technology, Himachal Pradesh

January 2020



Department of Computer Science and Engineering

CHITKARA UNIVERSITY

CHANDIGARH-PATIALA NATIONAL HIGHWAY

RAJPURA (PATIALA) PUNJAB-140401 (INDIA)

CHITKARA UNIVERSITY, PUNJAB

DECLARATION BY THE STUDENT

I hereby certify that the work which is being presented in this thesis entitled “**Design and Evaluation of QoS driven Energy-Efficient Framework for Cloud Computing**” is for fulfilment of the requirement for the award of Degree of **Doctor of Philosophy** submitted in the **Department of Computer Science and Engineering, Chitkara University, Punjab** is an authentic record of my own work carried out under the supervision of **Dr. Jaiteg Singh** and **Dr. Jagpreet Sidhu**.

The work has not formed the basis for the award of any other degree or diploma, in this or any other Institution or University. In keeping with the ethical practice in reporting scientific information, due acknowledgements have been made wherever the findings of others have been cited.

Nagma

CHITKARA UNIVERSITY, PUNJAB

CERTIFICATE BY THE SUPERVISORS

This is to certify that the thesis entitled “**Design and Evaluation of QoS driven Energy-Efficient Framework for Cloud Computing**” submitted by **Nagma, Regd. No. 1410931002** to the Chitkara University, Punjab in fulfilment for the award of the degree of **Doctor of Philosophy** is a *bona fide* record of research work carried out by her under our supervision. The contents of this thesis, in full or in parts, have not been submitted to any other Institution or University for the award of any degree or diploma.

Dr. Jaiteg Singh
Dean and Professor,
Department of
Computer Applications
Chitkara University,
Punjab

Dr. Jagpreet Sidhu
Assistant Professor, Department
of Computer Science and
Information Technology
Jaypee University of Information
Technology, Himachal Pradesh

ACKNOWLEDGMENT

I am grateful to God for giving me a chance to finish my Ph.D. thesis on time. I would especially like to thank my supervisors Dr. Jaiteg Singh and Dr. Jagpreet Sidhu for providing exceptional support throughout my research. Their professional and technical knowledge guided me in the right direction and their understanding encouraged me during difficult times. I would also like to thank my friends especially Kanika Goyal, Anu Marwaha, Chanpreet Singh, Himanshu Jindal, Mukesh Bhandari, Jatin Arora, Manish Krate, Jatin Bansal, Muskaan, Dr. Jaya Madan, Dr. Rahul Pandey, Shivani Wadhwa, Shivani Gupta, Divya Gupta and Rupali Gill who devoted their valuable time and helped me in all possible ways towards the successful completion of this work.

I would also like to thank Dr. Pankaj Kumar, Professor & Dean, Ph.D. Programme, Dr. Meenu Khurana, Professor & Dean, Department of Computer Science and Engineering, Ms. Deepika Sharma, Assistant Manager, Office of Ph.D. Programme, Chitkara University for their support in achieving my goal. I thank all those who have contributed directly or indirectly to this work.

Finally, I would like to express my love and gratitude to my parents, my husband Kunal Verma, my siblings Kritika, Hitesh Khattar, Ridham Khattar and other family members who stood by me during tough time and gave unconditional support during this journey. To all those great people who believed in me and supported me through this experience, thank you.

Nagma

PUBLICATIONS

Khattar N, Sidhu J, Singh J. Toward energy-efficient cloud computing: a survey of dynamic power management and heuristics-based optimization techniques. *The Journal of Supercomputing*. 2019 Sep;75(8):1-61. **(SCI)**

Khattar N, Singh J, Sidhu J. Multi-criteria based energy-efficient framework for VM placement in cloud data centers. *Arabian Journal for Science and Engineering*. 2019 Sep;44 (11):1-5. **(SCI)**

Khattar N, Singh J, Sidhu J. An adaptive energy-efficient VM Consolidation framework for cloud data centers. *Wireless Personal Communications*. 2019 *[In Press]*. **(SCI)**

Khattar N, Sidhu J, Singh J. A correlation based investigation of VM consolidation algorithms for cloud computing. *International Journal of Cloud Computing*. 2018 *[In Press]*. **(SCOPUS)**

Khattar N, Sidhu J, Singh J. Trends of publications and work done in different areas in IoT: A survey. In Kumar P, Bakshi S, Hatzilygeroudis K, Sahoo N, editors. *Recent Findings in Intelligent Computing Techniques*. Singapore: Springer; 2019. p. 385-396. **(SCOPUS)**

Khattar N, Sidhu J, Singh J. Trends of publications and work done in different areas in energy saving in cloud computing: A survey. In Choudhury S, Mishra R, Mishra G, Kumar A, editors. *Intelligent Communication, Control and Devices*. Singapore: Springer; 2018. p. 1661-1675. **(SCOPUS)**

Khattar N, Singh J, Sidhu J. A trace driven simulation study of VM consolidation algorithms in cloud computing. In 5th International Conference on Parallel, Distributed and Grid Computing. 2018 Dec (pp. 783-788). IEEE. **(SCOPUS)**

Khattar N, Singh J, Sidhu J. Comparative analysis of VM consolidation algorithms for cloud computing. *Procedia Computer Science*. 2019. **(SCOPUS)**

COMMUNICATIONS

Khattar N, Singh J, Sidhu J. Improved PROMETHEE-based energy-efficient host selection framework for cloud data centers. *Journal of Parallel and Distributed computing*. 2019. **(SCI)**

Khattar N, Singh J, Sidhu J. A multi-criteria based novel energy efficient VM consolidation framework for cloud computing. *Journal of Supercomputing*. 2019. **(SCI)**

ABBREVIATIONS

ACO	Ant Colony Optimization
AHP	Analytical Hierarchy Processing
AMBFF	Adaptive Minimal Bound First-Fit
APA-VMP	Adaptive Power-Aware Virtual Machine Provisioner
ATM	Automated Teller Machine
CCR	Center for Computational Research
CSP	Cloud Service Provider
DRR	Dynamic Round Robin
DT	Dynamic Threshold
DVFS	Dynamic Voltage Frequency Scaling
DVMC	Dynamic Virtual Machine Consolidation
DVS	Dynamic Voltage Scaling
EA	Evolutionary Algorithm
ECS	Energy Conscious Scheduling
FCFS	First Come First Served
FF	First Fit
GA	Genetic Algorithm
GMDH	Group Method of Data Handling
HEFT	Heterogeneous Earliest Finish Time
HPC	High Performance Computing
HPG	Highest Potential Growth
HS	High CPU Utilization-based Migration VM selection
I/O	Input/Output
IaaS	Infrastructure-as-a-Service
IQR	Interquartile Range
ITEED	Improved TOPSIS Energy-Efficient Deployment
ITUDA	Improved TOPSIS Underload Detection Approach
KAI	K-Means Clustering Algorithm-Average-Interquartile Range
KAM	K-Means Clustering Algorithm-Median Absolute Deviation
KMI	K-Means Clustering Algorithm-Midrange-Interquartile range
KMIMR	K-Medoid Interquartile Mid-Range Algorithm

LR	Local Regression
LRR	Local Regression Robust
MAD	Median Absolute Deviation
MADM	Multi-Attribute Decision Making
MBFD	Modified Best Fit Decreasing
MC	Maximum Correlation
MCC	Maximum Correlation Coefficient
MCDM	Multi-Criteria Decision Making
MIPS	Million Instructions Per Second
MIQP	Mixed Integer Quadratic Programming
MM	Minimization of Migrations
MMS	Minimum Memory Size
MMT	Minimum Migration Time
MPCBM	Minimize Product of CPU Bandwidth and Memory Utilization
MPSO	Modified Particle Swarm Optimization
MPCU	Minimum the Product of CPU utilization and Memory Utilization
MRCPBM	Maximum Ratio of CPU to Product of CPU Bandwidth and Memory
MRCU	Maximum Ratio of CPU utilization and Memory Utilization
MU	Minimum Utilization
NIST	National Institute of Science and Technology
NPA	Non Power Aware
OS	Operating System
PaaS	Platform-as-a-Service
PABFD	Power-Aware Best Fit Decreasing
PDM	Performance Degradation Due to Migrations
PM	Physical Machine
PSO	Particle Swarm Optimization
QoS	Quality of Service
RCP	Random Choice Policy
RR	Round Robin
RS	Random Selection
SaaS	Software-as-a-Service

SABFD	Space-Aware Best Fit Decreasing
SAPSO	Self-aware Particle Swarm Optimization
SLA	Service Level Agreement
SLATAH	Service Level Agreement Violation Time Per Active Host
ST	Single Threshold
SLAVDA	Service Level Agreement Violation Decision Algorithm
TASA	Thermal-Aware Task Scheduling Algorithm
TASA-B	Thermal-Aware Task Scheduling Algorithm-Backfilled
THR	Threshold
TOPSIS	Technique for Order of Preference By Similarity to Ideal Solution
VM	Virtual Machine

CONTENTS

Declaration by the Student	i
Certificate by the Supervisors	ii
Acknowledgement	iii
Publications	iv
Communications	v
Abbreviations	vi-viii
Contents	ix-xi
List of Figures	xii-xiii
List of Tables	xiv
Abstract	xv-xvi
Chapter 1	1-18
Introduction.....	1
1.1 Background and related terms.....	1
1.1.1 Entities in a cloud environment.....	4
1.2 Research issues in cloud computing	6
1.3 Energy consumption by cloud data centers.....	8
1.4 Energy-efficient techniques	9
1.4.1 Infrastructure-based solutions	9
1.4.2 Hardware-based solutions	10
1.4.3 Software-based solutions.....	12
1.5 Process of VM consolidation	13
1.5.1 Over-loaded hosts detection algorithms	14
1.5.2 VM selection algorithms	15

1.5.3 VM placement algorithms	16
1.5.4 Host underload detection algorithm	16
1.6 Energy performance trade-off	17
1.7 Organization of the thesis.....	18
Chapter 2	19-40
Literature Review	19
2.1 Research gaps.....	39
2.2 Objectives.....	40
Chapter 3	41-46
Design of QoS-based Energy-Efficient Framework for Cloud Computing	
Data Centers	41-48
3.1 The proposed framework	41
3.1.1 High level architecture	42
3.1.2 Low level architecture	43
3.2 Working of the framework.....	46
Chapter 4	49-60
Experimental Setup and Implementation of the Framework.....	49
4.1 Experimental setup.....	49
4.1.1 Workload trace	50
4.1.2 Metrics	51
4.2 The proposed resource management policies	52
4.2.1 Host overload detection algorithm	52

4.2.2 VM selection methods	53
4.2.3 VM deployment/placement algorithm.....	54
4.2.4 Host underload detection algorithm	60
Chapter 5	61-74
Evaluation of proposed framework via case study/scenarios	61
5.1 Checking whether the framework is generic.....	62
5.1.1 Analysing the framework for CPU intensive tasks	62
5.1.2 Analysing the framework for I/O intensive tasks.....	65
5.2 Evaluating the framework using different number of VMs	67
5.3 Evaluating the framework for different types of data	68
5.4 Evaluating the framework using multi-core CPUs	68
5.5 Case study to evaluate the performance of proposed VM placement algorithm	69
Chapter 6	75-76
Conclusion and Future Scope	75

LIST OF FIGURES

Fig. 1.1 Characteristics of cloud computing	2
Fig. 1.2 Cloud computing service models	3
Fig. 1.3 Interaction between cloud user and service provider	4
Fig. 1.4 Type I and Type II hypervisors	5
Fig. 1.5 High level communication between cloud computing entities.....	6
Fig. 1.6 Solutions for reduction of energy consumption	10
Fig. 1.7 Infrastructure-based optimizations	10
Fig. 1.8 Architecture of a DVFS-based system	12
Fig. 1.9 VM consolidation process	14
Fig. 1.10 Scenario representing VM consolidation	15
Fig. 2.1 Classification of work done related to consolidation	19
Fig. 3.1 High level architecture of the proposed framework	43
Fig. 3.2 Algorithms used by scheduling decision module	44
Fig. 3.3 Low level architecture of the proposed framework.....	45
Fig. 3.4 Working of the proposed framework	47
Fig. 4.1 Evaluation matrix	57
Fig. 4.2 Comparison matrix	58
Fig. 5.1 Identified scenarios.....	61
Fig. 5.2 Energy efficiency under diverse L values	63
Fig. 5.3 Energy consumption under diverse L values.....	63
Fig. 5.4 SLA violations under diverse L values	63
Fig. 5.5 SLATAH under diverse L values	63
Fig. 5.6 PDM under diverse L values	63
Fig. 5.7 Number of VM migrations under diverse L values.....	63
Fig. 5.8 Energy efficiency under diverse L values	66
Fig. 5.9 Energy consumption under diverse L values.....	66
Fig. 5.10 SLA violations under diverse L values	66
Fig. 5.11 SLATAH under diverse L values	66
Fig. 5.12 PDM under diverse L values	66
Fig. 5.13 Number of VM migrations under diverse L values.....	66
Fig. 5.14 Compliance of the various attributes for physical hosts.....	72

Fig. 5.15 Relative significance of the parameters for physical hosts	73
Fig. 5.16 Relative normalized weights of the parameters for physical hosts	73
Fig. 5.17 The efficiency of the hosts generated according to ranks	74

LIST OF TABLES

Table 2.1 Summary of work done on VM consolidation	35
Table 4.1 Configuration of VMs and hosts used in experiments.....	50
Table 4.2 Power consumption in watts	50
Table 4.3 Workload traces	50
Table 4.4 Criteria considered by ITEED	56
Table 4.5 The scale of the relative significance.....	58
Table 4.6 Random index values.....	59
Table 4.7 Description of parameters used in ITUDA	60
Table 5.1 Results of comparison of K-MRCPMB-1.0 with other policies.....	64
Table 5.2 Results of comparison of K-MPCBM-1.5 with other policies	67
Table 5.3 Results of comparison of K-MRCPMB-1.0, K-MPCBM-1.5 with other policies	68
Table 5.4 Normalized decision matrix.....	70
Table 5.5 Relative importance of parameters	71
Table 5.6 Ranks generated according to compliance	71
Table 5.7 Efficiency classification.....	72
Table 5.8 Hosts classification according to efficiency class	74

ABSTRACT

Cloud computing has revolutionized the information technology industry as it delivers scalable, elastic, on-demand and utility based computing features to end users. The cloud data centers use huge amount of electrical energy and deteriorate the atmosphere by releasing harmful gases. For sustainable future, energy-efficient utilization of resources needs sincere attention. Virtual machine consolidation techniques provide an effective solution for decreasing consumption of energy by cloud computing data centers. Although it is an active research area and there are many solutions for achieving energy efficiency, yet existing approaches for virtual machine consolidation ignore the service level agreement violations while reducing energy consumption and are not generic in the sense that they do not work for both CPU intensive and input/output intensive workloads. Moreover, they only consider single criteria related to CPU utilization. The real challenge is to maintain a satisfactory performance level (Quality of Service) without raising energy consumption.

This thesis proposes a multi-criteria based QoS-aware, energy-efficient generic framework for VM consolidation in cloud computing. The study provides introduction to QoS-aware energy-efficient approaches to illustrate the current work done in this field. It also presents the design and implementation of the proposed framework. The present study focuses on all phases of VM consolidation that includes algorithms for host underload detection, host overload detection, VM selection and VM placement. It considers both energy consumption and SLA-aware metrics.

Results show that the proposed algorithms increase the energy efficiency by more than 20% and 75% respectively for CPU intensive and input/output intensive tasks in comparison to the current state-of-art approaches.

Chapter 1

Introduction

Cloud computing is the 21st century vision of computing which simply means accessing and delivering computing services through internet. The first report in this context published in 1960's stated that computing services will be given to users in the form of a utility in the coming future [1]. The term "Cloud" was also referred in different contexts like in description of a large Automated Teller Machine (ATM) network in the 1990's. After Google's chief executive officer Eric Schmidt used the word for describing a business model for providing internet related services in 2006, this term actually gained popularity [2]. The long held dream of delivering computing services as a utility is accomplished by the adoption of cloud computing.

1.1 Background and related terms

The National Institute of Standards and Technology (NIST) has defined cloud computing as "a model that facilitates expedient and dynamic access to a large pool of computing resources that can be shared, dynamically allocated, and discharged without much managerial involvement or service providers' assistance" [3].

Fig. 1.1 illustrates the characteristics [4-5] of cloud computing as described by NIST and explained as follows:

On-demand self-provisioning: Services or resources are dynamically allocated and discharged without human interaction with Cloud Service Providers (CSPs).

Multi-tenancy: Resources are shared among numerous clients/users according to current computing demands. Different virtual and physical resources can be allocated and deallocated according to changing resource demands.

Broad access to network: This feature ensures availability of computing resources and capabilities to users over the broad network. The services can be availed through laptops, computers, mobiles etc.

Pay as you go: The significant advantage of cloud computing is that users do not need to actually purchase the services. The computing services are delivered as utility and users have to pay according to metered usage.

Elasticity characterizes dynamic allocation and de-allocation of resources with respect to dynamic demands for computing.

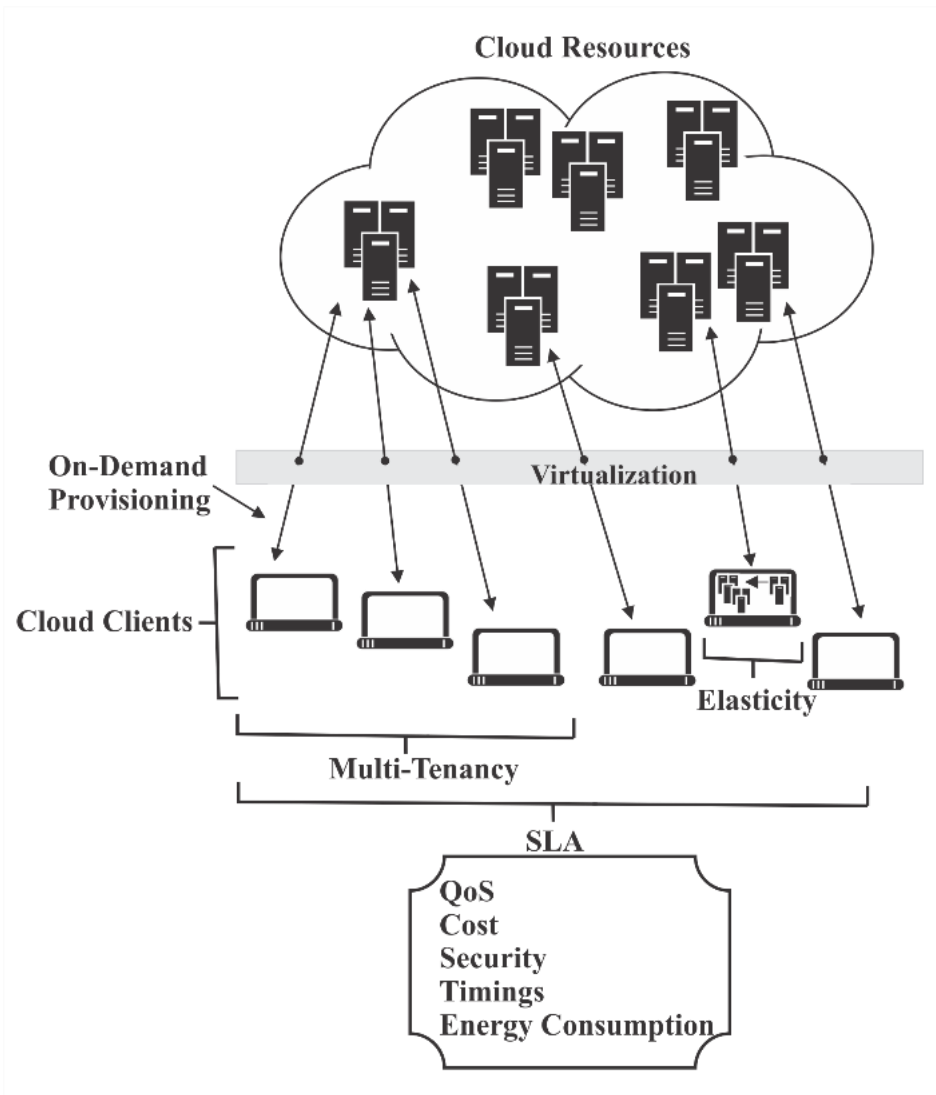


Fig. 1.1 Characteristics of cloud computing

NIST specified some service models [6-7] which are shown in Fig. 1.2.

Software-as-a-Service (SaaS) delivers software in the form of an interface that assists the user in accessing various applications online.

Platform-as-a-Service (PaaS) model allows users to create their own software by providing platform layer resources, software development frameworks, Operating System (OS) support, storage etc.

Infrastructure-as-a-Service (IaaS) model gives support for definite infrastructure which involves storage, networking facilities, computing facilities and additional key resources to consumers [7].

There are certain cloud deployment models also which include public cloud, private cloud, community cloud and hybrid cloud [8].

Public cloud: In public cloud deployment model, the infrastructure or resources are owned by the organisation selling cloud services. The services are made available to public using public network. The public CSPs generally use a common public network to serve numerous clients.

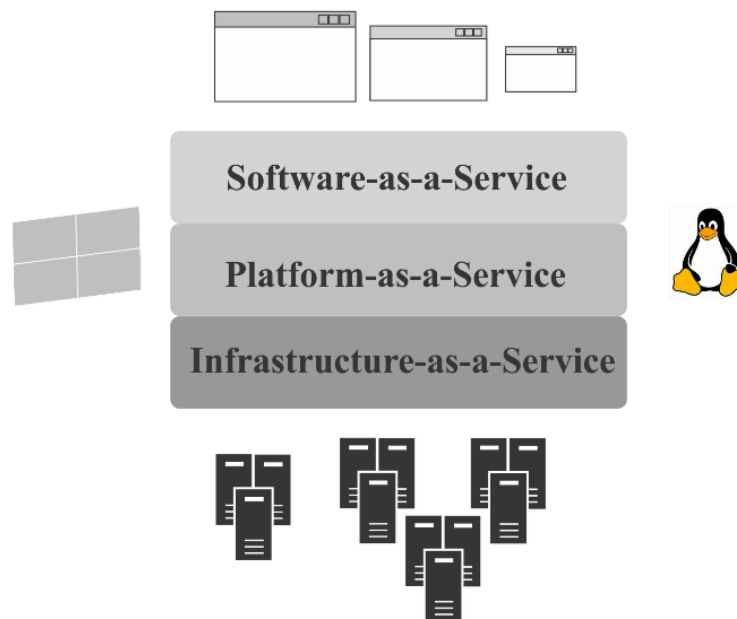


Fig. 1.2 Cloud computing service models

Private cloud: Private clouds are of two types: on-premises cloud or outsourced private clouds. On-premises private clouds are hosted on organizational premises whereas, outsourced clouds are managed by third party. The party gives access to specific organisation to use its services. On-premises clouds are generally managed by organisation itself. Private cloud offers more security in comparison to public cloud.

Community cloud: This type of cloud is for more than one organisation that share same specific concerns. They are shared by multiple organisations. Organisations may themselves manage it or they can be outsourced to a third party as done in the case of private cloud.

Hybrid clouds: These are the grouping of two or more type of clouds (public, private, community). They are bound by specific technologies. Tech giants such as

HP, IBM, Oracle, and VMware endeavour to make use a hybrid environment for proficient provisioning of cloud services [8].

1.1.1 Entities in a cloud environment

The technology behind the success of cloud computing is virtualization. Virtualization permits numerous Virtual Machines (VMs) to run on one Physical Machine (PM). CSPs offer services to its users in accordance with Service Level Agreement (SLA) as depicted in Fig. 1.3. SLA is a signed document between CSPs and users in which they both agree on Quality of Service (QoS) parameters including time, budget, etc. SLA mentions the terms and conditions of service delivery.

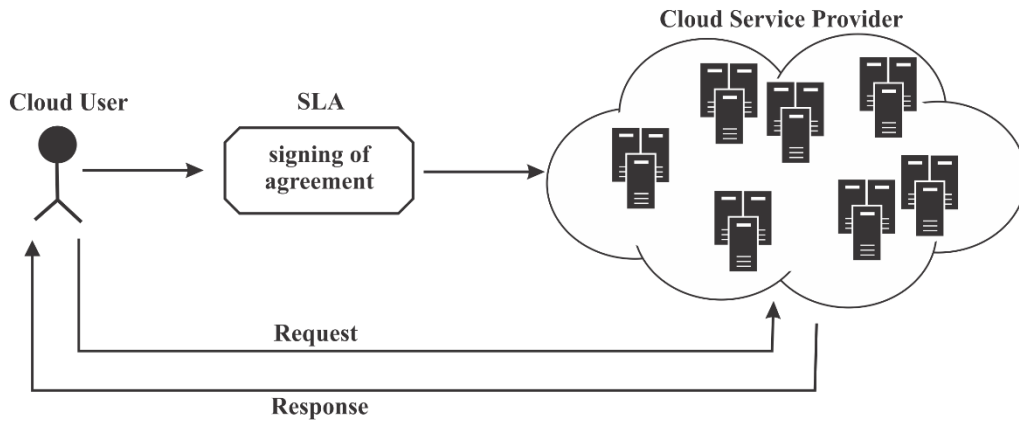


Fig. 1.3 Interaction between cloud user and service provider

The constituents of cloud environment [5] are explained in detail as follows:

Cloud users: Cloud user consumes the services provided by CSP according to SLA that mentions constraints, type etc. of service delivery. Cloud user can be a single client which avails services of a cloud provider by using a smart phone, personal computer, laptop, tablet etc. Cloud user can also be an organization which uses cloud for processing, storage of data and applications. Cloud user generates a request for service which is addressed by CSP in accordance with terms and conditions mentioned in SLA.

Cloud Service Providers: This entity serves the cloud users by providing them the services according to their demand. The services are delivered to users as mentioned in SLA. CSPs own or use one or more data centers which actually provision the services to end users. These data centers are the major sources of power consumption. The renowned CSPs like Google, Amazon, Facebook, etc. have established

their own data centers and are making efforts towards green cloud computing [9]. The main components of a data center are:

Physical Machines or Servers: PMs are main entities of a data center which store, process and transmit data. These are also referred to as hosts. They have their own capacities in terms of storage, computation power etc.

Virtual Machines: With the use of virtualization, the hardware resources can be converted into logical ones. VMs are software computers emulated on actual physical servers. Many VMs can run on same physical device. All the computing resources such as computation capacity, storage of a PM are shared among its VMs. PM uses a hypervisor- Type I or Type II [10] to install VMs on it as shown in the Fig. 1.4. Type I hypervisor is also called bare metal hypervisor as guest OS is directly loaded on it whereas Type II hypervisor is loaded on host OS.

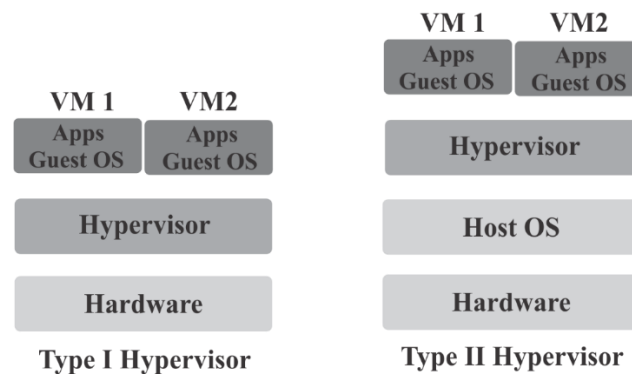


Fig. 1.4 Type I and Type II hypervisors

Networking components: These include devices such as switches, routers, cables, controllers etc. which help in transfer of data and providing connections between the various data center nodes.

Cooling equipment: The data centers release a lot of heat as very powerful servers operate inside them [11]. Thus, cooling equipment such as air conditioners are installed in data centers [12]. Other equipment that help to cool a data center are perforated tiles, raised floors and open aisles etc.

SLA manager: The role of SLA manager is to ensure that services are delivered to users in compliance with terms and conditions mentioned in SLA.

Cloud broker: This component acts as an intermediary between cloud user and CSP. A broker may provide security services such as encryption, certificates of compliance, budgeting information etc. to user. A broker may also be given the

privileges for negotiation of agreements with service provider on account of user. A user interface may be provided by broker to use in order to access the services provided by CSP. Fig. 1.5 shows the high level communication between the cloud computing entities.

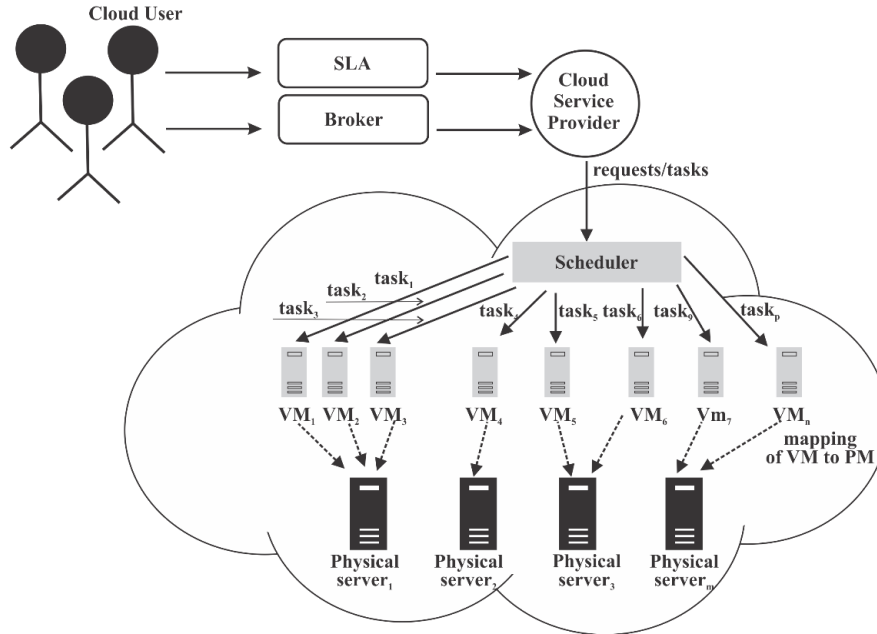


Fig. 1.5 High level communication between cloud computing entities

All the requests are submitted by user to CSP through broker. There is a two level mapping of resources. VMs in data centers are allocated to hosts. This is called VM allocation to hosts. The tasks are mapped to VMs for completion of the work assigned. This is termed as task scheduling or mapping. The scheduling is done in accordance with the constraints in SLA like time, cost etc.

1.2 Research issues in cloud computing

Though cloud computing is broadly accepted by the industry, but the research on it is still at its infancy stage. Various current problems are not completely addressed, whereas new challenges keep on rising from industry applications. Issues in cloud computing are related to security, load balancing, trust management, traffic management, maintaining QoS, energy consumption etc.

Security: Security of data is a critical issue in cloud computing [13-15]. Cloud users do not have physical control over data as it is managed by CSPs. Mechanisms

such as cryptography, security protocols and algorithms are needed for data transfer. A security mechanism must ensure high reliability ensuring authentication, authorization and availability of data.

Load balancing: Servers hosted in cloud computing data centers face the problem of load unbalancing [16-18]. Some of the servers are over-utilized while others sit idle for most of the duration. Load unbalancing leads to high energy consumption. CSPs must ensure automated provisioning of services by dynamically allocating and deallocating the resources according to need. The development of VM consolidation algorithms composed of migration techniques that can manage the trade-off between resource utilization and QoS is the need of the hour.

Trust management: One of the vital issues faced by CSPs is trust management [19-21]. Cloud clients avail services from trustworthy CSPs according to their level of reputation in the market. Deciding trustworthiness of CSPs is a critical challenge as trust is a subjective concept.

Traffic management: In order to have better client experiences, analysis and management of data traffic is very crucial. The planning and management of decisions must be done by analysing the flow of traffic. The strategies for traffic management [22, 23] must calculate traffic metric for thousands of nodes, effectively handle coupling between networking components, analyse link density between nodes.

Maintaining QoS: QoS is of paramount importance to user [24, 25]. A cloud client selects a CSP based on the its level of fulfilling SLA. SLA includes all the constraints related to delivery of services like budget, deadline, makespan etc. A crucial challenge is optimal utilization of resources while maintaining SLA. Scheduling techniques must ensure QoS in addition to load balancing, traffic management etc.

Energy consumption: Energy consumption by information technology industries has accelerated at a very fast rate due to advancements taking place in digital world. This problem has reached to its peak after the revolution of “computer” in 20th century [26]. With the discovery of internet, computers occupied whole globe and information technology industry matured, infrastructural requirements switched, resulting in construction of internet data centers. The power sector reported a great hike in electricity consumption [27]. International Energy Outlook has estimated that the power consumption across the world will increase by 56% from 2010 to

2040 and the main cause will be information technology industries [28]. Reducing consumption of energy by cloud computing data centers and improving energy efficiency is the prime goal of this study. The proposed solutions should not aggressively reduce energy consumption but also maintain QoS. This needs further research efforts [26].

1.3 Energy consumption by cloud data centers

Cloud environment provides scalability, virtualization, networking facilities to users who don't have to own systems and manage them [29]. Customers avail these infrastructural services, platforms and software applications given by service providers [30]. However, the servers housed in cloud data centers release a lot of heat. Thus, they need to be kept in an air conditioned environment leading to consumption of more power due to use of information technology equipment.

Cloud computing changed the way of offering computing services. Though it brought accepting repercussions in information technology industry, but it can't be considered a highly productive technology [31]. The current emission of greenhouse gases by data centers is 2% which will turn out to be 16% in the coming future. The earth's temperature is estimated to rise expansively by 2047 resulting in unfit living conditions [32]. Therefore, cloud computing may not be highly productive, but energy-efficient cloud computing can prove to be lucrative.

The revolution brought by information technology industry is due to agile and swift delivery of services by cloud computing data centers. However, the energy consumption by these data centers has reported a tremendous hike. They consume 3% of global electrical energy which leads to high operational costs and thus increased total cost of ownership. These data centers also degrade the environment by harmful carbon emissions.

The unusual fact is that resources in idle state in a data center too consume high amount of power at leisure [11]. Servers housed in cloud data centers are extremely away from energy uniformity. Even at 20% utilization, they draw 80% peak power. This is a major cause of energy inefficiency. The servers sit idle frequently as their utilization is between 10%-50% of their peak load [33].

The major concern is inefficient utilization of resources which causes high power consumption. The over-utilized resources degrade the performance and under-utilized resources consume high amount of power even when not in use. The real task is to maintain an acceptable performance level without increasing the energy costs [34]. The scheduling strategy should take into account optimal utilization of resources and load balancing considering QoS (SLA) constraints. The idle servers should be turned off when not in use. So, there is a need of developing an energy-aware framework which considers QoS constraints simultaneously for optimum resource utilization.

1.4 Energy-efficient techniques

The energy efficiency techniques in cloud data centers comprise of evolution of forefront technologies which further includes usage of equipment utilizing lesser voltage, green computing devices [35, 36]. Moreover, the innovative technologies also incorporate the design of hardware and software-based power-aware algorithms e.g. Berkeley project [37]. There are various more appeals [38, 39] that endorse improving energy to performance ratio, using advanced cooling equipment, replacing nonrenewable energy generation sources with inexhaustible ones as done by popular technology giants- Yahoo and Google [40]. Finally, other proposals target on the development of energy-efficient job scheduling strategies [41, 42]. However, decreasing power consumption and maintaining performance is a bigger challenge [43] and in-depth assessment of the profits on performance and energy savings achieved from a scheduling strategy often breaks into the trade-off among these two goals. The real task is to maintain an acceptable performance level without increasing the energy costs [44]. The infrastructural [45-47], hardware [48-50] and software-based solutions [26] for reducing energy consumption are shown in Fig. 1.6.

1.4.1 Infrastructure-based solutions

These involve building data centers in locations near to water resources away from living habitat, using energy-efficient equipment [45], including energy star ratings

(information technology and non-information technology equipment) (green buildings) [46]. Positioning of air conditioned racks, perforated tiles, floor raising etc. as shown in Fig. 1.7 also help to minimize energy consumption (cooling infrastructure or thermal-aware techniques).

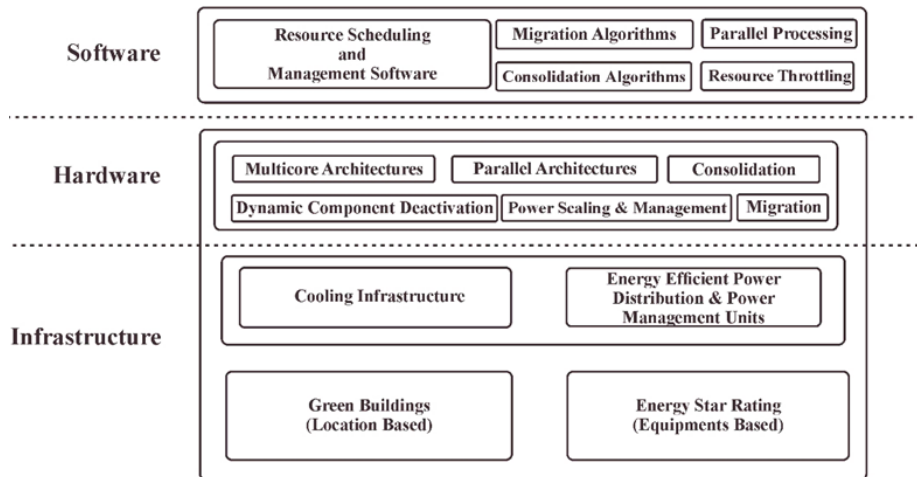


Fig. 1.6 Solutions for reduction of energy consumption

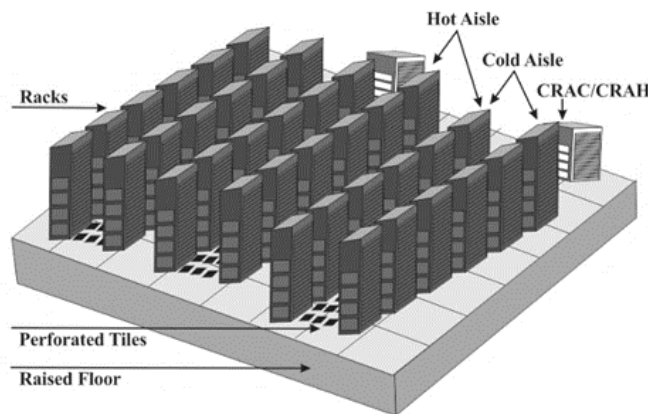


Fig. 1.7 Infrastructure-based optimizations

Energy-efficient power distribution and management of units also come under infrastructure-based solutions. But these techniques resulted in large expenditures [47] and even after infrastructural changes, energy consumption did not reduce much. So, various hardware-based optimizations were done.

1.4.2 Hardware-based solutions

Hardware-based solutions employ multi-core architectures, parallel architectures (architectural changes). The facilitation of numerous processing cores on a single

chip escalates efficacy by processing various tasks simultaneously. Various processing cores accelerates the performance of the tasks by dividing workload among different cores rather than making use of a single-core processor [48]. In contrast to single-core processors, multi-core processors show improved performance maintaining same cooling load and power [49].

They also include usage of energy-efficient hardware components, network devices (hardware replacement) and power management and scaling methods (dynamic performance management).

They involve various practices that can be applied to computer components resulting in dynamic change of performance in accordance with power consumption. Major techniques used are Dynamic Voltage Frequency Scaling (DVFS) and DVFS with slack reclamation. DVFS decreases consumption of power by increase or decrease of supplied frequency as per need. If a processor executes at a lesser frequency, low voltage will be needed consuming lesser power. Thus, frequency and voltage are adjusted dynamically for decreasing the power consumption. Processor running at lesser frequency needs more time to execute an instruction. Processor's governor screens this procedure for maintaining performance. Usually, processor initiates at lesser frequencies and gradually rises with workloads, so is the power consumption. Fig. 1.8 shows architecture of a system using DVFS.

In order to complete tasks in a certain deadline, slack reclamation technique is used simultaneously with DVFS. In concurrent processing, various tasks execute in parallel. Moreover, if complete job execution is dependent upon two preceding jobs and these two jobs complete their execution at different timings, previous job which finishes first can use extra execution time known as slack. This extra time can be utilized by DVFS for maintaining energy efficiency [50].

However, hardware and infrastructure-based solutions alone are not self-sufficient without optimal resource usage [11]. So, consumption of energy does not only depend on efficiency of physical servers but also on strategies for management of resources. Thus, software-based solutions were deployed.

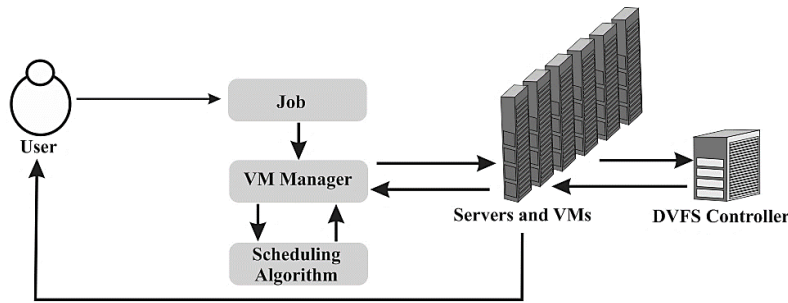


Fig. 1.8 Architecture of a DVFS-based system

1.4.3 Software-based solutions

Software-based solutions involve resource management strategies (allocation and scheduling) [26]. High Performance Computing (HPC) systems demand effective resource provisioning and scheduling strategies. The scheduling of received requests and allocation of resources is a vital task. The resources include networking components, storage devices, CPU, etc. The scheduling strategy should focus on maintaining performance goals as well as power, budget, timing constraints. The efficient scheduling of resources helps in optimum resource utilization and initiates efficient program execution. The optimal utilization of resources helps in lowering down the power demands.

The energy consumption is closely related to level of resource utilization. It is well clear from the fact that under-loaded or over-loaded resources consume more power. Thus, the optimal infrastructure usage improves energy efficiency.

Software-based solutions involve virtualization driven consolidation and migration algorithms. VM migrations reduce the number of active PMs by detecting over-loaded and under-loaded machines. Migrations allow to achieve workload balancing among machines. In reality, workload balancing aids to overcome situation where some machines operate at heavy loads and other remain idle, leisurely consume power, or perform little work.

Besides performing the VM migrations, cloud computing implements VM consolidation depending upon the present resource requirements of VMs.

Consolidation means merging into one, which helps to decrease energy consumed due to switching-off idle machines and dividing the work among less number of machines [51].

Cloud computing supported by virtualization performs consolidation at various levels- VM, task and server consolidation. The under-loaded servers are consolidated to a central location to ensure efficiency of a data center. Server consolidation helps to escalate the resource utilization and cut the increasing levels of energy consumption [52].

Task consolidation on the other hand, puts tasks with same necessities on a single resource. It is possible due to virtualization technology supported by cloud computing which permits various tasks to run simultaneously on a single machine. The tasks that have same requirements are allocated to same servers in order to prevent wastage of energy by execution of same type of tasks and avoiding overheads and complexities. Parallel processing of workload also comes under software-based solutions. The dual implementation of multi-core architectures and virtualization techniques facilitates parallel processing and offers energy-efficient computing [53]. The VMs can be isolated for concurrent execution on a single processor composed of multiple cores, without causing any interference among the VMs [54].

The prime focus of this study is VM consolidation. VM consolidation is an effective approach which leads to optimal utilization of resources [55-57]. VM consolidation involves shifting the VMs from less utilized servers and over-utilized servers to normally loaded servers, So, less utilized servers may be switched off to save power.

1.5 Process of VM consolidation

VM consolidation results in increased number of hosts shutdown through live migration of VMs. The VMs deployed on the under-loaded hosts are consolidated onto single PM to prohibit other machines from consuming excessive power. Resources can be consolidated onto lesser number of machines by turning off under-utilized machines or by migrating the resources from over-utilized machines to manage energy consumption. Fig. 1.9 presents the process of VM consolidation.

VM consolidation involves (i) Detection of over-loaded hosts (ii) Detection of under-loaded hosts (iii) VM Selection (iv) VM Placement.

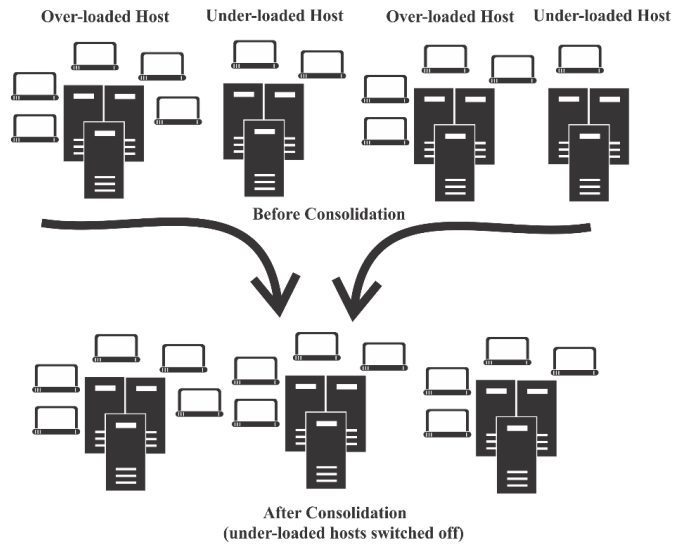


Fig. 1.9 VM consolidation process

Fig. 1.10 represents scenario depicting the VM consolidation. Let H_1, H_2, H_3, H_4 and H_5 be the hosts in a data center. $VM_1H_1, VM_2H_1, VM_3H_1, VM_4H_1$ are the VMs on H_1 . H_2 has two VMs- VM_1H_2, VM_2H_2 . H_3 accommodates five VMs- $VM_1H_3, VM_2H_3, VM_3H_3, VM_4H_3, VM_5H_3$. H_4 has only one VM- VM_1H_4 and H_5 has two VMs- VM_1H_5 and VM_2H_5 . Suppose threshold value is 4 (ideal) means a host having less than 4 VMs is considered to be under-loaded and a host having more than 4 VMs is over-loaded. So, H_2, H_4, H_5 are under-loaded hosts marked with UH and H_3 is over-loaded host depicted by OH. After that VM selection algorithms are applied to select VMs to be migrated. Let VM_5H_3 and VM_1H_4 be the selected VMs. Now using VM placement algorithms these VMs are deployed on new hosts. Let VM_1H_4 and VM_5H_3 are placed on H_2 as it was under-loaded. Thus, H_4 can be switched off. This results in lesser power consumption.

1.5.1 Over-loaded hosts detection algorithms

The techniques found in literature for detecting over-loaded hosts are: (i) Adaptive host overload detection algorithms (ii) Regression-based host overload detection algorithms (iii) Static threshold-based algorithms.

Adaptive algorithms are robust and rely on statistical analysis of data of resource utilization by VMs to find over-loaded hosts. The standard adaptive algorithms include Median Absolute Deviation (MAD) and Interquartile Range (IQR) algorithms [58]. These algorithms calculate the threshold values and values may change

according to dynamic workload. The hosts having utilization more than the calculated upper threshold are considered to be over-loaded and hosts having utilization lesser than the calculated lower threshold are under-loaded.

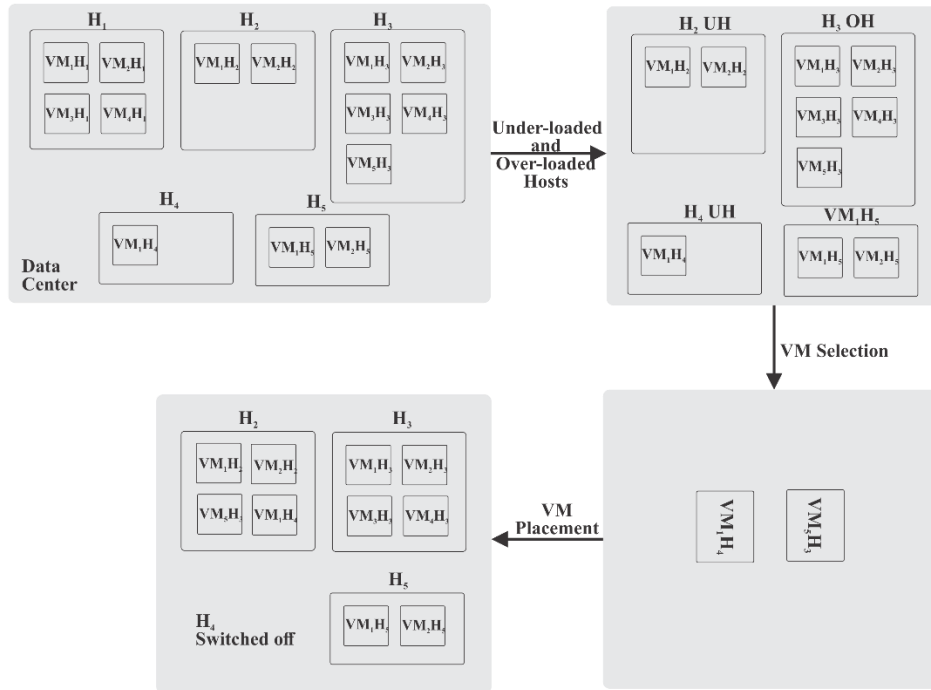


Fig. 1.10 Scenario representing VM consolidation

Regression-based algorithms are Local Regression (LR) and Local Regression Robust (LRR) algorithms. They are not based on threshold value but predict the future CPU utilization from past data. These algorithms give better estimation of over-loaded hosts by using Loess method [58].

Fixed threshold-based algorithms use static threshold value to calculate utilization thresholds. It detects whether the host is over-loaded or not. The simplest approach is using average threshold-based algorithm (THR) [58]. If the average of last n CPU utilizations is greater than specific threshold, then host is over-loaded. The disadvantage of fixed threshold is that it is not suitable for the environment in which dynamic workload is there.

1.5.2 VM selection algorithms

These are used to choose VM which is to be shifted from over-loaded/under-loaded host. Minimum Migration Time (MMT), Random Selection (RS), Maximum Correlation (MC) and Minimum Utilization (MU) policies are the standard algorithms

proposed by Beloglazov [58] for choosing the VMs that are to be shifted. MMT policy migrates VM with least migration time (calculated by dividing RAM by spare network bandwidth) in comparison to other VMs accommodated on the same PM. RS policy chooses any random VM and shifts that.

MC policy works on the idea that if there is a higher correlation of resource utilization by the applications on a server, then the probability of server overloading is also higher. Thus this policy selects those VMs that have higher correlation of CPU utilization than other VMs. So, coefficient of multiple correlation is calculated to select the VMs to be migrated. MU policy works same as static threshold-based algorithms use static value to calculate utilization thresholds to detect whether the host is over-loaded or not. If the current CPU utilization of a host is more than the fixed threshold, then host is considered to be over-loaded. The limitation with these policies is that they do not consider multiple resources during migration and are based on single criterion.

1.5.3 VM placement algorithms

VMs from under-loaded/over-loaded hosts are to be shifted to new target hosts. These algorithms deploy VMs on new hosts based on the scheduling criteria. Algorithm used by [58] for selection of hosts is Power-Aware Best Fit Decreasing (PABFD). VMs are arranged in descending order of CPU utilizations and allocation of VM to that host is done which causes least increase in power consumption. It considers single criterion only.

1.5.4 Host underload detection algorithm

These algorithms detect the hosts having lower utilization levels. The algorithm proposed by [58] is very simple. Host with MU is found using a threshold. VM consolidation allows placing VMs from under-loaded hosts to other hosts while not overloading them and the source host can be switched off. The process is repeated again and again for all hosts.

The algorithm explained here for different phases of VM consolidation are default algorithms in CloudSim 3.0.3. These algorithms have become the base for devel-

opment of other VM consolidation algorithms. Several algorithms found in literature are modified versions of these basic algorithms only. However, majority of the algorithms are based on single criterion and they only consider reducing energy consumption and do not focus on lowering SLA violations.

1.6 Energy performance trade-off

Delivering reliable QoS is one of the main goals of CSP. QoS can be explained in terms of SLAs. Though new virtualization techniques can guarantee performance isolation among VMs hosted on one computing node, but dynamic, heterogeneous workloads and aggressive consolidation lead to resource unavailability to VMs at the time they make a request for them. This may cause a loss of performance in terms of more response time, delays or failures in the worst case. So, CSPs have to focus on two competing goals of maintaining performance and reducing energy consumption. The key challenges that remain to be addressed are:

- How to design a generic framework for energy-efficient cloud?
- How to ensure optimal resource utilization?
- How to optimally solve the trade-off that occurs between energy savings and delivered performance?
- What parameters are to be focused upon for developing QoS-based energy-efficient framework?

This work focuses on development of QoS-aware energy-efficient framework for cloud data centers. As cloud computing is a dynamic technology, the proposed framework should also be able work with heterogeneous workloads. This study is an effort towards designing of multi-criteria based QoS-aware energy-efficient framework that works for both CPU intensive and Input/Output (I/O) intensive tasks.

Other deliverables that would be produced as a part of this research work will include the analysis documents, design documents and information related to energy-efficient techniques in cloud computing. Analysis documents will represent the study, critical analysis and comparison of different energy efficiency techniques

applicable to cloud computing. Design documents will include the different components of energy-efficient framework, their interconnection and working mechanism.

1.7 Organization of the thesis

The thesis is organized as follows:

Chapter 2 presents the related work done in the field of energy efficiency in cloud computing. Detailed analysis of models developed for reducing energy consumption is presented. It represents the key VM consolidation techniques to achieve energy efficiency in cloud computing. The datasets and simulation toolkits used by researchers are mentioned in this Chapter. Research gaps existing in the literature are highlighted. This Chapter also presents the objectives of the current work.

Chapter 3 presents design of QoS-aware energy-efficient VM consolidation framework. This Chapter illustrates the design of framework using the novel algorithms for detection of under-loaded/over-loaded hosts, VM selection and VM placement. It takes into account the dynamic nature of cloud computing. The working of the framework is also described in this Chapter.

Chapter 4 describes the algorithms developed for all the phases of VM consolidation, the experimental settings and the methodology adopted to implement these algorithms. It explains the configuration of hosts and VMs used in the experiment. The description of dataset used is also provided. This Chapter also throws light on the energy and SLA related performance metrics used in the study.

Chapter 5 presents the results of implementation of the framework. Different scenarios identified for demonstrating usability and applicability of the proposed framework like testing the designed framework for real and synthetic datasets, multi-core CPUs etc. The comparison of results with previous frameworks is made in this Chapter through graphical interpretations.

Chapter 6 provides the conclusion of the study and future research directions.

Chapter 2

Literature Review

Previous literature lays a foundation to formulate the objective of research. The existing literature shows the state of art technologies used for achieving energy efficiency in cloud computing.

In early stages, work on energy-aware resource management was carried on for mobile devices to improve battery lifetime [59, 60]. Later on, the focus has been changed to data centers [61, 62] and virtualized cloud environment.

VM consolidation is of paramount importance in providing energy-efficient solutions for cloud computing. A lot of work is found in literature on development of algorithms for detecting under-loaded hosts, over-loaded hosts, VM selection for migration and VM placement. Apart from this, literature shows considerable amount of work in development of energy-efficient framework using resource management and scheduling algorithms. The studies on consolidation techniques can be classified based on (i) datasets (ii) type of consolidation (iii) type of threshold (iv) criteria considered as shown in Fig. 2.1. The following literature explains the work done on energy-efficient resource management based on these classifications.

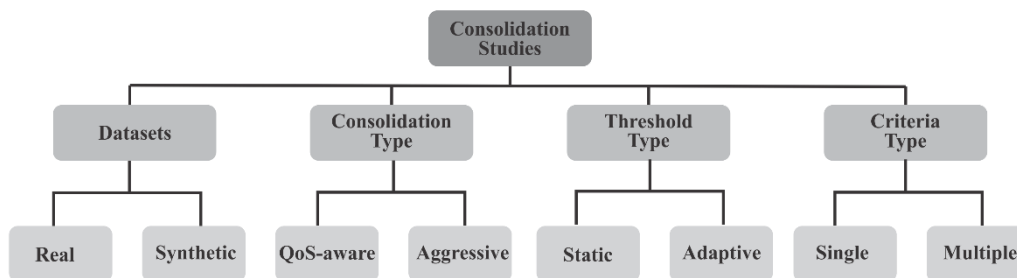


Fig. 2.1 Classification of work done related to consolidation

Srikantaiah et al. (2008) [63] formulated consolidation as a multi-dimensional bin packing problem. The authors considered only two criteria- CPU and disc utilization. The study revealed that there exists a trade-off between energy and performance for consolidation and disclosed the existence of optimal operating points. They set up a cloud environment, collected their own data and formulated a bin packing problem using static random threshold values.

Gang Zeng et al. (2009) [64] performed energy-efficient periodic real-time task scheduling on Dynamic Voltage Scaling (DVS) enabled systems with multiple processors following certain constraints. The constraints included power consumption at idle state, platform heterogeneity, heterogeneous applications etc. Synthetic multi-processors with homogeneous architecture were used. Authors proposed an Adaptive Minimal Bound First-Fit (AMBFF) algorithm for multi-processor dynamic and fixed-priority systems. Random data was used on environment comprising of synthetic multi-processors with homogeneous architecture. Results indicated that designed algorithm could achieve average 16.1%, 17.2%, and 18.1% energy reduction w.r.t. worst fit decreasing method for benchmark models- XScale, PowerPC and DSP, respectively. Authors ignored constraints associated with parallel DVFS applications which must be addressed for real time applications.

Buyya et al. (2010) [65] proposed energy-aware resource allocation and scheduling algorithms considering QoS constraints. The authors divided VM consolidation problem into two parts. First part allocates the resources to target machines and in the next part optimization of allocation is done. Four heuristics- “Single Threshold (ST), Minimization of Migrations (MM), Highest Potential Growth (HPG) and Random Choice Policy (RCP)” were presented for detecting over-loaded, under-loaded hosts and selection of VMs for migration. The threshold values were static. Modified Best Fit Decreasing (MBFD) algorithm was used for placement of VM. CloudSim toolkit was used for validating the proposed framework. The dataset of CPU utilization was generated randomly. Results showed the effectiveness of this work.

Beloglazov and Buyya (2010) [66] used ST, MM algorithms and bin packing heuristics for VM consolidation. The evaluation was done using random data. Authors tried to optimally deal with the trade-off between energy savings and desired performance. They performed VMs consolidation according to current resource utilization, network topologies used in VMs and thermal state. An energy-aware resource scheduling system was proposed which used heuristics for allocation of VMs and live migration. Authors formulated it in the form of bin packing problem and

used CloudSim toolkit for evaluation of the work using fixed thresholds. The designed algorithm could efficiently manage hard QoS requests, heterogeneous architecture. The developed procedures were workload and application independent. The algorithms needed no information about type of applications running on VMs. The results illustrated that dynamic VM consolidation using adaptive thresholds is much better than making use of static thresholds in terms of energy savings. The MM algorithm was compared with ST, Non Power Aware (NPA) and DVFS schemes. MM algorithm performed better in terms of energy savings in comparison to NPA, DVFS and ST by 83%, 66% and 23% respectively with thresholds kept as 30-70% causing SLA violations of 1.1%. The MM policy yielded 6.7 % of SLA violations, 87%, 74% and 43% more energy savings in contrast to NPA, DVFS and ST policies when threshold value was kept as 50-90%.

Youness et al. (2010) [67] proposed a procedure for designing multiple cores system by making best use of scheduling strategy resulting in optimal number of cores, buses, partitioning of hardware and software from task precedence graphs. Hard and large graph based problem scenarios were used to test its optimality. Benchmark programs were used for data collection. Geometric algorithm, A-star best fit state space search, algorithms for partitioning and pruning were used which resulted into savings of significant number of cores for fault tolerance and reliability.

Beloglazov et al. (2011) [46] presented heuristics for energy-efficient allocation of resources. The policy provisioned resources to consumer applications in an energy-efficient way guaranteeing QoS making use of energy-efficient mapping heuristics. VM consolidation was performed. For VM placement, MBFD algorithm was used and three double-threshold VM selection policies MM, HPG, RCP were used. CPU utilization values were randomly generated. Fixed thresholds were used. CloudSim toolkit was used to evaluate the work. Results made it clear that the consumption of power decreased by 77% and 53% in contrast to policies -NPA and DVFS respectively with SLA violations of 5.4%.

Beloglazov and Buyya (2011) [68] proposed a dynamic VM consolidation approach based on the fact that fixed thresholds do not fit the dynamic cloud environment. The authors presented dynamic thresholds value by statistically analyzing

CPU utilization history. The reallocation was done using dynamic threshold algorithm. MBFD algorithm was used for placement of VMs. The SLA-aware metrics were also considered. Results of execution of algorithm on CloudSim toolkit using real Planet Lab trace indicated the worthiness of proposed framework. However, only single-core CPUs were used in the proposed model and only single resource-CPU was considered.

Lin et al. (2011) [69] proposed two algorithms for VM consolidation and scheduling named as Dynamic Round Robin (DRR) and hybrid algorithm which was a combination of First Fit (FF) and DRR algorithms. Many experiments were conducted under varied experimental settings using two sets of collected data. Only CPU was considered in the model. Experiments were conducted on PMs having Xen hypervisor and simulations were also performed using different values of utilization thresholds. Experimental results proved that DRR and hybrid algorithm contributed to 56.4% and 55.9% less energy consumption respectively in comparison with Round Robin (RR) algorithm in Eucalyptus.

Lee et al. (2011) [70] performed Energy Conscious Scheduling (ECS) for distributed systems under varying operating conditions. The energy saving of ECS and ECS+ idle was enabled by making use of DVS technique. Data used was from real world applications. Statistical analysis was made to indicate the importance of the work. Algorithms used were heuristic task prioritization methods, Heterogeneous Earliest Finish Time (HEFT), makespan conservative energy reduction technique, DVS. Limitation of this work was that clock frequency transition overheads were ignored.

Feller et al. (2011) [71] considered consolidation as a multi-dimensional bin packing problem and developed VM placement procedure motivated from nature inspired “Ant Colony Optimization” (ACO) technique. The algorithm computed placement dynamically based on current load. Authors designed their own simulator to conduct experiments measuring values from the created test bed. Results showed efficiency of proposed heuristic in terms of energy consumption as compared to greedy approaches. However, authors assumed all physical servers to be homogeneous having similar capacity.

Jeyarani et al. (2012) [72] presented an “Adaptive Power-aware Virtual Machine Provisioner” (APA-VMP) which provided least energy consumption while maintaining performance. Randomly generated data was used for implementation. Self-aware Particle Swarm Optimization (SAPSO) technique was executed on CloudSim and it aimed to achieve significant power reduction. Future work included designing the system considering modern data centers that conserve energy by making use of adaptive provisioning techniques and neural networks.

Lee and Zomaya (2012) [50] proposed energy-aware consolidation heuristics which considered both energy consumption in the idle and active state, maximizing utilization of resources. The designed heuristics performed mapping of tasks to resources without degrading the performance. Simulation results of the experiments conducted using randomly generated tasks and variable resource utilizations proved the effectiveness of proposed framework.

Goiri et al. (2012) [73] designed a scheduling algorithm for managing a virtualized data center. The designed policy mapped VMs to nodes of data centers considering multiple parameters such as SLA efficiency, energy efficiency and overheads. Power-aware simulator was used to evaluate the proposed algorithm using heterogeneous workloads considering SLA and energy related metrics. The provider’s profit was increased by 30% using this policy. Limitation of the work was that it could not support all applications with varying SLAs. The designed policy could also be merged with cost-benefit model which would have resulted in decrease of number of nodes and energy consumption.

Cao and Dong (2012) [74] presented algorithms for dynamically consolidating VMs to reduce energy consumption and SLA violations in virtualized data center. Authors proposed host overload detection, VM selection and allocation policy. To detect over-loaded hosts, authors used standard deviation and mean of CPU utilization values. To select VMs for migration, MCC policy was used. For VM allocation, mean and variance related computations were performed. Results of experiments conducted on CloudSim using Planet Lab traces showed that the proposed framework composed of above policies outperforms the previous policies in terms of consumption of energy and maintaining QoS as a whole. However, in terms of only

consumption of energy its performance was a little worse. Thus, the challenge is to manage energy-performance trade-off.

Beloglazov and Buyya (2012) [75] proposed a host overload detection algorithm as overutilization of resources leads to performance degradation. The designed algorithm used Markov chain model for maximizing the mean time used for migration under the defined QoS level. The algorithm was adapted for non-stationary workloads. It also performed workload estimation employing multi-size sliding window technique. Planet Lab data was used to evaluate the proposed policy on CloudSim toolkit which showed more optimal results as compared to best benchmark algorithm and 88% of the performance of the optimal offline algorithm. However, authors focused on one phase of VM consolidation i.e. host overload detection.

Jeong et al. (2013) [76] used VM migration as a power capping method. The parameters that were focused upon included energy consumption, overheads related to performance, time of execution, utilization of CPU. The proposed policy simultaneously estimated, controlled consumption of power while maintaining performance to fit power capping mechanisms. Authors showed that live migration approaches added more power consumption to originating node or cause damage to performance of remaining VMs. Two workarounds were introduced for prevailing precopy live-migration schemes based on limited scheduling and DVFS that could quickly reduce the power consumed and also the effect on remaining VMs performance. SPEC CPU 2000 benchmark suite was used to perform experiments on a system using Linux OS and Xen hypervisor for the x86 configuration. The use of DVFS significantly reduced power consumption but presented policy was not suitable for reducing controlled energy consumption.

Farahnakian et al. (2013) [77] tried to decrease consumption of energy by reducing the number of active physical servers in a virtualized data center through dynamic VM consolidation. The authors focused on determining over-loaded and under-loaded hosts by proposing regression based K-nearest neighbour algorithm to predict resource usage. Two types of workloads: real and synthetic were used to

evaluate the performance of algorithms using CloudSim toolkit which indicated its efficiency.

Gao et al. (2013) [78] focused on VM placement problem by designing a multi-objective ant colony algorithm. The aim was to generate pareto solutions that simultaneously considered minimization of both energy consumption and wastage of resources. Random problem instances comprising of demand of CPU memory utilization for 200 VMs were generated. The proposed algorithm was compared with multi-objective genetic algorithm, bin packing algorithm and a max–min ant system algorithm and its performance was better in comparison to these. However, all servers used in experiments were considered to be homogeneous.

Wu et al. (2014) [79] utilized DVFS based scheduling policy for virtualized cloud data center. The proposed scheduling policy could proficiently escalate utilization of resources decreasing the power consumed for running jobs. The supplied voltage and frequency for servers was managed dynamically by DVFS technique. This policy proved beneficial in reducing power consumption of a server in idle state or under light load. The simulation conducted on CloudSim showed that the proposed method reduced the energy consumption of the minimum and maximum frequency-DVFS policy by 5%–25%.

Cao and Dong (2014) [55] redesigned VM consolidation framework proposed by authors in [46]. To obtain a better QoS, SLA Violation Decision Algorithm (SLAVDA) was proposed which identified whether a host is over-utilized with SLA violation or not. The execution of framework was done using Planet Lab workload traces on CloudSim toolkit. Results proved the worthiness of the proposed framework as it reduced energy consumption, SLA violation and energy-performance metric by 11.8%~27%, 57.9%~78.4% and 63.2%~84.1% respectively in comparison to [46].

Horri et al. (2014) [56] proposed a QoS-aware algorithm for consolidation of VMs based on the level of utilization of resources by VMs. CloudSim toolkit was used for implementing and evaluating the proposed algorithms. Obtained results made it

clear that there exists a trade-off between energy consumption and QoS in the virtualized environment offered by cloud computing. An improvement in consumption of energy and QoS metrics was also observed. The results showed 50% decrease in number of VMs migrations in contrast to LR/MMT/ Simple method/ PABFD algorithm with 70% reduction in SLA violation.

Szynkiewicz et al. (2014) [80] focused on dynamically managing power in computing network and data intensive systems. Authors concentrated on reduction of power, QoS parameters and traffic management. The framework was designed for centralized and hierarchal settings for a network consuming lower energy. Implementation of two control levels was done, central unit to implement network wide control strategies and local unit to implement local control mechanisms. Simulations were done using randomly generated real time tasks and authors developed parametric simulator to imitate adaptive real time systems. State space search, branch and bound, dynamic programming, DVS techniques were used for getting desired utility gain and run time. Designed model did not consider utilization of resources like memory, communication networks, disk etc.

Netjinda et al. (2014) [31] focused on execution time, makespan, total cost to optimize IaaS purchasing expenditure for achieving execution of scientific workflow within particular deadlines. It helped cloud clients by making decisions easier. Authors made use of Particle Swarm Optimization (PSO) technique considering parameters like price, number of machines etc. for minimization of net cost. Four types of scientific workflows were used for experimental purpose. Amazon EC2 instances were used for cloud set up. The proposed work did not consider the overheads during communication and the case where execution time of task is not known in advance.

Somasundaram et al. (2014) [81] presented “CLOUD Resource Broker” (CLOUDRB) framework for efficiently managing resources in cloud data centers. Real HPC applications were used to test on MATLAB and Eucalyptus based cloud environment. The merged deadline-based job scheduling and PSO-based resource scheduling mechanism prioritized application requests of cloud clients considering timings, cost constraints. Results indicated that PSO performed 1.23 times better

than ACO, 1.3 times better than Genetic Algorithm (GA) and 1.77 times better than rank based allocation in terms of number of jobs fulfilling deadline constraints. Future work included mixing of semantic resource detection and SLA to improvise job scheduling procedure and delivered QoS.

Kumar et al. (2014) [82] proposed migration and communication energy-efficient scheduling of tasks for consistent system having multiple cores. An offline task remapping technique, a “Mixed Integer Quadratic Programming” (MIQP), “energy-gradient based heuristic” was used to reduce timings of exploration of design space by 99% with highest deviation of 5% from ideal solution got by tackling the MIQP problem directly. Results using synthetic & real data on a Linux system using Intel Xeon 2.4 GHz server, C++, MATLAB showed that communication energy was reduced by 35% on an average and it also reduced migration overhead by 20% on an average considering single and double fault cases. The presented algorithm delivered 50% high throughput per unit energy in contrast to existing approaches. Authors did not consider energy consumption for task computation and reduction of mapping storage overhead.

Hosseinimotlagh et al. (2014) [83] designed “smart energy-aware task scheduling” algorithm in real time cloud computing environment based upon better utilization level with ideal power consumption as well as guaranteeing delivery of QoS. It regulated VMs computing resources mapped to a host leading to discovery of ideal energy level in the host. The results of simulation in CloudSim showed that the presented technique not only reduced overall consumed energy of a cloud by 60%, but also reduced turnaround timings of real tasks by 94%. It further increased the task acceptance rate by 96%.

Corradi et al. (2014) [84] focused on VM consolidation from a more practical point of view by considering aggregated resource consumption of collocated VMs to maintain QoS. It is found that interference among co-located VMs must be considered to ensure SLA guarantees. For experiments, cloud environment available in small industries was replicated by using personal computers having distinct physical server’s configuration. Open stack release -Diablo was installed and CPU load was increased from 20% to 99%. Limitation was that the authors did not consider

historic data of resource utilization to better characterize side-effects of VM consolidation.

Sampaio et al. (2014) [85] presented a dynamic scheduling policy for consolidating VMs by finding energy optimizing opportunities resulting in decrease of energy consumption. The system was tested for both synthetic and real workload traces of Google. However, the system was not generic as authors did not consider data-intensive jobs and platform heterogeneity.

Peng et al. (2014) [86] focused on managing trade-off between energy consumption and performance as aggressively consolidating VMs result in degradation of performance. The proposed framework used Evolutionary Algorithm (EA) based Group Method of Data Handling (GMDH) approach for predicting the load and for migration VMs, MMT policy was used. The evaluation of GMDH-EA policy was done using Google data centers using host load traces. Results proved the effectiveness of proposed system in terms of both reduction of energy consumption and adherence to SLA.

Farahnakian et al. (2015) [87] presented an approach for VM consolidation based on current utilization of resources and prediction of their future utilization to reduce unnecessary VM migrations and maintain high level of SLA. A regression based model using K-nearest neighbour algorithm was proposed for prediction of future resource utilization. The system was evaluated using Planet Lab workload and CloudSim toolkit which indicated its effectiveness in terms of reduction of power consumed, VM migrations and violations in SLA.

Fu et al. (2015) [88] focused mainly on the development of VM selection algorithm. Authors used existing algorithms in literature for detection of over-loaded and under-loaded hosts. For selection of VMs to be migrated, authors not only focused on CPU usage but it further defined a variable which represented the level of resource satisfaction to prevent SLA violations. A novel policy of VM placement was also developed which placed a VM with minimum correlation coefficient on the target host as VM with high correlation has greater influence on the VMs existing on target host. The performance of the framework was evaluated on CloudSim

platform using randomly generated set of VMs and CPU utilizations. Results proved that the proposed system performed better in terms of reduction of energy consumption, SLA violation and VM migration time. However, the system was not tested using real data.

Mann et al. (2015) [89] found severe results by checking the efficiency of heuristics-based algorithms used in virtualized cloud data centers to consolidate VMs. Study was done to find what performance guarantees could be rigorously given by heuristics (MM and MBFD). Synthetic dataset was used for analysis. Algorithms MM and MBFD gave ideal results under suitable conditions but their combination did not give optimal results always. Limitation of the work were not practically applied.

Viswanathan et al. (2015) [90] presented a framework using best fit heuristic to be employed by backup service providers for re-mapping of incomplete tasks. Image of an object to be recognized was submitted by service requester mentioning a deadline. The mobile devices using Linux and Android based OS with variable capabilities were selected for evaluation. Profound influence on response period, relevance, mobile applications quality was obtained but presented system did not perform better than RR under heterogeneous and very stable working settings.

Lai et al. (2015) [91] proposed latency-aware DVFS for efficient transitions between states of power on architectures having multiple cores. Ratio of energy and performance, consumption of energy, energy delay product index, performance at run time were noted. To exploit variations applicable to shared-memory programming useful for chip multi-processors and non-trivial DVFS latencies, novel profile-guided DVFS was thought to be designed. Benchmarks like Graph 500, Malstone and Intel SCC running Barrelfish multi-kernel OS having multiple cores was used for experiment resulting in additional energy saving, reduction of product of energy-delay, reduction in execution time by 24%, 31.3%, 15.2% respectively in contrast to approaches that do not consider latency. The drawback is that authors did not consider latency incurred in frequency scaling and how effect on power of CPU chip made by temperature.

Dauwe et al. (2016) [92] analyzed energy modeling and working of HPC node having co-locations of applications. Performance of system, utilization of cores, nodes were the parameters used to find out applications interference among cores. Authors used slack based mapping to design an architecture for refining system performance. To indicate efficiency of model for scheduling, a simulated large scale HPC environment was used along with co-location aware scheduling heuristic. HPC system composed of system nodes with 4-core Intel Xeon E3-1225v3 processors. The proposed system was proficient in terms of important enhancements in contrast to scheduling heuristic that was oblivious to co-location interference. Future work included replicating the experiments in a homogeneous or heterogeneous system having more cores in processor nodes.

Zhang et al. (2016) [93] performed joint optimization over information technology and cooling systems in data centers. Authors simulated “Thermal-Aware Task Scheduling Algorithm (TASA) and Thermal-Aware Task Scheduling Algorithm-Backfilled (TASA-B)” based on task temperature profile, job information, thermal maps, and resource information obtained in Center for Computational Research (CCR) log files. The evaluation was done by comparing them with original job execution information logged in the CCR, which was scheduled by First Come First Served (FCFS). Authors simulated a real data center environment based on the CCR of State University of New York at Buffalo. Results indicated 16.1 F of average temperature and 6.1 F maximum temperature can be reduced by TASA with an overhead of increased job response time of 13.9%. 14.6 F of average temperature and 4.9 F maximum temperature can be reduced by TASA-B with an overhead of increased job response time of 11%.

Li et al. (2016) [94] focused on energy-aware VM migration and consolidation taking into account multiple resources which can reduce consumption of energy and maintain QoS. A meta-heuristic based algorithm- Modified Particle Swarm Optimization (MPSO) was developed for mapping of VMs to hosts. The algorithm was compared to traditional MBFD heuristic. CloudSim toolkit was used for evaluation

of the proposed work. Results showed reduction in energy consumption, VM migrations and number of active nodes as compared to MBFD. The work considered multiple resources for scheduling of VMs which imitated real life scenarios.

Dashti et al. (2016) [95] focused on dynamic VM placement. To reallocate migrated VMs from over-loaded hosts, authors modified PSO algorithm. Dynamic consolidation of under-loaded hosts was performed. The evaluation of proposed algorithm was done using CloudSim and results showed 14% reduction in energy consumption and significant reduction in number of migrations in comparison to previous works. However, algorithms for all phases of VM consolidation process were not designed.

Vasudevan et al. (2017) [96] performed profile based dynamic application assignment with a repairing GA for greener data centers. A profile based approach was presented for dynamically assigning applications to VMs in energy-efficient way. There was no compromise on QoS parameters such as workload balance and resource utilization. MetaCentrum2 workload trace and real data center (Czech National Grid Infrastructure) was used. Results indicated that the offered system outperformed present application allocation algorithm by 48% in terms of energy savings. Static application controlling approach lacked behind by 10% in terms of efficient resource utilization in comparison to proposed policy. Limitation was that some of the VMs mapping patterns were not investigated.

Gabaldon et al. (2017) [97] used blacklist multi-objective GA for energy saving in heterogeneous environments which considered both optimization and global makespan to reduce energy consumption. Real workload traces were used to evaluate the effectiveness of proposed algorithm. Results showed that there was 10% decrease in consumption of energy with the use of multi-objective GA in comparison to other techniques. Future work included using advanced plans for computation of needed resources for future workload.

Duan et al. (2017) [98] designed a scheduling algorithm called as “Pre Ant” policy. It comprised of a prediction technique based on fractal mathematics and ant colony

algorithm. The estimation model predicted the load and triggered execution accordingly. The scheduling was performed while reducing power consumption maintaining specific QoS. Google traces of workload were used for evaluation. Authors designed a simulator similar to CloudSim. Results indicated that “Pre Ant Policy” represented 76.17%, 75.28%, and 60.98% decrease in consumption of energy and the mean CPU utilization amplified by about 6%, 52% and 65% compared to FF, RR, and MM policies respectively. Limitation was that the assumptions not applicable to practical systems were made.

Zhou et al. (2017) [99] focused on addressing the energy-performance tradeoff. Authors proposed energy-efficient algorithms that also aimed to minimize SLA violations. Real world data was used and experiments were conducted on CloudSim toolkit to indicate the effectiveness of proposed algorithms.

Xu et al. (2017) [100] focused on addressing the energy performance trade-off by proposing a balanced VM scheduling method. Authors reduced degradation in VM performance by using a joint optimization model for energy consumption and performance. VM migration was performed very carefully so that performance should not be degraded. Experiments conducted on CloudSim indicated the effectiveness of the system.

Malekloo et al. (2018) [101] focused on striking an equilibrium among consumption of energy and delivered QoS. Authors proposed a multi-objective optimization algorithm for placement of consolidated VMs. The designed policy named as “Multi-objective Ant Colony Optimization” (MACO) approach was tested using variable data center settings configured in CloudSim toolkit. The policy was composed of two distinct multi-objective algorithms- ACO VMs’ placement and consolidation algorithm. The approach for VM consolidation focused on finding efficient VM placement solution reducing wastage of resources, communication cost and energy consumption. Based on VMs’ placement algorithm outcomes, the algorithm developed for consolidation optimized resource utilization by migration of VMs considering minimization of active servers, VM migrations, energy consumption and SLA violations at the same time. Results of the proposed approach were

compared with many heuristic and meta-heuristic algorithms indicating its superiority in terms energy efficiency and QoS maintenance.

Chaabounil and Khemakhem (2018) [102] focused on development of load detection policy for finding the under-loaded and over-loaded hosts. The load detection policy was motivated from MAD algorithm. The developed policy utilized two thresholds for finding over-loaded and under-loaded hosts by statistically analysing resource utilization history. The upper threshold was computed by addition of host median load strength with standard deviation. Subtraction of the standard deviation from the value of host median load strength gave the value of lower threshold. The host whose utilization increased beyond upper threshold was considered to be over-loaded and whose utilization was below lower threshold, it was considered to be under-loaded. CloudSim toolkit and Planet Lab workload traces were used to evaluate the proposed algorithms. The comparison was made with LR and MAD policies. The previously existing VM selection algorithm- MMT and VM placement algorithm- PABFD were used in experiments. Results made it clear that the proposed policies saved about 40% more energy in comparison to default algorithms in CloudSim. However, the developed policies resulted in more SLA violations than the other techniques.

Wang and Tianfield (2018) [103] focused on energy-efficient Dynamic Virtual Machine Consolidation (DVMC) by proposing a new algorithm – “Space-aware Best Fit Decreasing” (SABFD) policy for VM placement. Authors also designed a policy for VM selection named as “High CPU Utilization-based Migration VM selection” (HS). The system was tested using different combinations of policies using CloudSim toolkit and Planet Lab workload. Results illustrated that DVMC plans with combination of SABFD and HS yielded the best performance.

Mishra et al. (2018) [104] aimed to develop a complete VM placement policy involving mapping of a task to VM and VM to PM. Authors designed an “Energy-aware Task-based Virtual Machine Consolidation” algorithm. The evaluation of the proposed method was performed on CloudSim toolkit using randomly generated resource requirements and length of requests. The comparison of proposed algorithm was done with some standard techniques like FCFS, RR [29] and the results

indicated worthiness of the designed system as compared to benchmark techniques. The designed method minimized the task rejection rate, makespan, and energy consumption by switching-off the idle nodes.

Ashraf and Porres (2018) [105] designed a multi-objective based VM consolidation algorithm for cloud data centers using ant colony algorithm. The key aim was to decrease the VM migrations and maximize the number of released hosts. The main idea was to propose a VM migration plan to shift VMs from over-utilized servers and switch-off under-utilized ones for optimal utilization of resources. The proposed algorithm was tested using series of experiments conducted on Intel Core i7-4790 processor using randomly generated workloads. Authors considered environment composed of homogeneous VMs and PMs. Results showed that the proposed algorithm outperformed two existing ACO based approaches.

Arani et al. (2018) [106] focused on reduction of energy consumption by proposing a VM placement algorithm. VMs were mapped to hosts by using a policy based on best fit decreasing algorithm simultaneously reducing energy consumption and SLA violations. The designed algorithm used theory of learning automata, correlation coefficients and ensemble prediction algorithm for allocation of VMs to hosts. The algorithm was based on placing VM on that host whose VMs had minimum correlation with the selected VM for placement. Simulation results on CloudSim platform showed considerable improvement in reduction of energy consumption and SLA violations as compared to other baseline policies. However, the results could further be improved by using combination of neural networks and learning automata theory based approach. Moreover, author did not propose algorithms for other phases of VM consolidation.

Table 2.1 presents the summary of recent work on VM consolidation.

Cloud computing itself provides an effective solution for maintaining energy efficiency due to virtualization support. The continuous adoption of cloud computing will lead to a reduction of data center energy consumption by 31% during 2010-2020 [107]. Although energy requirements of cloud data centers are high, yet cloud installations have higher server utilization levels without compromising desired performance.

Table 2.1 Summary of work done on VM consolidation

Work	Focused on	Host load detection policy	VM selection policy	VM placement/ allocation policy	Summary/ Gaps
[107]	Energy-efficient VM placement	CSP decided the utilization threshold	Epoch method in the beginning of time slot	Dynamic programming, local search	The designed algorithm was 20% better in terms of reduction of energy consumption. QoS related parameters were not mentioned. Authors only focused on VM placement and ignored communication, network related parameters and I/O resources
[64]	Energy-aware task scheduling on multi-processor systems	Not Applicable	Not Applicable	Proposed Adaptive Minimal Bound First Fit (AMBFF) algorithm for tasks scheduling	The proposed algorithm was tested using random data and it was based on single criterion. Moreover, authors considered homogeneous architecture for testing
[65]	Energy-efficient Management of resources through VM consolidation	Fixed value of Single Threshold (ST)	“Minimization of Migrations” (MM), “Highest Potential Growth” (HPG), “Random Choice Policy” (RCP)	Modified Best Fit Decreasing (MBFD)	The designed framework did not consider QoS parameters. The values of CPU utilization were generated randomly. It was based on single criterion- CPU utilization
[66]	Design of generic framework for addressing energy performance trade-off	ST (fixed value)	MM	Bin packing heuristics	The designed framework was tested using random data. It was also based on single criterion- CPU utilization
[46]	Design of energy-efficient, QoS-aware VM consolidation framework	Double thresholds (fixed)	MM HPG RCP	MBFD	The designed framework was tested using random data. It was also based on single criterion- CPU utilization
[68]	Dynamic VM consolidation considering SLA related metrics	Dynamic thresholds computed by statistically analyzing CPU utilization history	MM HPG RCP	MBFD	The designed system was tested using real dataset of CoMon project –Planet Lab workload traces. However, authors considered only single-core CPUs and single criterion

Table 2.1 Summary of work done on VM consolidation (contd.)

Work	Focused on	Host load detection policy	VM selection policy	VM placement/ allocation policy	Summary/ Gaps
[71]	Dynamic VM placement	Not Mentioned	Not Mentioned	Used nature inspired Ant Colony Optimization (ACO) algorithm	The authors created their own simulator for evaluation of the algorithm. However, all the servers considered were homogeneous (same capacity)
[72]	Energy-efficient VM provisioning	Not Applicable	Not Applicable	Developed “Adaptive Power-aware Virtual Machine Provisioner” (APA-VMP) using “Self-aware Particle Swarm Optimization” (SAPSO) algorithm	The authors tested the provisioner using CloudSim, random data. The future work included improving the performance of algorithm using combination of neural networks and adaptive algorithms
[73]	Energy-efficient Management of virtualized data center	Not Applicable	Not Applicable	Designed a scheduling algorithm for mapping of VMs to data center nodes	Authors used a power-aware simulator to evaluate the proposed algorithm using heterogeneous workloads on the basis of energy and SLA related metrics. However, the designed algorithm was not suitable for applications with variable SLAs
[74]	Dynamic VM consolidation for reduction of energy consumption and SLA violations	Calculated threshold using standard deviation and mean of CPU utilization values	Maximum Correlation Coefficient (MCC)	Algorithm used mean and variance related computations	Authors used CloudSim platform for evaluation on Planet Lab data. The proposed system was better in terms of energy savings and reducing violations in SLA as a whole but it performed a little worse in terms of reduction of only energy consumption as compared to previous algorithms
[77]	Detection of over-loaded hosts	Maximum mean inter-migration time (Markov model)	Not mentioned	Not mentioned	Authors used CloudSim platform for evaluation on Planet Lab data. However, only phase of VM consolidation –load detection was targeted
[55]	VM Consolidation	SLA violation decision algorithm (SLAVDA)	Previous algorithms	Previous algorithms	Authors designed a SLAVDA algorithm to determine whether a host is over-loaded with SLA violations or not. However, only single resource (CPU) was considered

Table 2.1 Summary of work done on VM consolidation (contd.)

Work	Focused on	Host load detection policy	VM selection policy	VM placement/ allocation policy	Summary/ Gaps
[79]	Multi-objective VM placement	Not Mentioned	Not Mentioned	Multi-objective Ant Colony Optimization (MACO)	The authors considered all the servers to be homogeneous for evaluating the algorithms and randomly generated problem instances were used
[88]	Design of VM selection algorithm	Used previous policies	Designed a policy based on CPU utilization and level of resource satisfaction	Minimum correlation coefficient	Authors used CloudSim platform for testing using synthetic data.
[94]	Energy-aware VM migration and consolidation	Not mentioned	Not mentioned	Mapping of VMs to hosts was done using Modified Particle Swarm Optimization (MPSO) algorithm	Evaluation of framework showed that it outperformed traditional MBFD algorithm. Authors considered multiple parameters.
[101]	VM placement and optimization	Not mentioned	Not mentioned	MACO	CloudSim, was used to evaluate the work. Results indicated minimization of violations in SLA, energy consumption and number of VM migrations
[102]	Development of load detection policy	Double thresholds based on Median Absolute Deviation (MAD)	Previously designed MMT policy	Previously designed Power-aware Best Fit Decreasing (PABFD)	To evaluate the proposed algorithm, CloudSim was used. However there were more SLA violations than other techniques
[103]	Energy-efficient Dynamic VM Consolidation (DVMC)	Not mentioned	High CPU utilization based migration VM selection	Space-aware Best Fit Decreasing (SABFD)	Framework was evaluated using CloudSim toolkit and Planet Lab real workload data. However, it only considered single criteria – CPU utilization
[104]	Energy-aware task based VM consolidation algorithm	Not mentioned	Not mentioned	Mapping of tasks to VM and VM to PM Energy-aware Task based VM Consolidation	The framework was evaluated using CloudSim toolkit and randomly generated resource requirements. However, authors did not target all phases of VM consolidation

Table 2.1 Summary of work done on VM consolidation (contd.)

Work	Focused on	Host load detection policy	VM selection policy	VM placement/ allocation policy	Summary/ Gaps
[105]	Minimizing the number of migrations and maximizing the number of released hosts	Not mentioned	Not mentioned	Proposed a VM migration plan to migrate VMs from over-utilized hosts. The proposed algorithm used MACO	Authors designed a test best for evaluating the proposed work using random workload. Moreover, all PMs were considered to be homogeneous
[106]	Reduction of energy consumption and SLA violations	Not mentioned	Not mentioned	VM placement based on theory of learning automata, ensemble prediction algorithm and correlation coefficients	Authors used CloudSim platform for validation of work. However, the results could further be improved. Moreover, author did not propose algorithm for other phases of VM consolidation.

Major information technology organizations have initiated efforts towards offering green cloud computing. Apple, Google, Yahoo and Facebook are using renewable energy in their data centers as reported by Greenpeace [108, 109]. QoS is the basic demand for cloud user so cloud provider must keep it in foremost priority. To make data centers efficient not only in regards of reduction of energy consumption but also minimizing SLA violations is the need of the hour.

2.1 Research gaps

Energy efficiency in cloud computing is an active and vast research area. But researchers are working on sub parts of the solution domain. Very few researchers have tried to investigate the trade-off between energy efficiency and QoS. However, QoS is the foremost parameter of concern for a user. Cloud applications can present varying workloads. It is therefore essential to carry out a study of approaches that can potentially lead to more efficient resource provisioning and consequently prevail energy efficiency [65]. An optimal approach that collectively handles energy efficiency, performance and gives economic benefits has to be followed. Most of frameworks try to aggressively reduce energy consumption without focusing on QoS factors. The challenges include maintaining reliability of a server because power cycling may reduce it, performing VM consolidation without affecting QoS, accurate application performance management in a virtualized environment while maintaining SLA [65]. Major research gaps are:

- i. Development of framework for energy efficiency which should be holistic as limited work exists in the literature [110, 111].
- ii. Current state of the art in cloud infrastructure has no or limited consideration for supporting energy-aware service allocation that meets QoS [65, 99, 112].
- iii. Leverage the combination of energy efficiency and placement of VM [99].
- iv. Solution for VM performance degradation [72, 100].
- v. Implementation of the optimization techniques for energy efficiency on cloud environment taking into consideration the real host and VMs configuration [113, 114].
- vi. Investigation of impact of network characteristics, communication cost, overhead, system components like disk, main memory [115].

From the literature studied, it is well clear that following are the limitations of most of the existing energy-efficient VM consolidation frameworks:

- i. The thresholds for detecting under-loaded or over-loaded hosts are not based on analysis of historical data and are fixed [65, 66].
- ii. The factors other than CPU utilization like memory, bandwidth are ignored during VM selection for migration [55, 65, 66, 69, 70].
- iii. VM placement problem is addressed using single criterion based solution [55, 69, 70].
- iv. Most of the existing frameworks are not generic and do not consider both CPU and I/O tasks [55, 65, 66, 69, 70].

Researchers do not target all phases of VM consolidation and propose solutions for a particular phase.

2.2 Objectives

The objectives of this study are:

- i. To study and analyze QoS driven cloud computing approaches with respect to energy-efficient techniques.
- ii. To design a QoS-based energy-efficient framework for cloud computing.
- iii. To simulate the proposed framework for a proof of concept.
- iv. To demonstrate the validity and usability of proposed framework via case study/scenarios.

In the present research work, a holistic QoS-aware multi-criteria-based energy-efficient framework for cloud computing data centers which considers both CPU intensive and I/O intensive tasks. The proposed framework addresses all the four phases of VM consolidation using adaptive thresholds, considering CPU, memory, bandwidth utilization during VM migration and providing multi-criteria based solution for VM placement.

Chapter 3

Design of QoS-based Energy-Efficient Framework for Cloud Computing Data Centers

This Chapter presents the design of QoS-based energy-efficient VM consolidation framework for cloud computing data centers.

VM consolidation is a vital method used for reducing power consumption. Existing VM consolidation frameworks generally are based on single criteria. They also do not consider memory, network related parameters apart from CPU and are not generic. Moreover, they aggressively reduce energy consumption without considering QoS parameters.

This thesis illustrates QoS-aware energy-efficient framework for efficient provisioning of resources in cloud data centers. The work relies on VM consolidation approach for reducing energy consumption and maintaining energy efficiency.

The major idea is to balance the resource utilization by turning off idle servers and distributing the VMs from over-utilized servers.

3.1 The proposed framework

A novel multi-criteria based VM consolidation framework is presented in this study. The proposed framework is holistic as it solves the allocation of over-loaded and under-loaded hosts in one step.

The key features of the proposed framework are:

- It is a generic framework as it considers both CPU and I/O intensive tasks.
- It relies upon adaptive threshold in order to suit dynamic cloud environment for detection of over-loaded hosts.
- The memory and network related parameters apart from CPU utilization are also considered during VM migration.
- A Multi-Criteria Decision Making (MCDM) algorithm is used to address the problem of VM placement and detection of under-loaded hosts.
- It also considers SLA metrics for maintaining QoS.

- The algorithms are proposed for all phases of VM consolidation.

This study uses novel algorithms to tackle the resource scheduling problem in cloud data centers. It is based on the idea of solving the allocation problem for both under-loaded and over-loaded hosts in one step rather than separate allocation. Therefore, both under-loaded and over-loaded hosts are put in migration list and then proposed multi-criteria based deployment algorithm is applied on them to find new VM placement. Thus, this framework provides a holistic approach to solve the problem. VM migration is done keeping in view both CPU and I/O intensive tasks considering multiple parameters related to CPU, memory and bandwidth.

VM placement and detection of under-loaded hosts is also addressed by proposing a multi-criteria based solution. For host overload detection, an adaptive threshold based algorithm is proposed in order to comply with dynamic cloud environment. The effectiveness of proposed framework is illustrated by execution on CloudSim [116] platform using real dataset and comparison with state-of-art algorithms.

3.1.1 High level architecture

The high level architecture of the proposed framework is illustrated in Fig. 3.1. At highest level the users communicate with CSPs. CSPs serve the requests made by users by scheduling them on various VMs hosted by servers/PMs (hosts) in cloud data centers.

The high level architecture shows two major modules for communication:

Monitoring module: This unit monitors the values of resource utilization and related data. It checks whether the incoming workload is CPU intensive or I/O intensive for making migration decision. The input data corresponds to configuration of hosts and VMs, Planet Lab dataset. Energy consumption is monitored by using real data of SPEC power benchmark [117]. This module also checks for SLA violations. SLA is violated when an entity is not able to get the required amount of resource when requested. Various SLA related metrics are used for monitoring the performance of the framework as discussed in Chapter 4.

Scheduling decision module: It is composed of sub-modules for under-loaded hosts detection, over-loaded hosts detection, VM selection and VM placement or

allocation. The new requests are assigned based on allocation algorithm and existing resources are optimized using the procedure of VM consolidation involving the above mentioned four phases resulting in hosts being switched on/off and corresponding values of power consumption are shown by power meter.

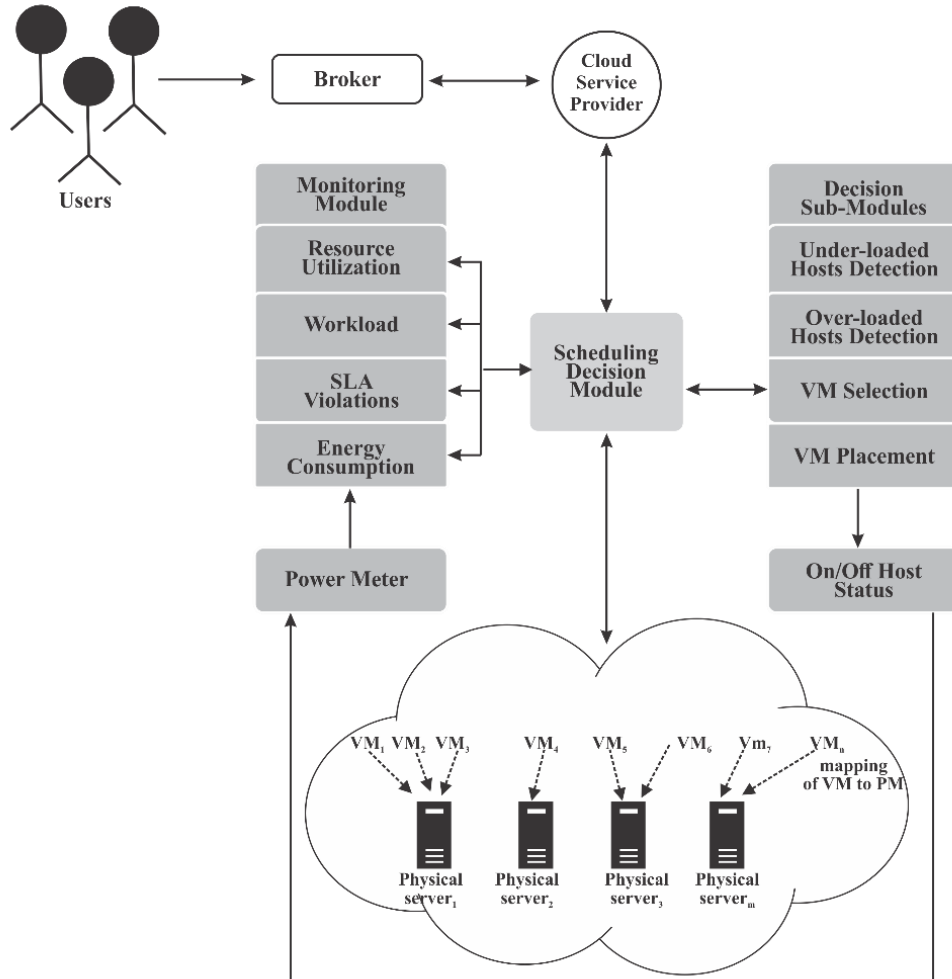


Fig. 3.1 High level architecture of the proposed framework

3.1.2 Low level architecture

The low level architecture presents a detailed view of the framework. The components are described as follows:

Monitoring module: This module is same as described in high level architecture. The monitored values of parameters are extracted by parameter selection unit which are given to scheduling decision making sub-modules according to their needs. Memory, CPU and network related parameters are used.

Scheduling decision module: It is composed of sub-modules marked as Module 1 for under-loaded hosts detection, Module 2 for over-loaded hosts detection, Module 3 for VM selection and Module 4 for VM placement or allocation.

Fig. 3.2 shows the algorithms used by these modules (i) K-Medoid Interquartile Mid-Range (KMIMR) algorithm to find adaptive threshold for detecting over-loaded host as shown in Module 2, (ii) Improved Technique for Order of Preferences by Similarity to Ideal Solution (TOPSIS)-based Underload Detection Algorithm (ITUDA) to find under-loaded hosts as illustrated in Module 1, (iii) two policies Maximum Ratio of CPU to Product of Memory and Bandwidth (MRCPBM), Minimum Product of CPU, Bandwidth and Memory (MPCBM) for CPU and I/O intensive tasks respectively considering CPU, memory, bandwidth utilization for VM selection as shown in Module 3, (iv) a multi-criteria based solution for VM placement and developed Improved TOPSIS-based Energy-Efficient VM Deployment (ITEED) algorithm as depicted in Module 4. It tries to minimize energy consumption and SLA violations at the same time. The detailed view of the framework is shown in Fig. 3.3.

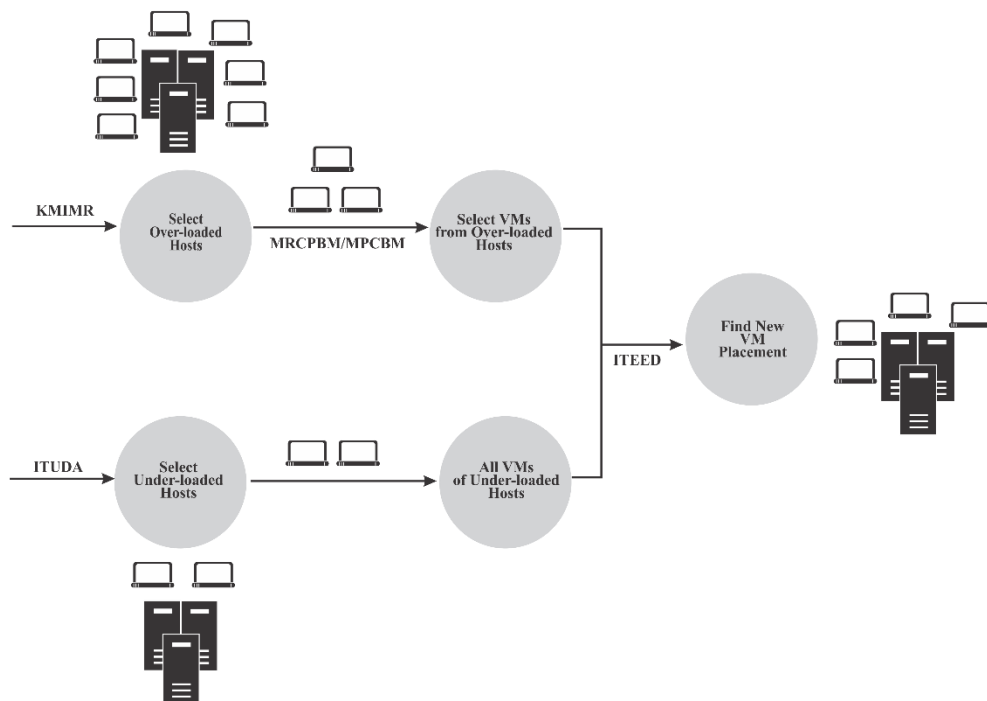


Fig. 3.2 Algorithms used by scheduling decision module

Module 1 uses ITUDA involving the steps of normalization of parameters taken from parameter selection unit. The next steps involve calculation of weights of these parameters, degree of similarity and dissimilarity to finally get under-loaded hosts. Module 2 uses KMIMR policy to find over-loaded hosts which includes division of values of resource utilization into different clusters, finding IQR of these clusters and finally calculating the mid-range. Outcome of this step is an adaptive threshold. The hosts having utilization more than this threshold are over-loaded. Module 3 finds the VMs to be migrated as per the received set of tasks. It uses two policies- one for CPU intensive tasks and other for I/O intensive tasks to select the VMs. Module 4 uses multi-criteria-based ITEED policy involving steps of normalization, calculation of weights, degree of similarity and dissimilarity to for allocation of VMs.

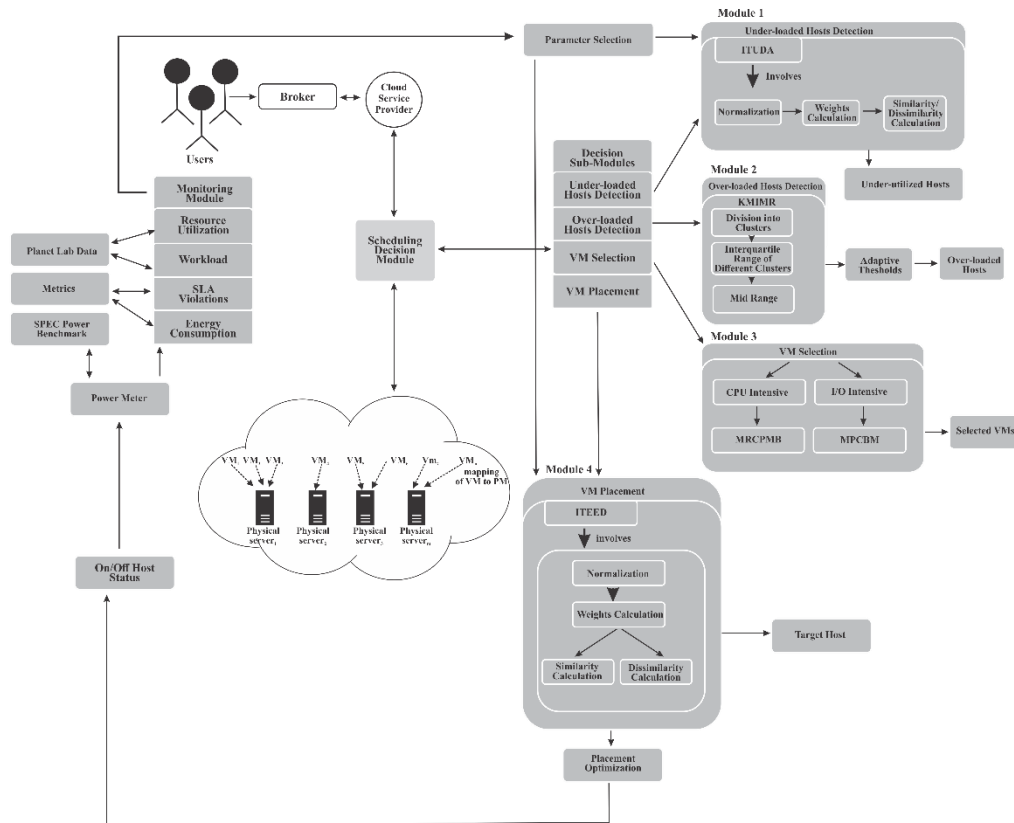


Fig. 3.3 Low level architecture of the proposed framework

All these modules use host monitoring unit followed by parameter selection unit to get input data. Placement is optimized and cycle is repeated resulting in turning

on/off hosts according to need. These algorithms are explained step by step in Chapter 4

3.2 Working of the framework

The working of designed framework is illustrated in Fig. 3.4. Our framework encapsulates novel solutions for all the four phases of VM consolidation and provides a holistic view of solution by migrating resources from over-loaded and under-loaded machines in one go rather than using separate steps.

The optimization cycle is repeated after every 300s and information about resource utilization is collected during this interval. In the beginning newly arrived VMs are assigned to existing hosts using ITEED algorithm. After that, all the hosts are checked to find over-loaded hosts according to the adaptive threshold based on resource utilization history.

KMIMR policy is proposed to find the value of threshold. The value indicates that a host having utilization more than this threshold value is over-loaded. The procedure of KMIMR is explained in next Chapter.

In the next step, VMs are selected from over-loaded hosts using proposed MRCPMB and MPCBM policies for CPU and I/O intensive tasks respectively. Then MBFD [65] policy is applied on selected VMs which are earlier sorted according to utilization for finding their probable target PMs.

If proper allocation is found for VM, then it is added to migration list. This process of finding over-loaded hosts is repeated until there are no more hotspots. In the next step, under-loaded hosts are found using proposed ITUDA policy.

Now, MBFD policy is again used to find a host which will be able to accommodate all the VMs (sorted) of under-loaded hosts.

If a suitable host is found which can accommodate all the VMs, then all the VMs are added to migration list, otherwise no VM is added.

Finally, ITEED is used to find new target hosts for VMs deployment based on multiple criteria and then migrations of VMs take place.

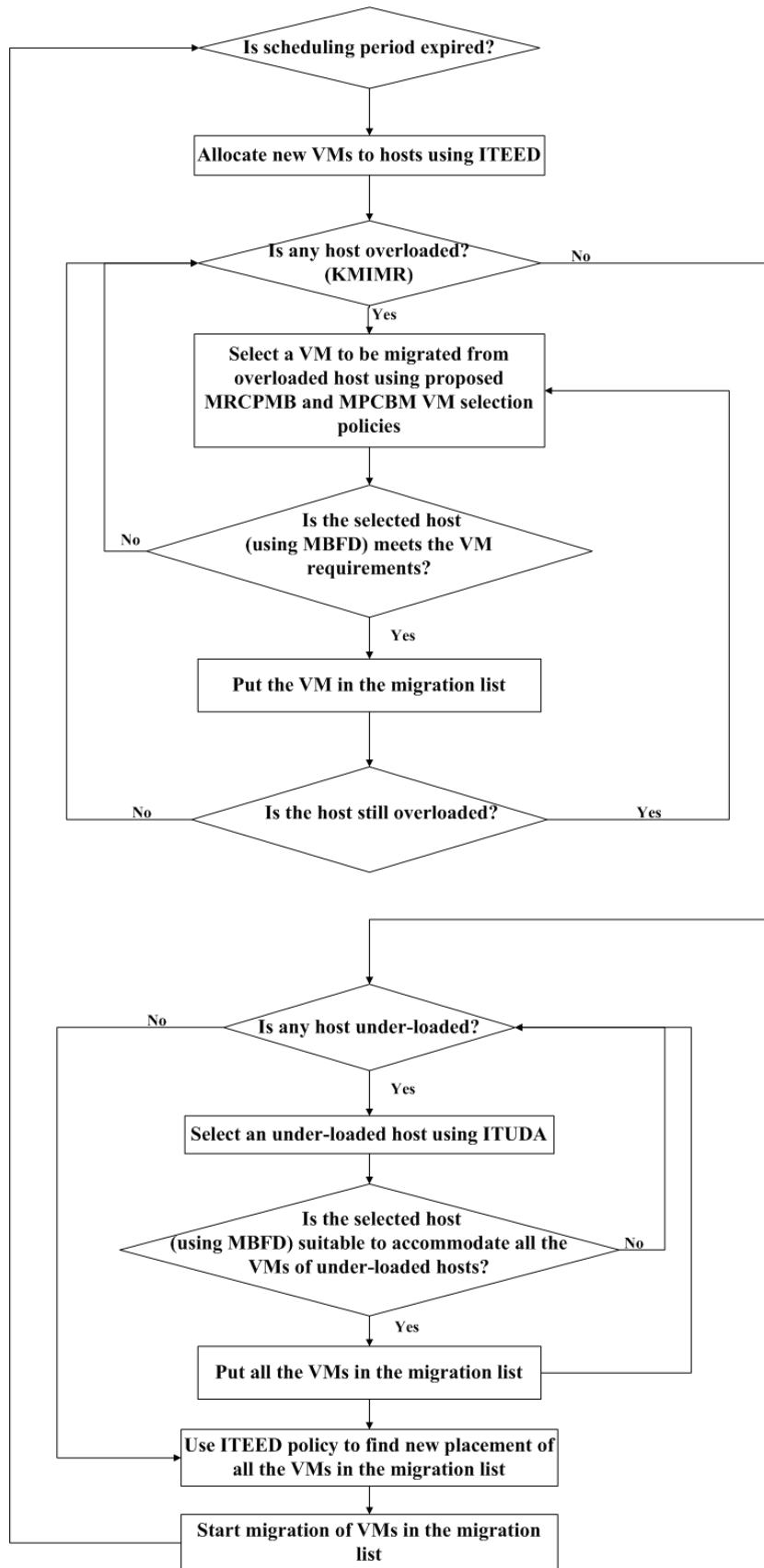


Fig. 3.4 Working of the proposed framework

Thus, a holistic approach is used to encapsulate the process of allocation to resources of under-loaded and over-loaded hosts in one unit. The designed framework has provided solutions for all phases of VM consolidation and is generic.

Chapter 4

Experimental Setup and Implementation of the Framework

This Chapter describes the environment, datasets and parameters used for testing the performance of the proposed framework. It also presents the algorithms designed for various phases of VM consolidation.

4.1 Experimental setup

CloudSim toolkit is used for evaluating the proposed framework. It is not feasible to perform large scale experiments repeatedly on real cloud infrastructure [55, 65, 66, 99]. Thus, CloudSim toolkit is preferred because of its ability to perform modelling of on-demand virtualized resources and their management. Power-aware simulations can be performed by extending the basic toolkit.

CloudSim is a java based execution environment developed by researchers in CLOUDS lab [116]. Simulation of power-aware data centers is performed by extending the power packages provided along with the simulator.

Multiple data centers having heterogeneous resources hosting various applications which run heterogeneous VMs result in formation of dynamic mixed workload on each host. There are 800 heterogeneous physical hosts including 400 HP ProLiant ML110 G4 and 400 HP ProLiant ML110 G5 hosts.

VMs are Amazon EC2 instances. Table 4.1 shows the characteristics of hosts and VMs used in the experiment.

The traditional power models consider CPU as the major source of power consumption. However, with the utilization of virtualization-based power management techniques in multi-core CPUs, it is not the only consumer of power [99]. This makes design of an analytical prediction model for power consumption a complex problem.

Thus, we have used real power consumption data taken from SPEC power benchmark [117]. The power models employed are HP ProLiant ML110 G4 and HP ProLiant ML110 G5 as shown in Table 4.2.

Table 4.1 Configuration of VMs and hosts used in experiments

Configuration	Host	VMs
Types	2	4
MIPS	{1860, 2660} MHz	{2500, 2000, 1000, 500} EC2 Compute Units
PES	{2, 2}	{1, 1, 1, 1}
RAM	{4, 4} GB	{0.85, 1.7, 1.7, 0.6} GB
BW	1 Gbit/s	100 Mbit/s

Table 4.2 Power consumption in watts

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

4.1.1 Workload trace

It is necessary to perform experiments using real data in order to ensure applicability of the proposed framework in real cloud environment. Thus Planet Lab workload trace [118] is utilized for simulation of data centers. This data was the part of ‘CoMon’ project. The infrastructure ‘CoMon’ monitored Planet Lab nodes. It contains CPU utilization of VMs collected after every five seconds. Planet Lab data was collected on 10 different days using 800 hosts and diverse number of VMs on every day. The description of data is as shown in Table 4.3. During the experiment, each VM is randomly given a workload trace of one of the VMs from the data of a particular day.

Table 4.3 Workload traces

Workload trace name	Meaning	Number of hosts	Number of VMs
20110303	This data is collected on 3 March, 2011	800	1052
20110306	This data is collected on 6 March, 2011	800	898
20110309	This data is collected on 9 March, 2011	800	1061
20110322	This data is collected on 22 March, 2011	800	1516
20110325	This data is collected on 25 March, 2011	800	1078
20110403	This data is collected on 3 April, 2011	800	1463
20110409	This data is collected on 9 April, 2011	800	1358
20110411	This data is collected on 11 April, 2011	800	1233
20110412	This data is collected on 12 April, 2011	800	1054
20110420	This data is collected on 20 April, 2011	800	1033

4.1.2 Metrics

Most of the existing frameworks try to aggressively reduce energy consumption without maintaining QoS. QoS is the measure of degree of compliance with SLA constraints. Aggressive consolidation may lead to SLA violations as some of the machines may not get the required resources when requested. The proposed VM consolidation framework tries to reduce both consumption of energy and SLA violations simultaneously. The output parameters which evaluate the performance of designed framework are energy efficiency, energy consumption, number of migrations, SLA Violation Time Per Active Host (SLATAH), Performance Degradation due to Migration (PDM) and overall SLA violation.

Energy consumption: Reducing consumption of energy by servers is crucial. It is defined as summation of power consumption during a time period (Eq. (4.1))

$$e(t) = \int_t p(t) \quad (4.1)$$

Number of migrations: It depicts the number of VM migrations performed by VM manager during VM consolidation.

SLATAH: The percentage of SLA violation time for which a host has experienced 100% utilization. SLATAH is defined in Eq. (4.2)

$$SLATAH = \frac{1}{n} \sum_{x=1}^n \frac{t_{s_x}}{t_{a_x}} \quad (4.2)$$

Where n is number of PMs; t_{s_x} is time for which host x has experienced 100% utilization and t_{a_x} is active time of host x.

PDM: Eq. (4.3) represents PDM, where m is number of VMs in data center; C_{d_y} is predicted PDM of VM y; C_{r_y} is total CPU utilization requested by VM y during its lifetime and C_{d_y} is 10% of requested CPU utilization of y during its migration.

$$PDM = \frac{1}{m} \sum_{y=1}^m \frac{C_{d_y}}{C_{r_y}} \quad (4.3)$$

Overall SLA violation (SLAV): It is defined as product of SLATAH and PDM as shown in Eq. (4.4)

$$SLAV = SLATAH * PDM \quad (4.4)$$

Energy Efficiency: It is defined as in Eq. (4.5)

$$\text{energy efficiency} = 1/(\text{energy consumption} * \text{SLA violations}) \quad (4.5)$$

4.2 The proposed resource management policies

This study uses novel algorithms to tackle resource scheduling problem in cloud data centers. It solves the allocation problem for both under-loaded and over-loaded hosts in one step rather than separate allocation. Therefore, both under-loaded and over-loaded hosts are put in migration list. VMs from these hosts are selected for migration according to proposed policies for CPU and I/O intensive tasks. The proposed multi-criteria based deployment algorithm is applied on them to find new VM placement. Thus, this framework provides a holistic approach to solve the problem. This Section presents the proposed policies for all the four phases of VM consolidation.

4.2.1 Host overload detection algorithm

Most of the existing algorithms in literature use fixed value of threshold to find over-utilized host. The static value of threshold is not suitable due to dynamic cloud environment. This work introduces an adaptive threshold based algorithm by applying K-medoids algorithm before statistical analysis of historical data. It divides the set of resource utilization into clusters. It is preferred to other clustering algorithms such as K-means because of being more robust to outliers.

Over-loaded hosts are detected according to the computed value of upper threshold. Hosts having utilization more than this threshold are considered to be over-loaded. KMIMR algorithm is used to calculate the value of upper threshold according to the prediction of resource usage based on historical values of resource utilization. KMIMR automatically adjusts the value of threshold as the workload changes.

Algorithm 1 shows the steps used in KMIMR. Firstly, number of different sets of utilization are created dependent on the number of clusters as shown in line 1. The number of clusters can be decided using empirical approach. The next step is to calculate the IQRs of different clusters (lines 2 to 4). The mid-range from the IQRs is found as shown in line 5. Line 6 calculates the value of upper threshold using safety parameter (L).

Algorithm 1: KMIMR

Input: set of utilization (U), number of clusters (k), safety parameter (L)

Output: Threshold T_1

1: $C = \text{KMedCluster}(U, k)$

2: **for** $x = 1$ to $C.$ length **do**

3: $\text{IQR}[x-1] = \text{TQ}_3 - \text{TQ}_1$

4: **end for**

5: $\text{Iqr} = (\text{Max}(\text{IQR}) + \text{Min}(\text{IQR}))/2$

6: $T_1 = (1 - L \cdot \text{Iqr})$

END

4.2.2 VM selection methods

The existing methods for VM selection consider single resource i.e. CPU and they do not consider both CPU and I/O intensive tasks. In this work, VM selection methods are proposed considering both I/O and CPU intensive tasks simultaneously. A task which is CPU intensive will take more CPU cycles and its completion time depends upon the processor speed. I/O intensive tasks, on the other hand perform more of the I/O operations and their completion time depends upon the time for which they kept waiting to complete I/O operations.

The policy proposed for processing of CPU intensive task is:

Maximum Ratio of CPU to Product of Memory and Bandwidth (MRCPMB)

The energy consumption by CPU is more as compared to memory and bandwidth in the case of tasks which are CPU intensive. So, we choose a VM whose CPU consumption is more, to migrate on another host to reduce consumption of energy. Suppose VM_i and VM_j are two VMs. VM_i is selected for migration such that the following condition holds:

$$\left(\frac{\text{CPU}}{\text{Memory} * \text{Bandwidth}} \right)_i > \left(\frac{\text{CPU}}{\text{Memory} * \text{Bandwidth}} \right)_j$$

The policy for processing I/O tasks is:

Minimum Product of CPU, Bandwidth and Memory (MPCBM)

In I/O intensive tasks, the CPU, memory and bandwidth all amount for considerable energy consumption. So, select such a VM which consumes less CPU, memory and bandwidth as its migration will be easier. Therefore, in this case, we select a VM_i such that the following condition holds:

$$(\text{CPU} * \text{Bandwidth} * \text{Memory})_i < (\text{CPU} * \text{Bandwidth} * \text{Memory})_j$$

These VM selection methods are used to select VMs for migration from overloaded hosts considering CPU, bandwidth and memory utilization.

4.2.3 VM deployment/placement algorithm

The existing literature shows that the problem of VM placement is a multi-dimensional bin packing problem [58, 66]. However, the authors in [119] report that it is similar to 3D bin packing but not precisely the same. The VM placement problem can actually be referred to as vector bin packing problem or a Non Polynomial (NP) hard problem. The work found in the literature on VM placement has various anomalies and drawbacks. E.g., the heuristics such as FF checks the machines sequentially and assigns VM to the first suitable machine. However, this creates a major unbalancing of the load. Thus, multidimensional optimization is a solution which tries to use parameters considering all these dimensions unlike single criteria based approaches.

In single-criteria based strategy, the mono objective function is used which focuses on a single parameter minimization or maximization. Whereas, in the case of multi-criteria based strategies, the multi-objective function is used which considers multiple parameters resulting in fulfilment of multiple goals. E.g., considering increase in power consumption, multiple resources capacities, resource availability, the delay caused due to migration (network) while placing a VM makes the proposed scheme in this study, a MCDM solution. The formulation of VM placement solution considering multiple criteria is the only feasible strategy in the real scenario because of dynamic cloud workloads. Most of the solutions in literature for selecting hosts include bin packing heuristics which are based on a single criterion. They only consider energy consumption and not energy efficiency (reducing power consumption and violations in SLA simultaneously).

In this thesis, an effort is made to design multi-criteria (energy consumption and SLA violations reduction) based VM deployment framework for finding an effective host for VM placement considering CPU, RAM and network related parameters. It manages the trade-off between energy consumption and SLA compliance. This work on VM placement contributes to:

- It presents a design of a VM deployment framework considering CPU, RAM and network related parameters to determine the target hosts for VM deployment while reducing the energy consumption and SLA violations simultaneously (multi-criteria).
- Improved Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS)-based Energy- efficient Deployment policy (ITEED) is proposed to find energy-efficient hosts.
- It categorizes hosts into five classes ranging from extremely energy-efficient to minimal energy-efficient.
- A case study based approach evaluates the efficiency of the proposed algorithm by ranking the hosts and is defined in Chapter 5.

VM deployment (placement) is an MCDM problem. Decision-making includes handling trade-offs or negotiations between various conflicting criteria. Hwang and Yoon developed the TOPSIS method [120] for decision making. It is established on the idea that the selected alternative should be the minimum distance (Euclidean distance) apart from the optimal solution and should have maximum distance from the non-optimal solution. For the assumed ideal solution, all the attributes have maximum values in the dataset of satisfying solutions, and for the case of the supposed negative ideal solution, all the attributes have minimum values in the dataset. Therefore, the solution generated by TOPSIS is not only nearest to optimum, but it is also farthest from non-optimum. In TOPSIS algorithm, all the parameters are given equal or random weights. In Improved TOPSIS technique, the decision maker decides the relative importance of weights systematically using Analytical Hierarchy Processing (AHP) Technique [121]. AHP is the most extensively used technique for solving MCDM problems. It is a Multi-Attribute Decision Making (MADM) method, which makes ill-structured difficult problems easier by structuring decision factors in a hierarchal way. The benefits AHP provide over other methods are its flexibility, spontaneous appeal to the decision makers and its capability to find irregularities. In addition to these advantages, it also decreases biases in decision making [122]. Because of these benefits of TOPSIS with AHP (improved TOPSIS), the designed VM deployment framework has applied ITEED for mapping of VMs on hosts.

Parameters for Host Selection

Power consumption is a vital parameter considered in this work. In addition to it, various other parameters are identified for selection of target hosts in terms of energy efficiency. The selection of criteria considered by ITEED is motivated from [57], and parameters are described in Table 4.4. ITEED policy benefits from the MCDM Improved TOPSIS algorithm.

Table 4.4 Criteria considered by ITEED

Criteria	Meaning
Available Capacity in terms of RAM (AC RAM)	The PM should have the maximum available capacity in terms of RAM
AC Capacity in terms of MIPS (AC MIPS)	The PM should have the maximum available computation capacity (power) in MIPS
Number of VMs Allocated (VMs)	The total VMs on the selected PM should be minimum
Migration Delay (MD)	The delay incurred by VM on allocating it to selected PM should be the minimum
Increase in Power Consumption (PI)	Increase in power of selected PM should be least

Scores of all candidate hosts represented as PMs (which can host VMs) are calculated. The host (PM) with the maximum score is chosen. Criteria can be of benefit or cost-type in ITEED policy. The higher value of benefit type criteria and the lowest value of cost non-benefit type criteria indicates solution is near to the optimum point. Scores are calculated in ITEED based on the following conditions:

- The host (PM) that is selected should have the maximum available capacity in terms of RAM.
- The host (PM) that is selected should have the maximum available computation capacity (power) in MIPS.
- The number of VMs in the selected host (PM) should be minimum.
- The delay incurred by VM on allocating it to selected host (PM) should be minimum.
- Increase in power consumption of the selected host (PM) should be least.

Proposed Improved TOPSIS-based Energy-efficient Deployment Policy

ITEED takes into account multiple resources related to CPU, RAM and network for decision-making. The proposed policy is energy-efficient as it considers parameters for reducing energy consumption and SLA violations both. It selects the host that has the minimum number of VMs. So, competent for shared resources become

less, leading to lesser SLA violations. Further, choosing hosts with the highest available RAM and computation power (MIPS) certifies that VMs will be allocated with ease and reduces SLATAH. Moreover, considering migration delay decreases violations in SLA during migration and therefore lowers down the PDM. The procedure of ITEED is:

Step (1): Make an evaluation matrix (Fig. 4.1) containing m hosts (options) and n attributes (criteria). The intersection of each option and parameter is called as $host_{ij}$, $i = 1, 2, \dots, m$ and $j = 1, 2, \dots, n$. Thus, the matrix is $(host)_{m \times n}$ (Eq. (4.6)), PH₁, PH₂, PH_m are the m hosts or PMs (alternatives) and Pr₁, Pr₂, ... Pr_n are the different parameters considered for the selection of hosts. P_{ij} denotes the value of j th parameter of i th host.

$$Host_{m \times n} = \begin{matrix} & \begin{matrix} Pr_1 & Pr_2 & Pr_3 & \dots & Pr_{(n-1)} & Pr_n \end{matrix} \\ \begin{matrix} PH_1 \\ PH_2 \\ \vdots \\ PH_{m-1} \\ PH_m \end{matrix} & \begin{bmatrix} P_{11} & P_{12} & P_{13} & \dots & P_{1(n-1)} & P_{1n} \\ P_{21} & P_{22} & P_{23} & \dots & P_{2(n-1)} & P_{2n} \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ P_{(m-1)1} & P_{(m-1)2} & P_{(m-1)3} & \dots & P_{(m-1)(n-1)} & P_{(m-1)n} \\ P_{m1} & P_{m2} & P_{m3} & \dots & P_{m(n-1)} & P_{mn} \end{bmatrix} \end{matrix} \quad m \times n$$

Fig. 4.1 Evaluation matrix

Step (2): $(Nhost)_{m \times n}$ matrix is constructed by normalizing the matrix $(host)_{m \times n}$ as shown in Eq. (4.7). The values in the formed matrix vary from 0 to 1, where 1 indicates the utmost relevant parameter, and 0 indicates the least relevant parameter.

$$Nhost = (host_{ij})_{m \times n}, \quad (4.6)$$

using the normalization method

$$Nhost_{ij} = \frac{host_{ij}}{\sqrt{\sum_{i=1}^m host_{ij}^2}}, \quad (4.7)$$

where $i = 1, 2, \dots, m$ and $j = 1, 2, \dots, n$

Step (3): A set of weights WT_j (for $j = 1, 2, \dots, n$) such that $\sum_{j=1}^n WT_j = 1$ is found for parameters (criteria). AHP technique was used to calculate the weights in an orderly manner [121]. The steps of AHP [123] using radical root technique for estimating weights are:

Step 3(i): A pair-wise assessment matrix (Fig. 4.2) is created using the relative significance scale of criteria shown in Table 4.5 [123]. As there are n parameters,

the pair-wise contrast of i^{th} parameters with j^{th} parameters produces a square matrix $PAR_{n \times n}$ where PAR_{ij} indicates the relative significance of parameters i in comparison to parameters j . In this matrix, $PAR_{ij} = 1$ when $i = j$ and $PAR_{ji} = 1/ PAR_{ij}$.

$$PAR_{n \times n} = \begin{array}{c|cccccc} \text{Parameters} & PAR_1 & PAR_2 & PAR_3 & \cdots & PAR_{(n-1)} & PAR_n \\ \hline PAR_1 & 1 & par_{12} & par_{13} & \cdots & par_{1(n-1)} & par_{1n} \\ PAR_2 & par_{21} & 1 & par_{23} & \cdots & par_{2(n-1)} & par_{2n} \\ \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots \\ PAR_{n-1} & par_{(n-1)1} & par_{(n-1)2} & par_{(n-1)3} & \cdots & 1 & par_{(n-1)n} \\ PAR_n & par_n1 & par_n2 & par_n3 & \cdots & par_n(n-1) & 1 \end{array}$$

Fig. 4.2 Comparison matrix

Table 4.5 The scale of the relative significance

Values	Level of importance
1	Similar Significance
3	Moderate Significance
5	High Significance
7	Very high Significance
9	Absolute Significance
2,4,6,8	For negotiating the values

Step 3(ii): Calculate the Weight (WT_j) of all parameters as follows: (i) the geometric mean of the i^{th} row is calculated (Eq. (4.8)), and (ii) the geometric means of the rows in comparison matrix are normalized in Eq. (4.9).

$$GMean_j = \left[\prod_{j=1}^n PAR_{ij} \right]^{1/n} \quad (4.8)$$

$$WT_j = GMean_j / \sum_{j=1}^n GMean_j \quad (4.9)$$

Step 3(iii): Calculate Normalized PAR Matrix as shown in Eq. (4.10).

$$NP_{n \times 1} = PAR_{n \times n} * WT_{n \times 1} \quad (4.10)$$

Step 3(iv): Calculate Relative Normalized PAR Matrix as shown in Eq. (4.11).

$$RNP_{n \times 1} = \frac{NP_{n \times 1}}{WT_{n \times 1}} \quad (4.11)$$

Step 3(v): Define largest Eigen value λ_{\max} which is calculated by finding the mean of matrix $RNP_{n \times 1}$ and compute the Consistency Index (CI) (Eq. (4.12)). The greater is the CI, the greater it deviates from consistency. Therefore, it should be less.

$$CI = \frac{(\lambda_{\max} - n)}{(n - 1)} \quad (4.12)$$

Step 3(vi): Find the RI (Random Index) for the used number of attributes in the process of decision making (Table 4.6).

Step 3(vii): Find the ratio of consistency (CR) as shown in Eq. (4.13). CR = 0.1 or less is satisfactory.

$$CR = \frac{\text{Consistency Index}}{\text{Random Index}} \quad (4.13)$$

Table 4.6 Random index values

Attributes	Values of Random Index
3	0.52
4	0.89
5	1.11
6	1.25
7	1.35
8	1.4
9	1.45
10	1.49

Step 4: Construct the weighted normalized decision matrix as shown in Eq. (4.14) and Eq. (4.15).

$$D = (d_{m \times n}) = (wt_j NVMP)_{m \times n} \quad (4.14)$$

$i = 1, 2, \dots, m$ & $j = 1, 2, \dots, n$, where

$$wt_j = WT_j / \sum_{j=1}^n WT_j \quad (4.15)$$

$j = 1, 2, \dots, n$ so that $\sum_{j=1}^n WT_j = 1$, where WT_j represents the original weight assigned to the criteria.

Step 5: Obtain the Ideal/Optimal alternative (Al_I) and the Non-Ideal/Non Optimal alternative (Al_N) for every criterion as shown in Eq. (4.16) and Eq. (4.17).

$$Al_I = \left\{ \left(\min(d_{ij} | i = 1, 2, \dots, m) | j \in J_- \right), \left(\max(d_{ij} | i = 1, 2, \dots, m) | j \in J_+ \right) \right\} \equiv \{d_{Ij} | j = 1, 2, \dots, n\} \quad (4.16)$$

$$Al_N = \left\{ \left(\max(d_{ij} | i = 1, 2, \dots, m) | j \in J_- \right), \left(\min(d_{ij} | i = 1, 2, \dots, m) | j \in J_+ \right) \right\} \equiv \{d_{Nj} | j = 1, 2, \dots, n\} \quad (4.17)$$

where,

$$J_+ = \left\{ j = 1, 2, \dots, n \mid \begin{array}{l} j \text{ is related to positively} \\ \text{impacting criteria} \end{array} \right\}$$

$$J_- = \left\{ j = 1, 2, \dots, n \mid \begin{array}{l} j \text{ is related to negatively} \\ \text{impacting criteria} \end{array} \right\}$$

Step 6: Calculate the separation measures. The separation of each alternative from the ideal one is represented in Eq. (4.18) and Eq. (4.19):

$$Se_{i+} = \left\{ \sum_{j=1}^n (d_{ij} - d_{Ij})^2 \right\}^{0.5} \quad (4.18)$$

$$Se_{i-} = \left\{ \sum_{j=1}^n (d_{ij} - d_{Nj})^2 \right\}^{0.5} \quad (4.19)$$

The values are $i= 1, 2, \dots, m$ & $j = 1, 2, \dots, n$.

Step 7: The comparative nearness of a specific alternative to the optimal/ideal solution is called as Opt_Host_i and represented in Eq. (4.20).

$$Opt_Host_i = \frac{Se_{i-}}{(Se_{i+} + Se_{i-})} \quad (4.20)$$

The value of $i=1, 2, \dots, m$.

Step 8: Rank the choices with respect to Opt_Host_i ($i = 1, 2, \dots, m$), where Opt_Host_i signifies the efficiency of an i th host. Thus, the ITEED method ensures that the selected host is least distance (Euclidean distance) apart from the optimal solution and the maximum distance apart from the non-optimal solution

4.2.4 Host underload detection algorithm

Most of the works in literature do not consider multiple criteria for detecting underloaded hosts. This thesis presents multi-criteria based underload detection algorithm. In this phase of VM consolidation, under-loaded hosts are detected according to the proposed ITUDA. ITUDA benefits from Improved TOPSIS as a MCDM method considering the parameters shown in Table 4.7. The procedure of ITUDA is same as of ITEED. However, RAM and MIPS are cost type criteria in ITUDA where as they are benefit type criteria in ITEED.

Table 4.7 Description of parameters used in ITUDA

Parameter	Description
Available RAM Capacity	The PM should have the minimum available RAM.
Available MIPS Capacity	The PM should have the minimum available MIPS capacity.
Number of VMs Assigned	The total number of VMs on the selected host should be least
Delay caused by Migration	The delay caused by VM on allocating it to selected host (PM) should be minimum

Chapter 5

Evaluation of proposed framework via case study/scenarios

This Chapter presents experimental results and evaluation of framework using various scenarios and a case study. The results of the evaluation of the framework are explained w.r.t various identified scenarios as shown in Fig. 5.1. The proposed algorithms are compared with benchmark algorithms found in literature and those works which have divided the energy-efficient management of resources in four phases as done in this study. The algorithms used for comparison are NPA [33], DVFS [124], MAD-MMT-2.5 [58], MAD-MMT-1.5 [58], IQR-MMT-1.5 [58], THR-MMT-1.0 [58], THR-MMT-0.8 [58], K-Means Clustering Algorithm-Average-Interquartile Range-Minimum Memory Size (KAI-MMS-1.0) [125], K-Means Clustering Algorithm-Average-Median Absolute Deviation- Minimum Memory Size (KAM-MMS-2.0) [125], K-Means clustering algorithm-Midrange-Interquartile range-Maximum ratio of CPU utilization to memory utilization (KMI-MRCU-1.0) [99] and K-Means clustering algorithm-Midrange-Interquartile range-Minimum the product of a CPU utilization and a memory utilization (KMI-MPCU-2.0) [99].

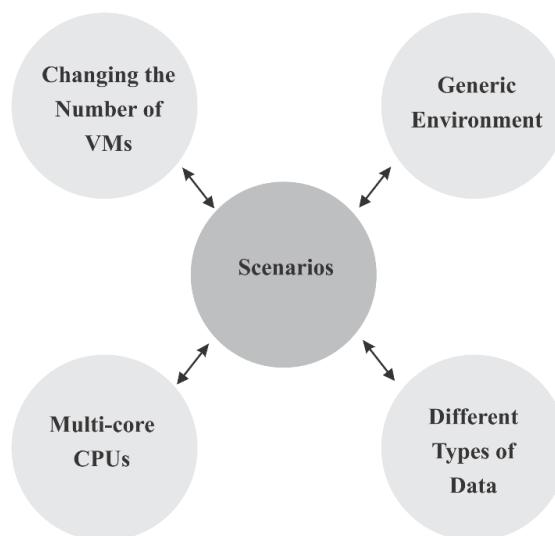


Fig. 5.1 Identified scenarios

The case study is presented to validate the proposed ITEED algorithm. The experiments are done repetitively and average results are reported.

The safety parameter i.e. L is to be determined for KMIMR policy for different workloads. In the conducted experiments, value of L is varied from 0.5 to 3.0 with an interval of 0.5 between them. The proposed framework using KMIMR, ITUDA, MRCPMB, ITEED for CPU intensive tasks and KMIMR, ITUDA, MPCBM, ITEED for I/O tasks is represented by K-MRCPMB- L and K-MPCBM- L respectively.

5.1 Checking whether the framework is generic

The first scenario is to validate whether the proposed framework is generic or not. This can be done by evaluating the framework for both CPU and I/O intensive tasks. The workload “20110303” and “20110322” in Planet Lab dataset are considered to be CPU intensive and I/O intensive respectively [99]. If the framework performs well for both CPU intensive and I/O intensive tasks, then it is a generic framework otherwise not.

5.1.1 Analysing the framework for CPU intensive tasks

The algorithm used for VM migration in case of processing of CPU intensive tasks is MRCPMB. Fig. 5.2 - 5.7 show the results of the experiments to evaluate the proposed framework conducted using “20103303” workload under different values of L .

Fig. 5.2 – Fig. 5.7 show the values of energy efficiency, energy consumption, SLA violations, SLATAH, PDM and number of VM migrations respectively using different L values. The minimum value of energy consumption is obtained when value of L is 1.0 as shown in Fig. 5.3. On the other hand, the minimum value of SLA violation is obtained under L value of 2.0.

The parameter i.e. energy efficiency which considers both energy consumption and SLA violation is found to be maximum under L value of 1.0. Therefore, the optimum algorithm is considered to be K-MRCPMB-1.0 and is used for comparison with other benchmark algorithms.

The results of comparison with other baseline policies (NPA, DVFS, MAD-MMT-2.5, IQR-MMT-1.5, THR-MMT-1.0, THR-MMT-0.8, KAI-MMS-1.0, KAM-MMS-2.0, and KMI-MRCU-1.0) are shown in Table 5.1.

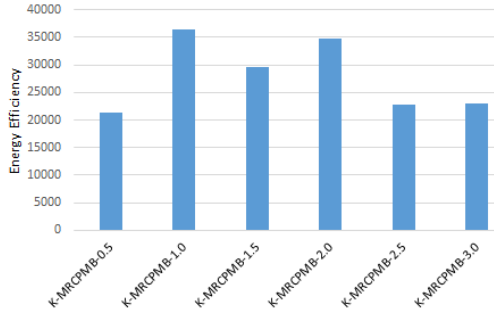


Fig. 5.2 Energy efficiency under diverse L values

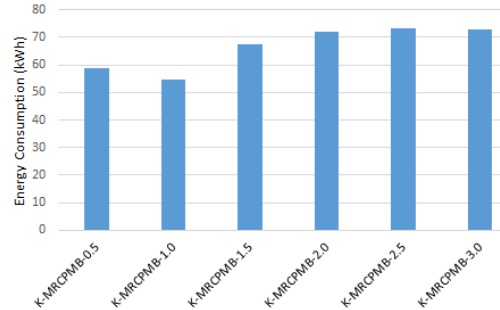


Fig. 5.3 Energy consumption under diverse L values

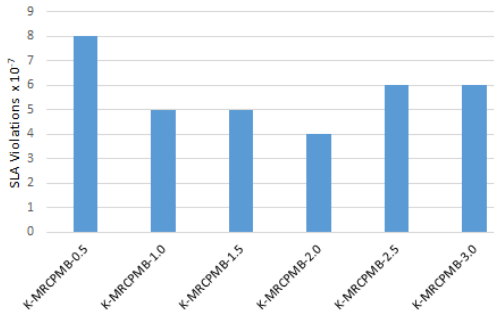


Fig. 5.4 SLA violations under diverse L values

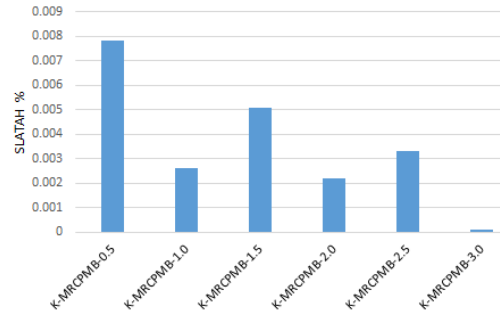


Fig. 5.5 SLATAH under diverse L values

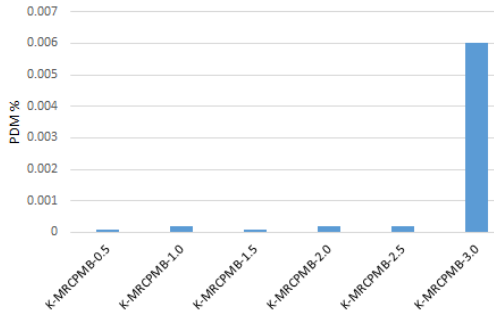


Fig. 5.6 PDM under diverse L values

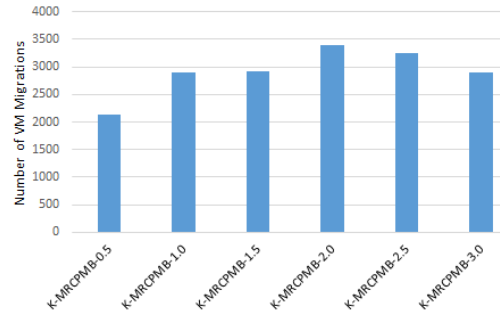


Fig. 5.7 Number of VM migrations under diverse L values

The comparison is done on the basis of parameters which include energy efficiency, energy consumption, SLA violations, SLATAH, PDM and number of VM migrations. The least values of all parameters are favourable except the value of energy efficiency.

Table 5.1 Results of comparison of K-MRCPMB-1.0 with other policies

Policies	Energy efficiency	Energy consumption (Kwh)	SLA violations (10^{-7})	SLATAH (%)	PDM (%)	Number of VM migrations
NPA	--	2410.8	--	--	--	--
DVFS	--	803.91	--	--	--	--
MAD-MMT-2.5	169	114.27	518	5.24	0.10	25923
IQR-MMT-1.5	166	117.08	514	5.08	0.10	26420
THR-MMT-1.0	38	99.95	2613	26.97	0.10	19852
THR-MMT-0.8	170	119.40	492	4.99	0.10	26567
KAI-MMS-1.0	6393	104.28	15	2.03	0.01	7519
KAM-MMS-2.0	9231	83.33	13	1.73	0.01	6808
KMI-MRCU-1.0	28484	87.77	4	0.90	0.004	2821
K-MRCPMB-1.0	36476	54.831	5	0.0026	0.0002	2890

The more the value of energy efficiency, the better it is. NPA policy does not use any energy-efficient procedure like VM migrations etc. Thus, there is nonexistence of energy efficiency and other metrics denoted by "--". DVFS uses techniques for scaling up/down voltage and frequency dynamically to cope with changing power needs. DVFS is better than NPA consuming 803.91 Kwh of energy, whereas the energy consumption of NPA is 2410.8 Kwh. K-MRCPMB-1.0 decreased the energy consumption by more than 45% and SLA violations by more than 95% in comparison to MAD-MMT-2.5, IQR-MMT-1.5, THR-MMT-1.0, THR-MMT-0.8 algorithms leading to a very high amount of energy efficiency. This can be explained by the fact that all these algorithms (MAD-MMT-2.5, IQR-MMT-1.5, THR-MMT-1.0, THR-MMT-0.8) use either a fixed value of threshold or perform statistical analysis of historical data and they only consider single criteria i.e. CPU utilization.

Moreover, their focus was just to decrease energy consumption. K-MRCPMB-1.0, on the other hand uses multiple criteria and it focuses on increasing energy efficiency.

In comparison to KAI-MMS-1.0 and KAM-MMS-2.0, the proposed scheme increases the energy efficiency by more than 70%, decreases the energy consumption

and SLA violations by more than 40% and 60% respectively. In comparison to KMI-MRCU-1.0, the presented algorithm increases the energy efficiency by more than 20%, reduces energy consumption by more than 35%. However, there is increase in SLA violations by 20%. Still, the overall energy efficiency is high in case of K-MRCPMB-1.0.

The probable reason of K-MRCPMB-1.0 being a more efficient algorithm can be that it uses K-medoids algorithm before statistical analysis of historical data to find adaptive threshold. Further, VM migration is performed considering multiple parameters related to CPU, bandwidth and memory. Also it uses a multi-criteria based VM placement and underload detection algorithm in contrast to single criteria based approaches. Its focus is not just to reduce energy consumption but also to increase energy efficiency.

5.1.2 Analysing the framework for I/O intensive tasks

The algorithm used for VM migration in case of processing of I/O intensive tasks is MPCBM. Fig. 5.8 - 5.13 show the results of experiments to evaluate the proposed framework conducted using “20110322” workload under different values of L. It is well clear from the Fig. 5.9 and 5.10 that minimum value of energy consumption and SLA violations is obtained under L value of 1.5 and 0.5 respectively. Considering the highest value of energy efficiency which is found under L value of 1.5. Thus, K-MPCBM-1.5 is considered to be an optimum algorithm and is used for comparison with others. The results of comparison of K-MPCBM-1.5 with other baseline policies (NPA, DVFS, MAD-MMT-2.5, IQR-MMT-1.5, THR-MMT-1.0, THR-MMT-0.8, KAI-MMS-1.0, KAM-MMS-2.0, and KMI-MRCU-1.0) are shown in Table 5.2. The parameters such as energy efficiency, energy consumption, SLA violations, SLATAH, PDM and number of VM migrations are used for comparison. The energy efficiency should be as much high as possible and values of all other parameters should be least. NPA policy does not use any energy-efficient procedure like migrations of VM etc.

Thus, there is nonexistence of energy efficiency and other metrics which is represented by symbol “-”. DVFS uses techniques for dynamically scaling up/down voltage and frequency to cope with changing power needs.

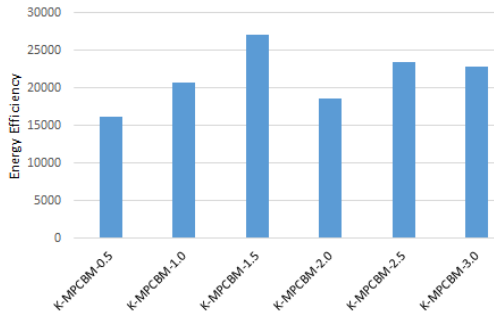


Fig. 5.8 Energy efficiency under diverse L values

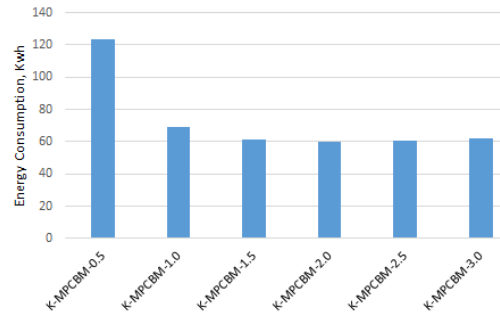


Fig. 5.9 Energy consumption under diverse L values

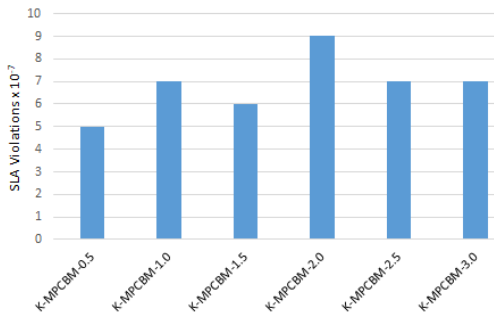


Fig. 5.10 SLA violations under diverse L values

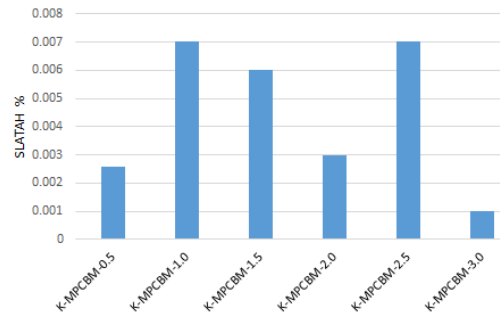


Fig. 5.11 SLATAH under diverse L values

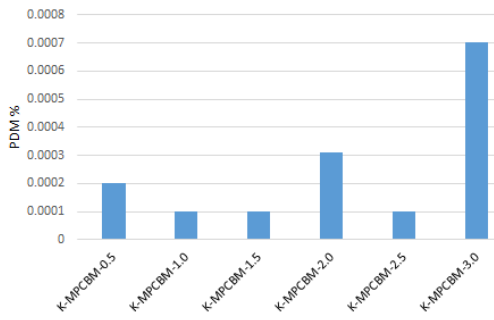


Fig. 5.12 PDM under diverse L values

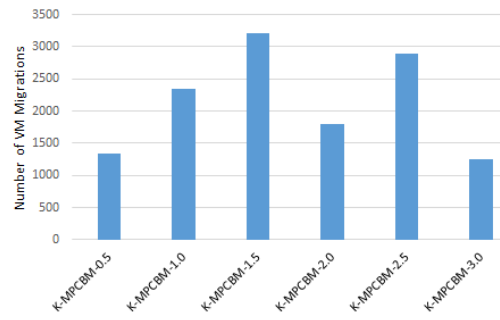


Fig. 5.13 Number of VM migrations under diverse L values

Thus, DVFS is better than NPA. In comparison to MAD-MMT-2.5, IQR-MMT-1.5, THR-MMT-1.0, THR-MMT-0.8 algorithms, K-MPCBM-1.5 increased the energy efficiency by more than 90%, decreased the energy consumption by more than 45% and SLA violations by more than 90%. This can be explained by the fact that all these algorithms (MAD-MMT-2.5, IQR-MMT-1.5, THR-MMT-1.0, THR-MMT-0.8) use either a fixed value of threshold or perform statistical analysis of historical data and they only consider single criteria i.e. CPU utilization. Moreover, their focus was just to decrease energy consumption. K-MPCBM-1.5, on the other

hand uses multiple criteria and it focuses on increasing energy efficiency. K-MPCBM-1.5 increased the energy efficiency by more than 60%, reduced energy consumption and SLA violations by more than 40% and 60% respectively in comparison with KAI-MMS-1.0 and KAM-MMS-2.0. In comparison to KMI-MPCU-1.0, K-MPCBM-1.5 increased the energy efficiency by more than 75%, reduces energy consumption and SLA violations by more than 8% and 75% respectively.

Table 5.2 Results of comparison of K-MPCBM-1.5 with other policies

Policies	Energy efficiency	Energy consumption (Kwh)	SLA violations (10^{-7})	SLATAH (%)	PDM (%)	Number of VM migrations
NPA	--	2410.8	--	--	--	--
DVFS	--	803.91	--	--	--	--
MAD-MMT-2.5	169	114.27	518	5.24	0.10	25923
IQR-MMT-1.5	166	117.08	514	5.08	0.10	26420
THR-MMT-1.0	38	99.95	2613	26.97	0.10	19852
THR-MMT-0.8	170	119.40	492	4.99	0.10	26567
KAI-MMS-1.0	6393	104.28	15	2.03	0.01	7519
KAM-MMS-2.0	9231	83.33	13	1.73	0.01	6808
KMI-MPCU-1.0	5694	67.55	26	2.90	0.01	12607
K-MPCBM-1.5	27039	61.64	6	0.006	0.0001	3214

The probable reason of K-MPCBM-1.5 being a more efficient algorithm can be that it uses K-medoids algorithm before statistical analysis of historical data to find adaptive threshold. Further, VM migration is performed considering multiple parameters related to CPU, bandwidth and memory. Also, it uses a multi-criteria based VM placement and underload detection algorithm in contrast to single criteria based approaches.

5.2 Evaluating the framework using different number of VMs

The second scenario is to validate whether the proposed framework works for diverse number of VMs. This can be done by evaluating the framework for workloads having diverse number of VMs. The different workloads in Planet Lab dataset contain data of diverse number of VMs as discussed in Chapter 4. The framework is tested for workload “20110303” containing 1052 VMs and for workload

“20110322” containing 1516 VMs. The results are already shown in Table 5.1 for 1052 VMs and Table 5.2 for 1516 VMs.

5.3 Evaluating the framework for different types of data

The system is tested for both real and synthetic data. The results of execution on real data (Planet Lab workload) are already shown. The generated synthetic data contains random values of CPU utilization. The results of execution on synthetic dataset are shown in Table 5.3. It is well clear from the results that K-MRCPMB-1.0 and K-MPCBM-1.5 outperformed all other algorithms in case of synthetic dataset also.

Table 5.3 Results of comparison of K-MRCPMB-1.0, K-MPCBM-1.5 with other policies

Policies	Energy efficiency	Energy consumption (Kwh)	SLA violations (10^{-7})	SLATAH (%)	PDM (%)	Number of VM migrations
MAD-MMT-2.5	1.658	12.06	0.05	1	0.05	1267
IQR-MMT-1.5	1.311	12.71	0.06	0.2	0.3	1271
THR-MMT-0.8	0.936	13.35	0.08	0.4	0.2	1836
KAI-MMS-1.0	0.739	6.76	0.2	0.1	0.2	127
KAM-MMS-2.0	2.666	7.5	0.05	0.2	0.25	142
KMI-MRCU-1.0	1.831	7.8	0.07	0.25	0.3	182
K-MRCPMB-1.0	7.802	3.56	0.036	0.2	0.18	172
K-MPCBM-1.5	5.779	4.12	0.042	0.7	0.6	167

It is because of use of K-medoids algorithm before statistical analysis of historical data to find adaptive threshold. Further, VM migration is performed considering multiple parameters related to CPU, bandwidth and memory. Also it uses a multi-criteria based VM placement and underload detection algorithm in contrast to single criteria based approaches. Its focus is not just to reduce energy consumption but also to increase energy efficiency.

5.4 Evaluating the framework using multi-core CPUs

Simulation of real multi-core configuration based servers is done to evaluate the behaviour of framework. Two types of servers are used - HP ProLiant ML110 G4

(Intel Xeon 3040, 2 cores _ 1860 MHz, 4 GB), HP ProLiant ML110 G5 (Intel Xeon 3075, 2 cores _ 2660 MHz, 4 GB) in all the experiments conducted. Therefore, the results of execution of algorithms using multi-core servers is already shown in Section 5.1 and 5.3.

5.5 Case study to evaluate the performance of proposed VM placement algorithm

This case study selects the target host for VM deployment. The target host is found using the proposed multi-criteria based ITEED algorithm which makes use of Improved TOPSIS method as explained in Chapter 4. Although many scheduling approaches are proposed in the literature [45–47], yet they have limited applicability to dynamic cloud computing environment as they do not consider multiple criteria. In this case study, ITEED is used to find the optimal host for VM deployment. In this process, compliance is generated considering five parameters. A sample dataset generated from Planet Lab data of CPU utilization and real cloud VM and host configurations is used in this case study. The data center has 800 heterogeneous hosts having 400 HP ProLiant ML110 G4 and 400 HP ProLiant ML110 G5. Four types of VMs that correspond to Amazon EC2 instances are used. Power models employed are taken from SPEC power benchmark [117].

Throughout the simulations, each VM is arbitrarily given a workload trace from one of the VMs from the corresponding day. For the demonstration of results, sample dataset comprises of performance of 40 hosts accessed on five parameters. The parameters are available capacity in terms of RAM (AC RAM), available capacity in terms of MIPS (AC MIPS), number of VMs allocated on a host, migration delay, an increase in power consumption. AHP method is used to calculate relative weights that are to be assigned to these parameters. The parameters- AC RAM and AC MIPS are presumed to be beneficial attributes in the case study. From the dataset generated by using Planet Lab workload trace and real hosts and VMs configuration, the normalized decision matrix (Table 5.4) is formed. It is 40 x 5 matrix, which represents 40 physical hosts on five attributes. The relative significance among parameters is given in Table 5.5. The steps described in Chapter 4 are used to calculate relative normalized weights of all attributes.

Table 5.4 Normalized decision matrix

Physical Host	AC RAM	AC MIPS	Number of VMs Allocated	Migration Delay	Increase in Power Consumption
PH_1	0.0620	0.1799	0.0576	0.0265	0.0183
PH_2	0.0508	0.1138	0.1153	0.0596	0.0356
PH_3	0.0396	0.0476	0.1729	0.1022	0.0356
PH_4	0.0421	0.1138	0.0576	0.0735	0.0183
PH_5	0.0309	0.0476	0.1153	0.1124	0.0183
PH_201	0.0620	0.2858	0.0576	0.0265	0.0178
PH_204	0.0171	0.0212	0.2882	0.2384	0.0394
PH_207	0.0110	0.0873	0.1153	0.1653	0.0394
PH_209	0.0197	0.0873	0.1729	0.1625	0.0178
PH_210	0.0084	0.0212	0.2306	0.2294	0.0394
PH_211	0.0046	0.0873	0.0576	0.1620	0.0178
PH_212	0.0577	0.0212	0.0576	0.0367	0.0178
PH_401	0.1353	0.3219	0.0576	0.0265	0.0275
PH_402	0.1241	0.2557	0.1153	0.0596	0.0275
PH_403	0.1129	0.1896	0.1729	0.1022	0.0275
PH_404	0.1016	0.1234	0.2306	0.1589	0.0577
PH_405	0.0904	0.0573	0.2882	0.2384	0.0275
PH_406	0.1154	0.2557	0.0576	0.0735	0.0275
PH_407	0.0843	0.1234	0.1153	0.1653	0.0275
PH_408	0.0778	0.1234	0.0576	0.1620	0.0275
PH_409	0.1310	0.0573	0.0576	0.0367	0.0577
PH_410	0.1042	0.1896	0.1153	0.1124	0.0577
PH_411	0.0929	0.1234	0.1729	0.1625	0.0275
PH_412	0.0817	0.0573	0.2306	0.2294	0.0577
PH_413	0.0730	0.0573	0.1729	0.2229	0.0275
PH_414	0.0666	0.0573	0.1153	0.2121	0.0275
PH_601	0.2819	0.3396	0.0576	0.0265	0.2137
PH_602	0.2706	0.2735	0.1153	0.0596	0.2137
PH_603	0.2594	0.2073	0.1729	0.1022	0.2137
PH_604	0.2482	0.1412	0.2306	0.1589	0.3216
PH_605	0.2369	0.0750	0.2882	0.2384	0.2137
PH_606	0.2369	0.0750	0.3458	0.2980	0.3216
PH_607	0.2619	0.2735	0.0576	0.0735	0.2137
PH_608	0.2308	0.1412	0.1153	0.1653	0.2137
PH_609	0.1997	0.0089	0.1729	0.2833	0.2137
PH_610	0.2244	0.1412	0.0576	0.1620	0.2137
PH_611	0.2775	0.0750	0.0576	0.0367	0.2137
PH_612	0.1832	0.0089	0.0576	0.2592	0.5104
PH_613	0.2507	0.2073	0.1153	0.1124	0.2137
PH_614	0.2395	0.1412	0.1729	0.1625	0.2137

The relative normalized weights of parameters are AC RAM = 0.1786, AC MIPS = 0.2143, number of VMs allocated = 0.1786, migration delay = 0.1071, increase in power consumption = 0.3214. The value of λ_{\max} = 5.4222, CI = 0.1056, and CR=0.0950, which is smaller than the allowed CR value of 0.1. It denotes the worthy reliability in the judgments for evaluating the weights of attributes. According to the normalized weights, effective host for VM deployment is found. They are

based on the ranks generated according to compliance of parameters, as shown in Table 5.6.

Table 5.5 Relative importance of parameters

	AC RAM	AC MIPS	Number of VMs Allo- cated	Migration Delay	Increase in Power Con- sumption
AC RAM	1.00	0.83	1.00	1.67	0.56
AC MIPS	1.20	1.00	1.20	2.00	0.67
Number of VMs Allo- cated	1.00	0.83	1.00	1.67	0.56
Migration Delay	0.60	0.50	0.60	1.00	0.33
Increase in Power Con- sumption	1.80	1.50	1.80	3.00	1.00

Table 5.6 Ranks generated according to compliance

Physical Host	Rank (Normal- ized)	Rank	Physical Host	Rank (Normal- ized)	Rank
PH_1	0.8820	0.7687	PH_409	0.8053	0.7018
PH_2	0.8195	0.7143	PH_410	0.8765	0.7639
PH_3	0.7627	0.6647	PH_411	0.8267	0.7205
PH_4	0.8303	0.7236	PH_412	0.7425	0.6471
PH_5	0.7784	0.6784	PH_413	0.7735	0.6741
PH_201	0.9343	0.8143	PH_414	0.7831	0.6825
PH_204	0.7041	0.6137	PH_601	0.7944	0.6923
PH_207	0.7785	0.6785	PH_602	0.7632	0.6652
PH_209	0.7829	0.6823	PH_603	0.7146	0.6228
PH_210	0.7157	0.6238	PH_604	0.4913	0.4282
PH_211	0.7941	0.6921	PH_605	0.5976	0.5208
PH_212	0.7817	0.6813	PH_606	0.4241	0.3696
PH_401	1.0000	0.8715	PH_607	0.7711	0.6721
PH_402	0.9524	0.8300	PH_608	0.6836	0.5958
PH_403	0.8834	0.7699	PH_609	0.5854	0.5102
PH_404	0.7940	0.6920	PH_610	0.6944	0.6052
PH_405	0.7473	0.6513	PH_611	0.6792	0.5919
PH_406	0.9548	0.8321	PH_612	0.2940	0.2562
PH_407	0.8357	0.7283	PH_613	0.7271	0.6337
PH_408	0.8406	0.7326	PH_614	0.6710	0.5848

From the Table 5.6, it is evident that PH_401 is evaluated to be the most effective host for VM deployment with compliance of parameters (AC RAM = 0.1353, AC MIPS = 0.3219, number of VMs allocated = 0.0576, migration delay = 0.0265, and increase in power consumption = 0.0275). PH_612 is assessed to be the most non-optimal host with compliance of parameters (AC RAM = 0.1832, AC MIPS = 0.0089, number of VMs allocated = 0.0576, migration delay = 0.2592, and increase in power consumption = 0.5104). A simulation run of 800 hosts for the data center

is performed. Improved TOPSIS method evaluates the compliance of hosts as described in Chapter 4. The efficiency of hosts is classified in Table 5.7.

Table 5.7 Efficiency classification

S. no.	Efficiency Class	Value
1	Extremely Energy-Efficient	$\geq .8$
2	Energy-Efficient	.7999- .6
3	Moderately Energy-Efficient	.5999 - .4
4	Base Line Energy-Efficient	.3999 - .2
5	Minimal Energy-Efficient	$< .2$

Fig. 5.14 illustrates the evaluation of physical hosts on various parameters. Fig. 5.15 presents the relative importance of parameters utilized in this study of host selection. Figure 5.16 indicates the comparative significance of parameter AC RAM, relative to other parameters (AC MIPS, number of VMs allocated, migration delay, increase in power consumption) is (0.83, 1.00, 1.67, and 0.56). Similarly, the comparative significance of parameter, increase in power consumption compared to other parameters (AC RAM, AC MIPS, number of VMs allocated, and migration delay) is (1.80, 1.50, 1.80, and 3.00).

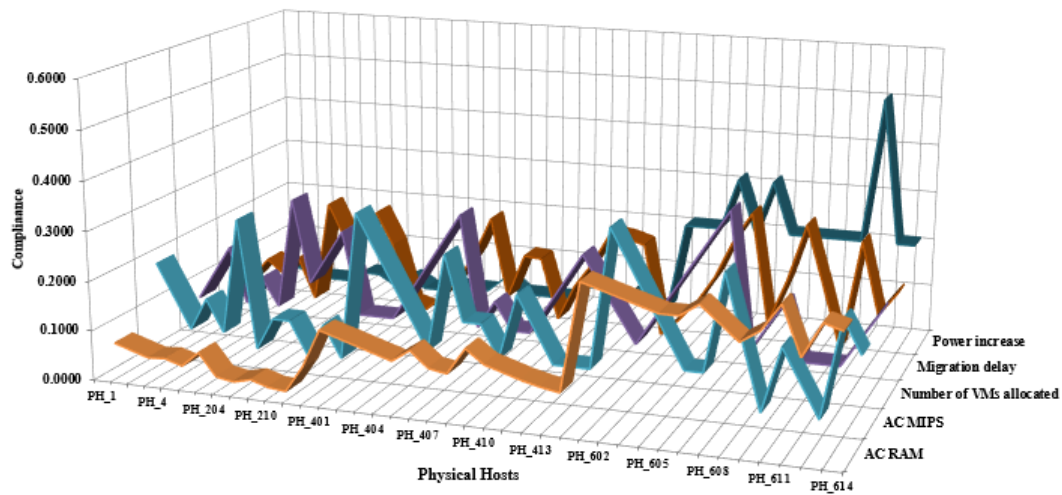


Fig. 5.14 Compliance of the various attributes for physical hosts

Fig 5.16 shows the normalized comparative weights for parameters, increase in power consumption has the maximum relative normalized weight (increase in power consumption = 0.3214). In comparison, migration delay has the minimum relative normalized weight (migration delay = 0.1071).

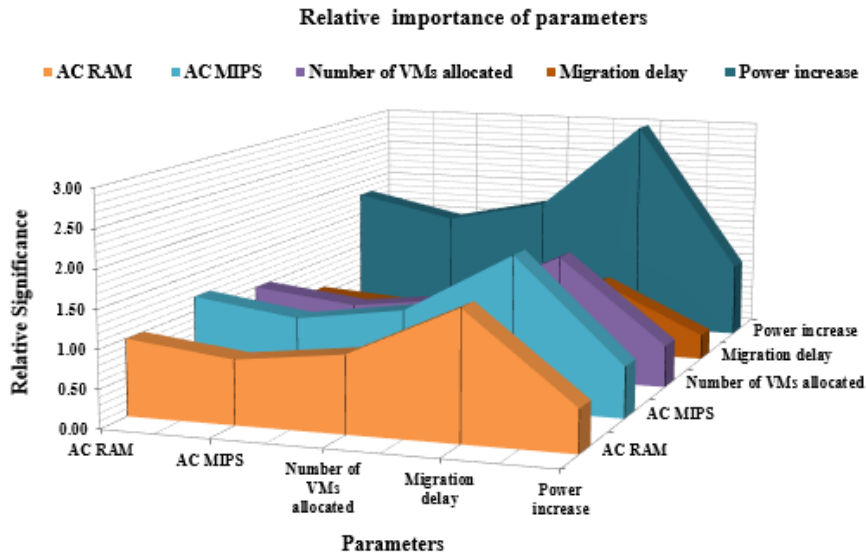


Fig. 5.15 Relative significance of the parameters for physical hosts

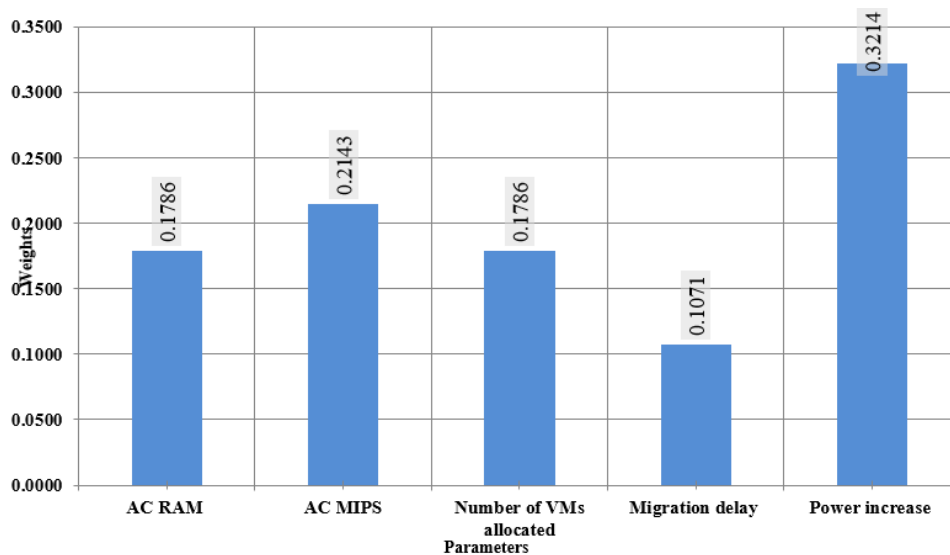


Fig. 5.16 Relative normalized weights of the parameters for physical hosts

In other words, increase in power consumption has been considered the most significant parameter and migration delay as the least significant parameter.

Fig. 5.17 shows the ranks of physical hosts according to compliance (Fig. 5.14) and the normalized comparative weights (Fig. 5.16). It is seen that PH_401 is evaluated to be the most effective host. In contrast, PH_612 is evaluated to be the least effective host for VM placement. The grouping of hosts concerning the classification pattern given in Table 5.7 is illustrated in Table 5.8.

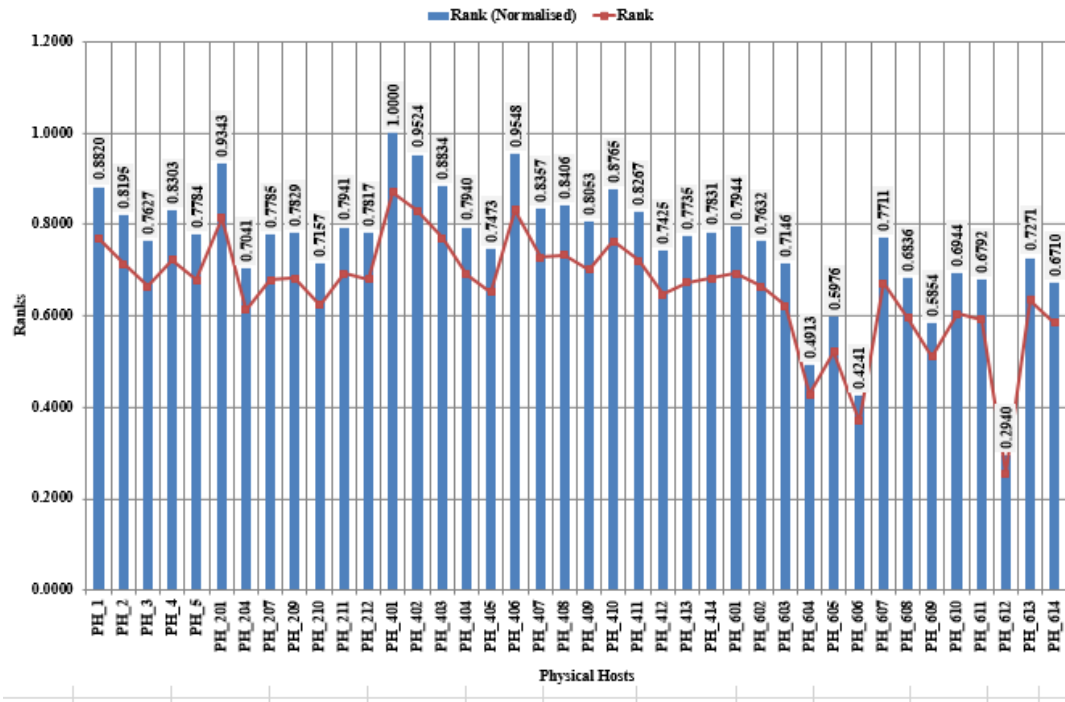


Fig. 5.17 The efficiency of the hosts generated according to ranks

Table 5.8 Hosts classification according to efficiency class

S. no.	Efficiency Class	Value	Hosts
1	Extremely Energy-Efficient	$\geq .8$	PH_401, PH_406, PH_402, PH_201
2	Energy-Efficient	.7999 - .6	PH_403, PH_1, PH_410, PH_408, PH_407, PH_4, PH_411, PH_2, PH_409, PH_601, PH_211, PH_404, PH_414, PH_209, PH_212, PH_207, PH_5, PH_413, PH_607, PH_602, PH_3, PH_405, PH_412, PH_613, PH_210, PH_603, PH_204, PH_610
3	Moderately Energy-Efficient	.5999 - .4	PH_608, PH_611, PH_614, PH_605, PH_609, PH_604
4	Base Line Energy-Efficient	.3999 - .2	PH_606, PH_612
5	Least Energy-Efficient	$< .2$	--

The correctness of framework is verified from: (i) it uses TOPSIS-a standard MCDM technique, (ii) calculated weights using AHP which reduces biasness, (iii) considered parameters related to CPU, memory, network and multiple criteria i.e. energy efficiency and SLA efficiency. Results are validated as ranking of hosts are done based on a real dataset, real hosts and VMs' configuration because case study is an in-depth, detailed, qualitative validation of proposed work representing the living reality [56]. Moreover, the case study approach is widely used in literature [57-59] for validation of the proposed work.

Chapter 6

Conclusion and Future Scope

Cloud computing is one of the most prominent computing paradigms in this world of information technology. The issue of increasing energy consumption by cloud data centers must be given utmost priority. The servers hosted in these data centers consume tremendous amount of power and release harmful gases to the atmosphere. To ensure sustainability, energy-efficient utilization of resources is mandatory.

Energy consumption depends upon resource utilization. Over-loaded hosts consume more power and degrade performance. On the other hand, under-loaded hosts consume power at leisure. There should be optimum utilization of resources to maintain performance. Although there are many energy-efficient cloud computing frameworks, but they do not focus on maintaining QoS simultaneously. The focus of researchers should not be only on reducing energy consumption but should also be on reducing violations in SLA.

In this work, an effort is made to develop an energy-efficient VM consolidation framework that optimizes the resource utilization by detecting under-utilized, over-utilized hosts, selecting VMs from them and finding new target hosts for their placements. This results in switching-off the idle hosts leading to lesser energy consumption. The SLA related metrics are also defined in this study. The framework is evaluated for both energy consumption and SLA related metrics.

This work presents novel algorithms for all phases of VM consolidation. The presented framework is generic as it works for both CPU intensive and I/O intensive tasks. The framework adapts well to the dynamic cloud environment as it uses adaptive thresholds for detection of over-loaded hosts and a multi-criteria based host underload detection algorithm. This thesis presents the design and simulation of QoS-aware energy-efficient VM consolidation framework for optimal resource utilization in cloud data centers. The flow diagram illustrates the working of the framework by using algorithms for all the phases. It solves the resource allocation problem for both under-loaded and over-loaded hosts in one step rather than separate allocation. The proposed framework provides a holistic view to solve the problem.

The unique features of the proposed framework which makes it different from other frameworks are:

- It is a generic framework as it considers both CPU and I/O intensive tasks.
- It relies upon adaptive threshold in order to suit dynamic cloud environment for detection of over-loaded hosts.
- The memory and network related parameters apart from CPU utilization are also considered during VM migration.
- A MCDM algorithm is used to address the problem of VM placement and detection of under-loaded hosts.
- It also considers SLA metrics for maintaining QoS.

The evaluation of framework was done using various scenarios and a case study. The results of the evaluation of the framework are compared with benchmark algorithms. The case study was done to validate the proposed ITEED algorithm.

Results of simulation using real workload traces on CloudSim platform show the efficiency of the proposed framework. It can be inferred from the results that some computations before statistical analysis of historical data to find adaptive threshold may prove to be useful in case of energy-efficiency. Further, considering multiple parameters related to CPU, bandwidth and memory during VM migration, VM placement, and host underload detection increases efficiency of the proposed framework.

As a future work, efforts can be made to implement the proposed framework on a real-cloud platform. Apart from this various green computing metrics like power usage effectiveness, data center infrastructure efficiency etc. can also be focused upon apart from energy consumption and SLA related metrics. The reduction of total cost of ownership is also another domain of future study. It would also be exciting to study resource management algorithms from the aspect of newly emerged edge and fog computing environments that may provide very low latency services to end users.

References

- [1] Parkhill D. The challenge of the computer utility. Reading, Mass.: Addison-Wesley; 1966.
- [2] Zhang Q, Cheng L, Boutaba R. Cloud computing: state-of-the-art and research challenges. *Journal of Internet Services and Applications*. 2010 May;1(1):7-18.
- [3] Singh A, Hemalatha M. Cloud computing for academic environment. *International Journal of Information and Communication Technology Research*. 2012 Feb;2(2):97-101.
- [4] García G, Espert B, García H. SLA-driven dynamic cloud resource management. *Future Generation Computer Systems*. 2014 Feb;31(1):1-11.
- [5] Buyya R, Yeo S, Venugopal S, Broberg J, Brandic I. Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. *Future Generation Computer Systems*. 2009 Jun;25(6):599-616.
- [6] Durao F, Carvalho F, Fonseka A, Garcia C. A systematic review on cloud computing. *The Journal of Supercomputing*. 2014 Jan;68(3):1321-1346.
- [7] Wang Z, Shuang K, Yang L, Yang F. Energy-aware and revenue-enhancing combinatorial scheduling in virtualized of cloud data center. *Journal of Convergence Information Technology*. 2012 Jan;7(1):62-70.
- [8] Fei, T. Management of data and scheduling of tasks on architecture distributtees [dissertation]. Centrale Paris; 2011.
- [9] Garg K, Buyya R. Green cloud computing and environmental sustainability. *Harnessing Green IT: Principles and Practices*. Wiley Online Library; 2012.
- [10] Sajid M, Raza Z. Cloud computing: issues & challenges. In *International Conference on Cloud, Big Data and Trust*. 2013 Dec (pp. 13-15).
- [11] Khattar N, Sidhu J, Singh J. Toward energy-efficient cloud computing: a survey of dynamic power management and heuristics-based optimization techniques. *The Journal of Supercomputing*. 2019 Aug;75(8):4750-4810.
- [12] Beloglazov A, Buyya R, Lee C, Zomaya A. A taxonomy and survey of energy-efficient data centers and cloud computing systems. *Advances in Computers*. 2011 Jul;82(1):47-111.
- [13] Zhang C, Chang E, Yap H. Tagged-MapReduce: A general framework for secure computing with mixed-sensitivity data on hybrid clouds. In *14th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*. 2014 May (pp. 31–40).
- [14] Watson P. A multi-level security model for partitioning workflows over federated clouds. *Journal of Cloud Computing: Advances, Systems and Applications*. 2012 Jul;1(1):1-15.

- [15] Sharif S, Taheri J, Zomaya Y, Nepal S. MPHIC: preserving privacy for workflow execution in hybrid clouds. In IEEE International Conference on Parallel and Distributed Computing, Applications and Technologies. 2013 Dec (pp. 272–280).
- [16] Luo J, Rao L, Liu X. Temporal load balancing with service delay guarantee for energy cost optimization in internet data centers. IEEE Transactions on Parallel and Distributed Systems. 2014 Mar;25(3):775–784.
- [17] Lee K, Kulkarni I, Pompili D, Parashar M. Proactive thermal management in green data centers. The Journal of Supercomputing. 2012 May;60(2):165–195.
- [18] Ma V, Kim J, Park S, Kim J, Jang J. An efficient session_weight load balancing and scheduling methodology for high-quality tele-healthcare service based on WebRTC. The Journal of Supercomputing. 2016 Oct;72(10):3909–3926.
- [19] Singh S, Sidhu J. Compliance-based multi-dimensional trust evaluation system for determining trustworthiness of cloud service providers. Future Generation Computer Systems. 2017 Feb;67(1):109–132.
- [20] Sidhu J, Singh S. Compliance based trustworthiness calculation mechanism in cloud environment. Procedia Computer Science. 2014 Jan;37:439–446.
- [21] Sidhu J, Singh S. A novel cloud auditor based trust management framework for cloud computing. International Journal of Grid and Utility Computing. 2016 Jan;7(3):219–235.
- [22] Xu G, Dai B, Huang B, Yang J, Wen S. Bandwidth-aware energy efficient flow scheduling with SDN in data center networks. Future Generation Computer Systems. 2017 Mar;68(1):163–174.
- [23] El-Boghdadi H. Power-aware routing for well-nested communications on the circuit switched tree. Journal of Parallel and Distributed Computing. 2009 Feb;69(2):135–142.
- [24] Faragardi R, Rajabi A, Shojaee R, Nolte T. Towards energy-aware resource scheduling to maximize reliability in cloud computing systems. In 10th IEEE International Conference on Embedded and Ubiquitous Computing and High-Performance Computing and Communications. 2013 Dec (pp. 1469–1479).
- [25] Tchernykh A, Lozano L, Bouvry P, Pecero E, Schwiegelshohn U, Nesmachnow S. Energy-aware online scheduling: ensuring quality of service for IaaS clouds. In IEEE International Conference on High-Performance Computing and Simulation. 2014 Jul (pp. 911–918).
- [26] Kaur T, Chana I. Energy efficiency techniques in cloud computing: a survey and taxonomy. ACM Computing Surveys. 2015 Nov;48(2):1–46.
- [27] Huber P, Mill P. Dig more coal—the PCs are coming. United States. 1999. Available from: <http://www.forbes.com/forbes/1999/0531/6311070a.html> [accessed 12.12.16].

- [28] Mobil E. The Outlook of Energy: A View to 2040. 2012. Available from: http://www.exxonmobil.com/corporate/files/news_pub_eo.pdf [accessed 01.12.16]
- [29] Mei L, Chan K, Tse H. A tale of clouds: Paradigm comparisons and some thoughts on research issues. In 3rd IEEE Asia-Pacific Services Computing Conference. 2008 Dec (pp. 464-469).
- [30] Baliga J, Ayre W, Hinton K, Tucker S. Green cloud computing: Balancing energy in processing, storage, and transport. In: Proceedings of the IEEE Computing Conference. 2011 Jan;99(1):149-167.
- [31] Netjinda N, Sirinaovakul B, Achalakul T. Cost optimal scheduling in IaaS for dependent workload with particle swarm optimization. The Journal of Supercomputing. 2014 Jun;68(3):1579–1603.
- [32] Avgerinou M, Bertoldi P, Castellazzi L. Trends in data centre energy consumption under the European code of conduct for data centre energy efficiency. Energies. 2017 Sep;10(10):1470.
- [33] Uchechukwu A, Li K, Shen Y. Energy consumption in cloud computing data centers. International Journal of Cloud Computing and Services Science. 2014 Jun;3(3):145-162.
- [34] Quarati A, Clematis A, Agostino D. Delivering cloud services with QoS requirements: Business opportunities, architectural solutions and energy-saving aspects. Future Generation Computer Systems. 2016 Feb;55(C):403-427.
- [35] ARM Cortex-M Series. Cambridge. United Kingdom:1990. Available from: <http://www.arm.com/products/processors/cortex-m/index.php> [accessed 31.12.14]
- [36] Tensilica. Available from: <http://www.tensilica.com/products/xtensa-customizable> [accessed 14.11.14]
- [37] Donofrio D, Olikier L, Shalf J, Wehner F, Rowen C, Krueger J, Mohiyuddin M. Energy-efficient computing for extreme-scale science. Computer. 2009 Nov;42(11):62-71.
- [38] IBM. Aquasar project. Made in IBM labs: IBM Hot Water-Cooled Supercomputer Goes Live at ETH Zurich. Available from: <http://www03.ibm.com/press/us/en/pressrelease/32049.wss> [accessed 16.12.16]
- [39] SuperMUC: supercomputer hosted at Leibniz Supercomputing Centre. Available from: <http://www.lrz.de/services/compute/supermuc/systemdescription> [accessed 13.07.18]
- [40] GreenPeace. How Clean is Your Cloud? Climate Report. Amsterdam (Netherlands): Greenpeace International. Available from: <http://www.greenpeace.org/international/publication/6986/how-clean-is-your-cloud> [accessed 14.11.17]

- [41] Galizia A, Quarati A. Job allocation strategies for energy-aware and efficient Grid infrastructures. *Journal of Systems and Software*. 2012 Jul;85(7):1588-606.
- [42] Zikos S, Karatza D. Performance and energy aware cluster-level scheduling of compute-intensive jobs with unknown service times. *Simulation Modelling Practice and Theory*. 2011 Jan;19(1):239-50.
- [43] Nozaki Y. Development of higher-voltage direct current power feeding system for ICT equipment. *NTT Technical Review*. 2009 Oct;7(10).
- [44] Quarati A, Clematis A, Agostino D. Delivering cloud services with QoS requirements: Business opportunities, architectural solutions and energy-saving aspects. *Future Generation Computer Systems*. 2016 Feb;55:403-427.
- [45] Blazek M, Chong H, Loh W, Koomey G. Data centers revisited: assessment of the energy impact of retrofits and technology trends in a high-density computing facility. *Journal of Infrastructure Systems*. 2004 Sep;10(3):98-104.
- [46] Beloglazov A, Abawajy J, Buyya R. Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. *Future Generation Computer Systems*. 2011 May;28(5):755-768.
- [47] Moore D, Chase S, Ranganathan P, Sharma K. Making scheduling "Cool": Temperature-aware workload placement in data centers. In *USENIX annual technical conference*. 2005 Apr (pp. 61-75).
- [48] Chai L, Gao Q, Panda K. Understanding the impact of multi-core architecture in cluster computing: A case study with intel dual-core system. In *7th IEEE International Symposium on Cluster Computing and the Grid*. 2007 May (pp. 471-478).
- [49] Loper J, Parr S. Energy efficiency in data centers: A new policy frontier. *Environmental Quality Management*. 2007 Jun;16(4):83-97.
- [50] Lee C, Zomaya Y. Energy efficient utilization of resources in cloud computing systems. *The Journal of Supercomputing*. 2012 May;60(2):268-80.
- [51] Scalable, energy-efficient data centres and clouds. Report of the Institute for Energy Efficiency. California. 2012. Available from: <http://iee.ucsb.edu/institute> [accessed 30.09.17].
- [52] Ahmed M, Chowdhury S, Ahme M, Rafee M. An advanced survey on cloud computing and state-of-the-art research issues. *International Journal of Computer Science Issues*. 2012 Jan;9(1):201-207.
- [53] Kim G, Eom H, Yeom Y. Virtual machine scheduling for multicores considering effects of shared on-chip last level cache interference. In *International Green Computing Conference*. 2012 Jun (pp. 1-6).
- [54] Vallee G, Naughton T, Engelmann C, Ong H, Scott L. System-level virtualization for high performance computing. In *16th Euromicro Conference on Parallel, Distributed and Network-Based Processing*. 2008 Feb (pp. 636-643).

- [55] Cao Z, Dong S. An energy-aware heuristic framework for virtual machine consolidation in Cloud computing. *The Journal of Supercomputing*. 2014 Jul;69(1):429-51.
- [56] Horri A, Mozafari MS, Dastghaibiyfard G. Novel resource allocation algorithms to performance and energy efficiency in cloud computing. *The Journal of Supercomputing*. 2014 Sep;69(3):1445-1461.
- [57] Arianyan E, Taheri H, Sharifian S. Novel energy and SLA efficient resource management heuristics for consolidation of virtual machines in cloud data centers. *Computers and Electrical Engineering*. 2015 Oct;47:222-240.
- [58] Beloglazov A, Buyya R. 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*. 2012 Sep;24(13):1397-420.
- [59] Neugebauer R, McAuley D. Energy is just another resource: Energy accounting and energy pricing in the Nemesis OS. In *Proceedings of 8th Workshop on Hot Topics in Operating Systems*. 2001 May (pp. 67-72).
- [60] Zeng H, Ellis S, Lebeck R, Vahdat A. ECOSystem: Managing energy as a first class operating system resource. *ACM SIGOPS Operating Systems Review*. 2002 Dec;36(5):123-32.
- [61] Pinheiro E, Bianchini R, Carrera V, Heath T. Load balancing and unbalancing for power and performance in cluster-based systems. In *Proceedings of the Workshop on Compilers and Operating Systems for Low Power*. 2001 (pp. 182-195).
- [62] Chase S, Anderson C, Thakar N, Vahdat M, Doyle P. Managing energy and server resources in hosting centers. *ACM SIGOPS Operating Systems Review*. 2001 Dec;35(5):103-16.
- [63] Srikantaiah S, Kansal A, Zhao F. Energy-aware consolidation for cloud computing. In *Proceedings of the Conference on Power Aware Computing and Systems*. 2008 Nov (pp. 1-5).
- [64] Zeng G, Yokoyama T, Tomiyama H, Takada H. Practical energy-aware scheduling for real-time multiprocessor systems. In *Proceedings of the IEEE International Conference on Embedded, Real-Time Computing Systems and Applications*. 2008 (pp. 383-392).
- [65] Buyya R, Beloglazov A, Abawajy J. Energy-efficient management of data center resources for cloud computing: a vision, architectural elements, and open challenges. *International Conference on Parallel and Distributed Computing, Applications and Technologies*. 2010 Jun (pp. 1-12).
- [66] Beloglazov A, Buyya R. Energy efficient resource management in virtualized cloud data centers. In *Proceedings of the 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing*. 2010 May (pp. 826-831).

- [67] Youness H, Hassan M, Salem A. A Design Space Exploration methodology for allocating Task Precedence graphs to multi-core system architectures. In IEEE International Conference on Microelectronics 2010 Dec (pp. 260-263).
- [68] Beloglazov A, Buyya R. Adaptive threshold-based approach for energy-efficient consolidation of virtual machines in cloud data centers. In Proceedings of the 8th International Workshop on Middleware for Grids, Clouds and e-Science. 2011 Nov (pp. 4-10).
- [69] Lin C, Liu P, Wu J. Energy-efficient virtual machine provision algorithms for cloud systems. In 4th IEEE International Conference on Utility and Cloud Computing. 2011 Dec (pp. 81-88).
- [70] Lee C, Zomaya Y. Energy conscious scheduling for distributed computing systems under different operating conditions. IEEE Transactions on Parallel and Distributed Systems. 2011 Dec;22(8):1374-1381.
- [71] Feller E, Rilling L, Morin C. Energy-aware ant colony based workload placement in clouds. In Proceedings of the 2011 IEEE/ACM 12th International Conference on Grid Computing. 2011 Sep (pp. 26-33).
- [72] Jeyarani R, Nagaveni N, Ram V. Design and implementation of adaptive power-aware virtual machine provisioner (APA-VMP) using swarm intelligence. Future Generation Computer Systems. 2012 May;28(5):811-821.
- [73] Goiri I, Berral L, Fito O, Julia F, Nou R, Guitart J, Torres J. Energy-efficient and multifaceted resource management for profit-driven virtualized data centers. Future Generation Computer Systems. 2012 May;28(5):718-731.
- [74] Cao Z, Dong S. Energy-aware framework for virtual machine consolidation in cloud computing. In 10th International Conference on High Performance Computing and Communications. 2013 Nov (pp. 1890-1895).
- [75] Beloglazov A, Buyya R. 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. 2012 Aug 15;24(7):1366-79.
- [76] Jeong J, Kim H, Lee J, Seo E. Analysis of virtual machine live-migration as a method for power-capping. The Journal of Supercomputing. 2013 May;66(3):1629-55.
- [77] Farahnakian F, Pahikkala T, Liljeberg P, Plosila J. Energy aware consolidation algorithm based on k-nearest neighbour regression for cloud data centers. In Proceedings of the IEEE/ACM 6th International Conference on Utility and Cloud Computing. 2013 Dec (pp. 256-259).
- [78] Gao Y, Guan H, Qi Z, Hou Y, Liu L. A multi-objective ant colony system algorithm for virtual machine placement in cloud computing. Journal of Computer and System Sciences. 2013 Dec;79(8):1230-1242.
- [79] Wu M, Chang S, Chan Y. A green energy-efficient scheduling algorithm using the DVFS technique for cloud data centers. Future Generation Computer Systems. 2014 Jul;37(7):141-147.

- [80] Niewiadomska-Szynkiewicz E, Sikora A, Arabas P, Kamola M, Mincer M, Kołodziej J. Dynamic power management in energy-aware computer networks and data intensive computing systems. *Future Generation Computer Systems*. 2014 Jul;37:284-296.
- [81] Somasundaram S, Govindarajan K. CLOUDRB: a framework for scheduling and managing high performance computing applications in science cloud. *Future Generation Computer Systems*. 2014 May;34(5):47-65.
- [82] Das A, Kumar A, Veeravalli B. Communication and migration energy aware task mapping for reliable multiprocessor systems. *Future Generation Computer Systems*. 2014 Jan;30(1):216-228.
- [83] Hosseinimotlagh S, Khunjush F, Samadzadeh R. SEATS: smart energy-aware task scheduling in real-time cloud computing. *The Journal of Supercomputing*. 2014 Nov;71(1):45-66.
- [84] Corradi A, Fanelli M, Foschini L. VM consolidation: a real case based on open stack cloud. *Future Generation Computer Systems*. 2014 Mar;32:118-127.
- [85] Sampaio M, Barbosa G. Towards high-available and energy-efficient virtual computing environments in the cloud. *Future Generation Computer Systems*. 2014 Nov;40: 30-43.
- [86] Peng C, Yang Q, Yu Y, Zhou Y, Wang Z, Du S. Host load prediction-based GMDH-EA and MMTP for virtual machines load balancing in cloud environment. In *Proceedings of the International Conference on Grid, Cloud, and Cluster Computing*. 2014 Sep (pp. 9-15).
- [87] Farahnakian F, Pahikkala T, Liljeberg P, Plosila J, Tenhunen H. Utilization prediction aware VM consolidation approach for green cloud computing. In *IEEE 8th International Conference on Cloud Computing*. 2015 Jun (pp. 381-388).
- [88] Fu X, Zhou C. Virtual machine selection and placement for dynamic consolidation in cloud computing environment. *Frontiers of Computer Science*. 2015 Apr;9(2):322-30.
- [89] Mann A. Rigorous results on the effectiveness of some heuristics for the consolidation of virtual machines in a cloud data center. *Future Generation Computer Systems*. 2015 Oct;51:1-6.
- [90] Viswanathan H, Lee K, Rodero I, Pompili D. Uncertainty-aware autonomic resource provisioning for mobile cloud computing. *IEEE Transactions on Parallel and Distributed Systems*. 2015 Jul;26(8):2363-72.
- [91] Lai Z, Lam T, Wang L, Su J. Latency-aware DVFS for efficient power state transitions on many-core architectures. *The Journal of Supercomputing*. 2015 Apr;71(7):2720-2747
- [92] Dauwe D, Jonardi E, Friese D, Pasricha S, Maciejewski A, Bader A, Siegel J. HPC node performance and energy modeling with the co-location of applications. *The Journal of Supercomputing*. 2016 Dec;72(12):4771-809.

- [93] Zhang W, Wen Y, Wong W, Toh C, Chen H. Towards joint optimization over ICT and cooling systems in data centre: A survey. *IEEE Communications Surveys and Tutorials*. 2016 Mar;18(3):1596-1616.
- [94] Li H, Zhu G, Cui C, Tang H, Dou Y, He C. Energy-efficient migration and consolidation algorithm of virtual machines in data centers for cloud computing. *Computing*. 2016 Mar;98(3):303-17.
- [95] Dashti E, Rahmani M. Dynamic VMs placement for energy efficiency by PSO in cloud computing. *Journal of Experimental and Theoretical Artificial Intelligence*. 2016 Sep;28(1):97-112.
- [96] Vasudevan M, Tian C, Tang M, Kozan E, Zhang W. Profile-based dynamic application assignment with a repairing genetic algorithm for greener data centers. *The Journal of Supercomputing*. 2017 Sep;73(9):3977–98.
- [97] Gabaldon E, Lerida L, Guirado F, Planes J. Blacklist multi-objective genetic algorithm for energy saving in heterogeneous environments. *The Journal of Supercomputing*. 2017 Sep;73(1):354-69.
- [98] Duan, H, Chen C, Min G, Wu Y. Energy-aware scheduling of virtual machines in heterogeneous cloud computing systems. *Future Generation Computer Systems*. 2017 Sep;74(9):142-50.
- [99] Zhou Z, Abawajy J, Chowdhury M, Hu Z, Li K, Cheng H, Alelaiwi A, Li F. Minimizing SLA violation and power consumption in Cloud data centers using adaptive energy-aware algorithms. *Future Generation Computer Systems*. 2017 Sep;86:836-50.
- [100] Xu X, Zhang X, Khan M, Dou W, Xue S, Yu S. A balanced virtual machine scheduling method for energy-performance trade-offs in cyber-physical cloud systems. *Future Generation Computer Systems*. 2017 Sep [*In press*].
- [101] Malekloo H, Kara N, Barachi M. An energy efficient and SLA compliant approach for resource allocation and consolidation in cloud computing environments. *Sustainable Computing: Informatics and Systems*. 2018 Mar;17:9-24.
- [102] Chaabouni T, Khemakhem M. Energy management strategy in cloud computing: a perspective study. *The Journal of Supercomputing*. 2018 Dec;74(12):6569-97.
- [103] Wang H, Tianfield H. Energy-aware dynamic virtual machine consolidation for cloud datacenters. *IEEE Access*. 2018 Mar;6:15259-73.
- [104] Mishra K, Puthal D, Sahoo B, Jayaraman P, Jun, S, Zomaya, Y, Ranjan R. Energy-efficient VM-placement in cloud data center. *Sustainable Computing: Informatics and Systems*. 2018 Feb;20:48-55.
- [105] Ashraf A, Porres I. Multi-objective dynamic virtual machine consolidation in the cloud using ant colony system. *International Journal of Parallel, Emergent and Distributed Systems*. 2018 Jan;33(1):103-20.

- [106] Ghobaei-Arani M, Rahmanian A, Shamsi M, Rasouli-Kenari A. A learning-based approach for virtual machine placement in cloud data centers. *International Journal of Communication Systems*. 2018 May;31(8):3537.
- [107] Goudarzi H, Pedram M. Energy-efficient virtual machine replication and placement in a cloud computing system. In 5th IEEE International Conference on Cloud Computing. 2012 Jun (pp. 750–757).
- [108] Cloud to Cut Energy Consumption 31% by 2020. 2016. Available from: <https://www.itracs.com/cloud-to-cut-energy-consumption-31-by-2020> [accessed 04.09.17].
- [109] How Clean Is the Energy Used by Tech Companies for Cloud Computing. 2011. Available from: <https://www.scientificamerican.com> [accessed 04.09.17].
- [110] Teng F, Yu L, Li T, Deng D, Magoules F. Energy efficiency of VM consolidation in IaaS clouds. *The Journal of Supercomputing*. 2017 Feb;73(2):782-809.
- [111] Moreno S, Yang R, Xu J, Wo T. Improved energy-efficiency in cloud data-centers with interference-aware virtual machine placement. In 11th IEEE International Symposium on Autonomous Decentralized Systems. 2013 Mar (pp. 1-8).
- [112] Vilaplana J, Mateo J, Teixido I, Solsona F, Giné F, Roig C. An SLA and power-saving scheduling consolidation strategy for shared and heterogeneous clouds. *The Journal of Supercomputing*. 2015 May;71(5):1817-32.
- [113] Alahmadi A, Che D, Khaleel M, Zhu M, Ghodous P. An innovative energy-aware cloud task scheduling framework. In 8th IEEE International Conference on Cloud Computing. 2015 Jun (pp. 493-500).
- [114] Babukarthik G, Raju R, Dhavachelvan P. Energy-aware scheduling using hybrid algorithm for cloud computing. In 3rd International Conference on Computing, Communication and Networking Technologies. 2012 Jul (pp. 1-6).
- [115] Pietri I, Sakellariou R. Energy-aware workflow scheduling using frequency scaling. In 43rd International Conference on Parallel Processing Workshops. 2014 Sep (pp. 104-113).
- [116] Buyya R, Ranjan R, Calheiros N. Modeling and simulation of scalable cloud computing environments and the CloudSim toolkit: Challenges and opportunities. In IEEE International Conference on High Performance Computing and Simulation. 2009 Jun (pp. 1-11).
- [117] Standard Performance Evaluation Corporation. Available from: https://www.spec.org/power_ssj2008 [accessed 12.11.17]
- [118] Park K, Pai S. CoMon: a mostly-scalable monitoring system for PlanetLab. *ACM SIGOPS Operating Systems Review*. 2006 Jan;40(1):65-74.
- [119] Mishra M, Sahoo A. On theory of VM placement: anomalies in existing methodologies and their mitigation using a novel vector based approach. In 4th IEEE International Conference on Cloud Computing. 2011 Jul (pp. 275-282).

- [120] Yoon K, Hwang C. Multiple attribute decision making. Thousand Oaks, California.: Sage; 1995.
- [121] Rao R. Decision making in manufacturing environment using graph theory and fuzzy multiple attribute decision making methods. London: Springer; 2013.
- [122] Ramanathan R. A note on the use of the analytic hierarchy process for environmental impact assessment. *Journal of Environmental Management*. 2001 Sep;63(1):27-35.
- [123] Saaty T, Vargas L. Decision making with the analytic network process. Pittsburgh PA: Springer; 2010.
- [124] Hanson H, Keckler W, Ghiasi S, Rajamani K, Rawson F, Rubio J. Thermal response to DVFS: Analysis with an Intel Pentium M. In *Proceedings of the IEEE International Symposium on Low Power Electronics and Design*. 2007 Aug (pp. 219-224).
- [125] Zhou Z, Hu Z, Li K. Virtual machine placement algorithm for both energy-awareness and SLA violation reduction in cloud data centers. *Scientific Programming*. 2016 Mar;2016(1):1-11.