Power-Aware Virtual Machine Scheduling Technique for Cloud

Thesis submitted in partial fulfillment of the requirements for the award of degree of

Master of Engineering
in
Software Engineering

Submitted By
Taranpreet kaur
(Roll No. 801131027)

Under the supervision of:
Dr. Inderveer Chana
Associate Professor

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT
THAPAR UNIVERSITY
PATIALA – 147004

July 2013

CERTIFICATE

I hereby certify that the work which is being presented in the thesis entitled, "*Power-Aware Virtual Machine Scheduling Technique for Cloud*", in partial fulfillment of the requirements for the award of degree of Master of Engineering in *Software Engineering* submitted in Computer Science and Engineering Department of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of *Dr. Inderveer Chana* and refers other researcher's work which are duly listed in the reference section.

The matter presented in the thesis has not been submitted for award of any other degree of this or any other University.

(Taranpreet Kaur)

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

15/7/13

(Dr. Inderveer Chana)
Assistant Professor,
CSED,
Thapar University,
Patiala

Countersigned by

(Dr. Maninder Singh)
Head
Computer Science and Engineering Department
Thapar University
Patiala

(Dr. S. K. Mohapatra)
Dean (Academic Affairs)
Thapar University
Patiala

Acknowledgement

First of all, I am thankful to God for his blessings and showing me the right direction. With His mercy, it has been made possible for me to reach so far.

It gives me great pleasure to express my gratitude towards the guidance and help I have received from Dr. Inderveer Chana. I am thankful for her continual support, encouragement, and invaluable suggestion. She not only provided me help whenever needed, but also the resources required to complete this thesis report on time.

I am also thankful to Dr. Maninder Singh, Head, Computer Science and Engineering Department for his kind help and cooperation. I express my gratitude to all the staff members of Computer Science and Engineering Department for providing me all the facilities required for the completion of my thesis work.

I extend my thanks to Ms. Nidhi, Ms. Rajni, Ms. Tarunpreet kaur and Ms. Pankajdeep Kaur for sharing their expertise and time to help me accomplish this work.

I would like to say thanks to all my friends especially Rajinder Sandhu, Tejvir Kaur and Sukhpal Singh for their support. I want to express my appreciation to every person who contributed with either inspirational or actual work to this thesis.

Last but not the least I am highly grateful to all my family members for their inspiration and ever encouraging moral support, which enables me to purse my studies.

Taranpreet Kaur

(801131027)

Abstract

Cloud Computing offers services to end-users rather than a product, by sharing resources, software and other information under a usage based payment model. It enables hosting of various kinds of applications such as business, scientific, social network, etc. as it has key characteristics like multi-tenancy, scalability, performance, security, etc. Economic benefits are the main driver for the Cloud, since it promises the reduction of Capital Expenditure (CapEx) and Operational Expenditure (OpEx).

While Cloud Computing provides many benefits, it still has some issues as well. In recent years, Cloud Computing is facing many challenges like Data Security, Energy Consumption, Server Consolidation, Virtual Machine Migration, etc. This research work focuses on the study of energy-aware management of VMs in heterogeneous cloud environment. Energy-aware management of VMs in Cloud data centers point towards systematic and correct scheduling of server resources such that energy consumption and SLA violations are minimum, resulting in reduced operational costs which benefits end-users with decreased prices for resource usage.

In this thesis, existing virtual machine management approaches have been surveyed and a power-aware virtual machine scheduling policy is designed for Cloud systems. In particular, an optimized technique for efficient server allocation is discussed. Objective of this technique is to place VMs on host while keeping total utilization of CPU below defined threshold and then optimizing the VM allocation by constrained consolidation of VMs and switching idle nodes to sleep mode to minimize power consumption. As excessive consolidation of VMs can lead to significant SLA violation, in proposed approach not only overall number of used servers are minimized, but also number of migrations are minimized, resulting in minimum energy consumption and least SLA violations. A simulated environment provided by CloudSim Toolkit has been used to validate experimental results, which demonstrates that the proposed approach can provide substantial energy savings, reduced SLA violations and fewer VM migrations as compared to existing techniques, while preventing frequent power cycling of servers, thus validating the proposed policy.

Table of Content

List of Figures and Tables

Figures

Tables

Chapter 1

Introduction

This chapter introduces Cloud computing, its evolution and various related technologies like distributed computing, grid computing, Cloud characteristics and the services offered by Cloud along with motivation of research and organisation of thesis.

1.1 Cloud Computing Evolution

The term Cloud has been used historically as a metaphor for the Internet. Cloud computing has been evolved through number of phases which include Grid and Utility computing, Application Service Provision (ASP), and Software as a Service (SaaS) as shown in Figure1.1 [1]. This concept of Cloud computing dates back to the 1960s, when John McCarthy opined that computation may someday be organized as a public utility [1].

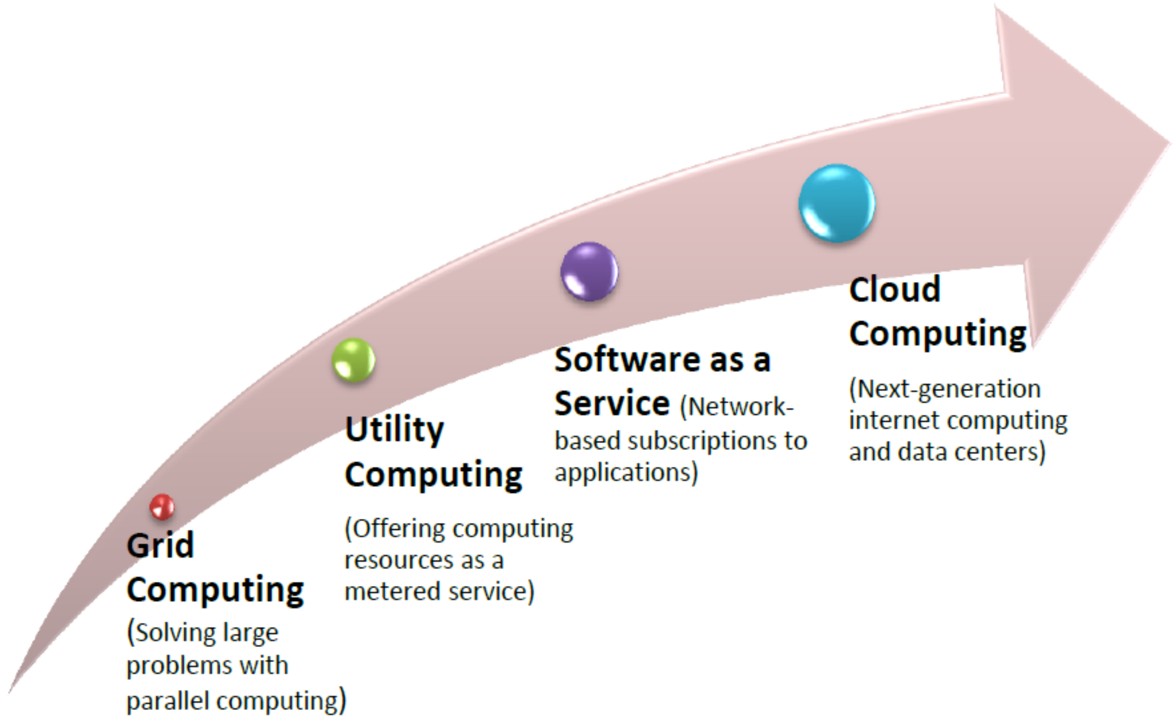


Figure 1.1: Evolution of Cloud Computing [1]

The concepts inspired by the notion of utility computing have recently combined with the requirements and standards of Web 2.0 [2] to create Cloud computing [3]. Cloud computing is defined as, "A large-scale distributed computing paradigm that is driven by economies of scale, in which a pool of abstracted, virtualized, dynamically-scalable, managed computing power, storage, platforms and services are delivered on demand to external customers over the Internet" [3].

The data in a Cloud is stored on to centralized location called data centers having a large size of data storage. The data as well as processing is somewhere on servers. So, the clients have to trust the provider on the availability as well as data security. The SLA is the only legal agreement between the service provider and client. "The basic principle of Cloud computing is to distribute the computing tasks to many distributed computers, not local computer or remote servers" [4].

In 1999 salesforce.com put the idea of Cloud computing in an application. They used a simple website to deliver its enterprise applications to users. Then in 2002 Amazon launched their Cloud based web services. Cloud computing does not stop here; it was then used by Microsoft to promote the Cloud computing services by Azure Services platform in October 27, 2008. Google in 2009 came into market via using Cloud computing concept by delivering Google Apps [6].

Cloud computing offers service to end-users rather than a product, by sharing resources, software and other information under a usage based payment model. Cloud computing refers to both the applications delivered as services over the Internet and the hardware and systems software in the data centers that provide those services [7]. Figure 1.2 shows the main aspects of Cloud computing. There are many proposed definitions of the Cloud computing due to its growing popularity defining its characteristics. Some of the definitions given by many well-known scientists and organizations are:

- Rajkumar Buyya defines the Cloud computing in terms of its utility to end user as "A Cloud is a type of parallel and distributed system consisting of a collection of interconnected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resources based on service-level agreements established through negotiation between the service provider and consumers" [3].
- National Institute of Standards and Technology (NIST) defines Cloud computing as follows: "Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This Cloud model

promotes availability and is composed of five essential characteristics, three service models, and four deployment models" [5].

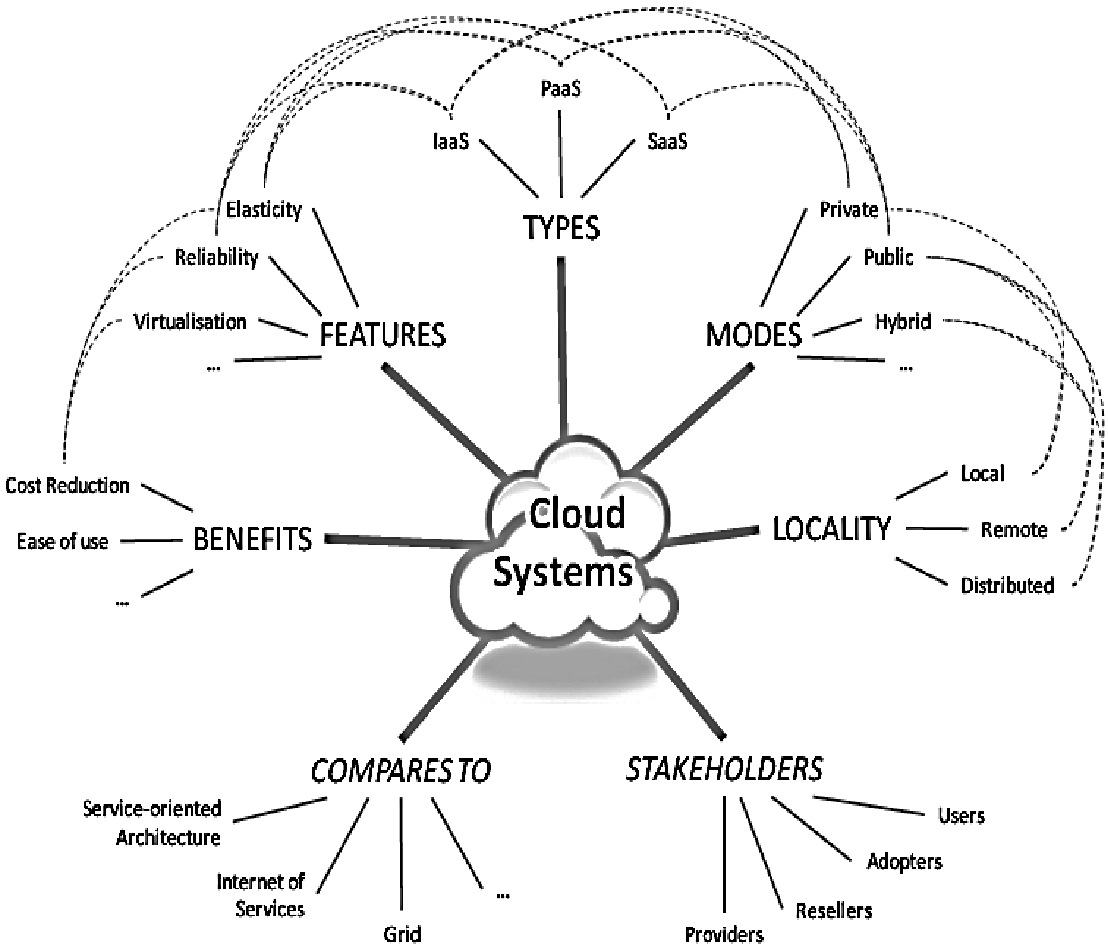


Figure 1.2: Non-Exhaustive view on the main aspects forming a cloud system [6]

1.1.1 Characteristics of Cloud Computing

The different characteristics of Cloud computing [7] are described below:

- Reduced cost: There are a number of reasons to attribute Cloud technology with lower costs. The billing model is pay as per usage; the infrastructure is not purchased thus lowering maintenance. Initial expense and recurring expenses are much lower than traditional computing.
- Increased Storage: With the massive Infrastructure that is offered by Cloud providers today, storage & maintenance of large volumes of data is a reality. Sudden workload

spikes are also managed effectively & efficiently, since the Cloud can scale dynamically.

- Flexibility: Cloud computing stresses on getting applications to market very quickly, by using the most appropriate building blocks necessary for deployment.
- Reliability: It is the capability to ensure constant operation of the system without disruption, i.e. no loss of data, no code reset during execution etc. Reliability is typically achieved through redundant resource utilization.
- Location independence: It enables users to access systems using a web browser regardless of their location or what device they are using.

1.1.2 Cloud Computing Services

Cloud computing is typically divided into three levels of service offerings [7] as shown in Figure 1.3.

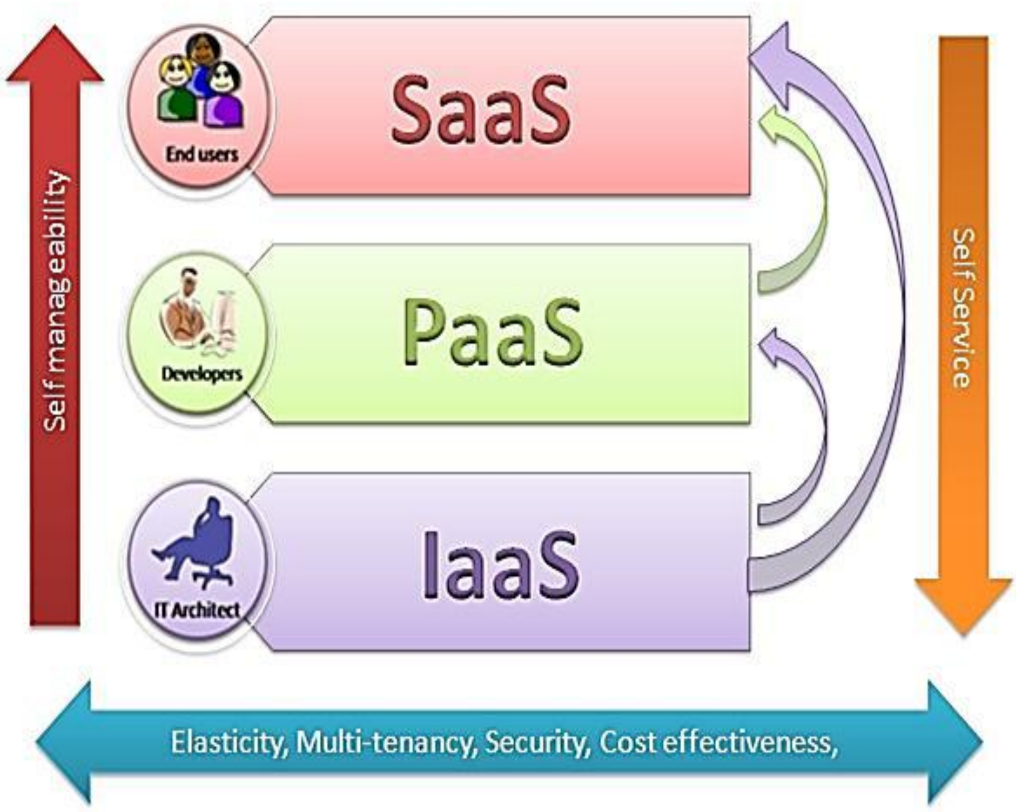


Figure 1.3: Services of cloud computing [7]

- Software as a Service (SaaS)

A complete application is offered to the customer, as a service on demand. A single instance of the service runs on the Cloud & multiple end users are serviced. On the customer side, there is no need for upfront investment in servers or software licenses, while for the provider, the costs are lowered, since only a single application needs to be hosted & maintained. Today SaaS is offered by companies such as Google, Salesforce, Microsoft, Zoho, etc.

- Platform as a Service (PaaS)

PaaS is an application development and deployment platform delivered as a service to developers over the Web. It facilitates development and deployment of applications without the cost and complexity of buying and managing the underlying infrastructure, providing all of the facilities required to support the complete life cycle of building and delivering web applications and services entirely available from the Internet. This platform consists of infrastructure software, and typically includes a database, middleware and development tools. PaaS providers offer a predefined combination of OS and application servers, such as LAMP platform (Linux, Apache, MySql and PHP), Google Apps Engine is a PaaS offering where developers write in Python or Java.

- Infrastructure as a service (IaaS)

Infrastructure as a Service is the delivery of hardware (server, storage and network), and associated software (operating systems virtualization technology, file system), as a service. It is an evolution of traditional hosting that does not require any long term commitment and allows users to provision resources on demand. Unlike PaaS services, the IaaS provider does very little management other than keep the data center operational and users must deploy and manage the software services themselves – just the way they would in their own data center. Amazon Web Services Elastic Compute Cloud (EC2) and Secure Storage Service (S3), GoGrid, 3 Tera are the examples of IaaS offerings.

1.1.3 Deployment Models of Cloud Computing

The different deployment models of Cloud Computing [7] are shown in Figure 1.4.

- Public Cloud

Public Cloud is the most common deployment model where services are available to anyone on Internet in a pay-as-you-go manner. To support thousands of public domain users, datacenters built by public Cloud providers are quite large comprising of thousands of servers with high speed network. Some of the famous public Clouds are Amazon Web Services (AWS), Google App Engine, and Microsoft Azure. Amazon EC2 provides infrastructure as a service, Google AppEngine provides platform as a service, and Salesforce.com provides software as a service. The characteristics of Public Clouds are multi-tenancy and scalability.

- Private Cloud

Private Clouds are deployed within the premise of an organization to provide IT services to its internal users. The private Cloud services offer greater control over the infrastructure, improving security and service resilience because its access is restricted to one or few organizations. Such private deployment poses an inherent limitation to end user applications i.e. inability to scale elastically on demand as can be done using pubic Cloud services. An organization can buy more machines according to expanding needs of its users, but this cannot be done as fast and seamlessly as with public Clouds.

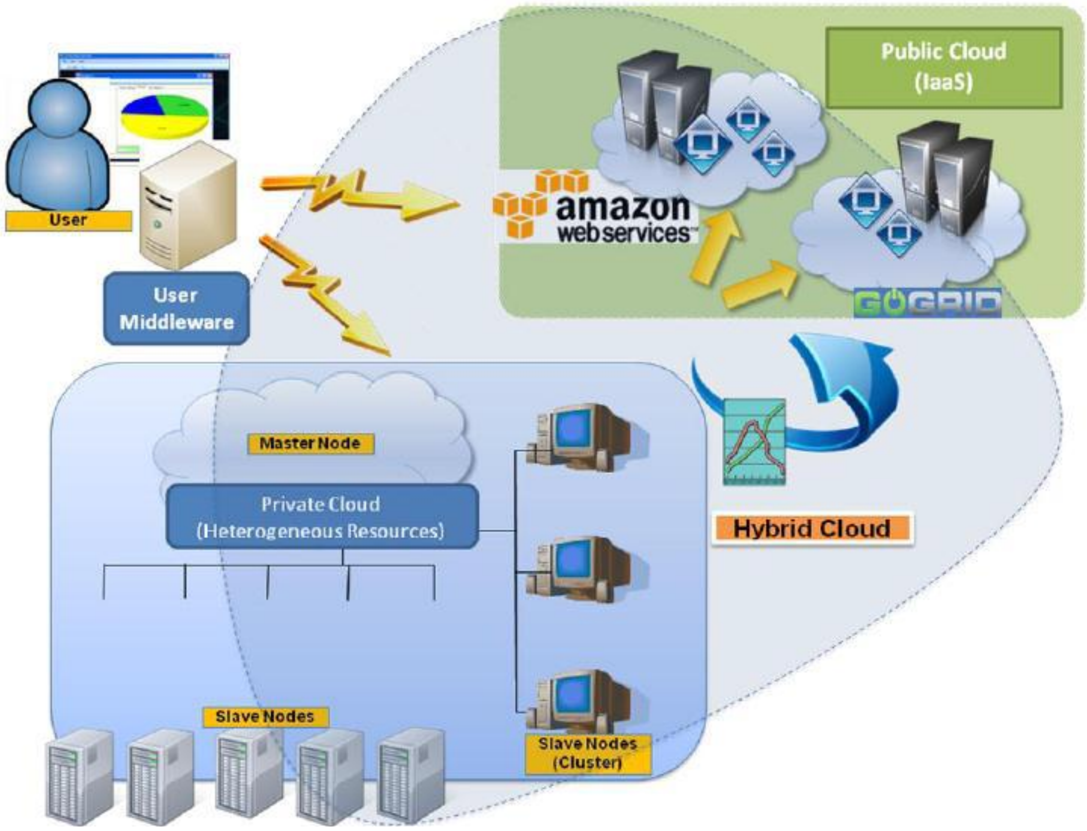


Figure 1.4: Cloud Computing Models [8]

- Hybrid Cloud

Hybrid Clouds is the deployment which emerged due to diffusion of both public and private Clouds advantages. In this model, organizations outsource non-critical information and processing to the public Cloud, while keeping critical services and data in their control. Therefore, organizations can utilize their existing IT infrastructure for maintaining sensitive information within the premises. With a Hybrid Cloud, service providers can utilize 3rd party Cloud Providers in a full or partial manner thus increasing the flexibility of computing.

1.2 Research Issues in Cloud Computing

The emergence of Cloud computing has made a tremendous impact on the Information Technology (IT) industry over the past few years. Currently IT industry needs Cloud computing services to provide best opportunities to real world. Cloud computing is in initial stages, with many issues [9], [10] still to be addressed.

- Automated Service Provisioning

The objective of a service provider in this case is to allocate and de-allocate resources from the Cloud to satisfy its service level objectives (SLOs), while minimizing its operational cost. These approaches typically involve: (i) Constructing an application performance model that predicts the number of application instances required to handle demand at each particular level, in order to satisfy QoS requirements; (ii) Periodically predicting future demand and determining resource requirements using the performance model; and (iii) Automatically allocating resources using the predicted resource requirements. Application performance model can be constructed using various techniques, including Queuing theory, Control theory and Statistical Machine Learning. The proactive approach uses predicted demand to periodically allocate resources before they are needed. The reactive approach reacts to immediate demand fluctuations before periodic demand prediction is available [9].

- Virtual Machine Migration

Virtualization can provide significant benefits in Cloud computing by enabling virtual machine migration to balance load across the data center. In addition, virtual machine migration enables robust and highly responsive provisioning in data centers. Authors have pointed out that migrating an entire OS and all of its applications as one unit allows to avoid many of the difficulties faced by process-level migration approaches, and analyzed the benefits of live migration of VMs. The major benefit of VM migration is to avoid hotspots. Detecting workload hotspots and initiating a migration lacks agility to respond to sudden workload changes [10].

- Server Consolidation

Server consolidation is an effective approach to maximize resource utilization while minimizing energy consumption in a Cloud computing environment. Live VM migration technology is often used to consolidate VMs residing on multiple under-utilized servers onto a single server, so that the remaining servers can be set to an energy-saving state. The problem of optimally consolidating servers in a data center is often formulated as a variant of the vector bin-packing problem, which is an NP-hard optimization problem. Various heuristics have been proposed for this problem. Server consolidation activities should not hurt application performance. For server resources that are shared among VMs, such as bandwidth, memory cache and disk I/O, maximally consolidating a server may result in resource congestion when a VM changes its footprint on the server. Hence, it is sometimes important to observe the fluctuations of VM footprints and use this information for effective server consolidation. Finally, the system must quickly react to resource congestions as and when they occur [10].

- Energy Management

Improving energy efficiency is another major issue in Cloud computing. It has been estimated that the cost of powering and cooling accounts for 53% of the total operational expenditure of data centers [11]. In 2006, data centers in the US consumed more than 1.5% of the total energy generated in that year, and the percentage is projected to grow 18% annually [11]. Hence infrastructure providers are under enormous pressure to reduce energy consumption. The goal is not only to cut down energy cost in data centers, but also to meet government regulations and environmental standards. Designing energy efficient data centers has recently received considerable attention. This problem can be approached from several directions. For example, energy-efficient hardware architecture that enables slowing down CPU speeds and turning off partial hardware components has become commonplace. Energy-aware job scheduling and server consolidation are two other ways to reduce power consumption by turning off unused machines. A key challenge in all the above methods is to achieve a good trade-off between energy savings and application performance.

- Data Security

Data security is another important research topic in Cloud computing. Since service providers typically do not have access to the physical security system of data centers, they must rely on the infrastructure provider to achieve full data security. Even for a virtual private Cloud, the service provider can only specify the security setting remotely, with-out knowing whether it is fully implemented. It is critical to build trust mechanisms at every architectural layer of the Cloud. Firstly, the hardware layer must be trusted using hardware trusted platform module. Secondly, the virtualization platform must be trusted using secure virtual machine monitors. VM migration should only be allowed if both source and destination servers are trusted [9].

1.3 Research Motivation

As new distributed computing technologies like Cloud has become increasingly popular, the dependence on power also increases. Currently it is estimated that data centers consume 0.5 percent of the world's total electricity usage [12] and if demand continues, is projected to quadruple by 2020. Therefore, it is imperative to enhance the efficiency and potential sustainability of large data centers.

One of the ways to reduce power consumption by a data center is to apply virtualization technology, leveraging Cloud computing paradigm. This technology allows consolidation of several servers to one physical node as virtual machines reducing the amount of the hardware in use. But to ensure efficient resource management and provide higher utilization of resources, Cloud providers have to deal with power-performance trade-off, as aggressive consolidation of VMs can lead to performance loss.

Aim of the thesis is to reduce power consumption by dynamic reallocation of VMs according to current resource requirements, while ensuring reliable Quality of Service (QoS), maximum resource utilization and minimum possible energy consumption. It can further result in reducing carbon footprints and hence assist in achieving Green computing.

1.4 Organization of Thesis

The rest of the thesis is organized as follows:

Chapter 2 – This chapter describes in detail literature survey carried out to study the concept of virtualization and its role in reducing power consumption. Also, various existing techniques for virtual machine management and the Green Cloud architecture have been elaborated in an organized manner.

Chapter 3 – The problem analyzed in thesis work is depicted in this chapter. It also gives the gap analysis and the problem statement.

Chapter 4 – Solution for the problem analyzed is explained in this chapter with the help of design and flowchart of the proposed technique.

Chapter 5 – This section focuses on implementation details and experimental results – description of CloudSim, Eclipse and snapshots of experimental results of the proposed technique.

Chapter 6 – This chapter concludes the research work and explains the contribution to the work done. It also suggests future research directions.

Chapter 2

Literature Review

This chapter aims at summarizing the current state of art in energy efficient virtual machine management in cloud computing system which is researched by proposing answers to the following questions:

(i) Does virtualization help in reduction of power consumption by data centers?

(ii) What are the different approaches for Energy-efficient Virtual machine management in the cloud?

(iii)What are the challenges and benefits of virtual machine migration in the cloud?

2.1 Virtualization

Virtualization is an another term for abstraction. It is a technology that allows running two or more operating systems side-by-side on just one PC or embedded controller [13]. It is rapidly adopted in the engineering world. It helps them to better utilize their available processing hardware to build more efficient systems. As multi-core processors with 4, 8, and 16 cores on a chip become a common place, many processor cores are likely to be underutilized in a typical system. Most applications will have only a finite amount of parallel tasks that can be executed at a given time, leaving many processor cores idle. Virtualization can solve this challenge by allocating group of processor cores to individual operating systems running in parallel as shown in Figure 2.1. Thus benefits of virtualization can be enumerated as [13]:

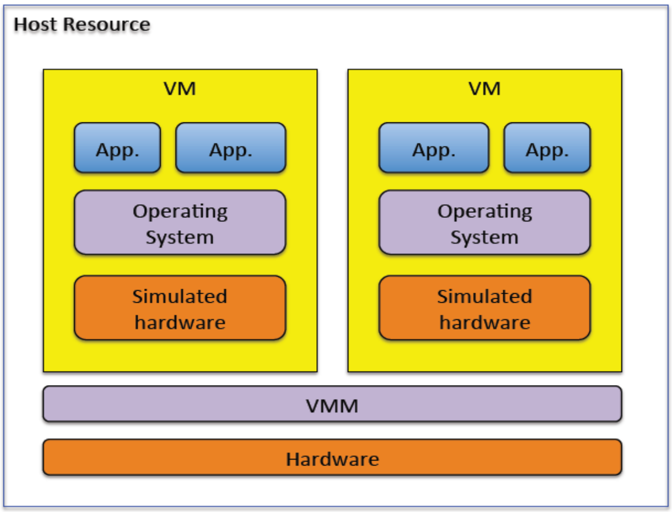


Figure 2.1: Virtual Machine Abstraction [13]

(i) Save hardware cost and footprint.

(ii)Take advantage of Operating System services.

(iii)Make use of Multicore Processors.

Virtualisation technology allows datacenters to lease virtual machines to end users, rather than traditional dedicated machines. This system is the basis of Cloud computing on demand, utility model. Virtualisation can also increase the efficiency of existing machine rooms, reducing the number of physical servers required through consolidation of existing applications by introducing multiple virtual machines per server and thereby increasing resource utilisation. It is assumed that this increases efficiency which further allows doing more work with less IT equipment. Increase in utilisation of a machine can also increase its energy efficiency, as a typical blade server consumes 50% of its peak power drawn when idling. Well utilising that machine will allow the idle cost to be amortized [13].

Energy consumption by a virtual machine [14] is shown in the Figure 2.2. The Figure shows boot up (t=10 to t=30), CPU burn(t=40 to t=100), shut down(t=110 to t=122) of a virtual machine. It is also shown that boot up and shutdown consume really less energy than CPU burn. So we should put virtual machine to sleep mode to save the power consumed when it does nothing [14].

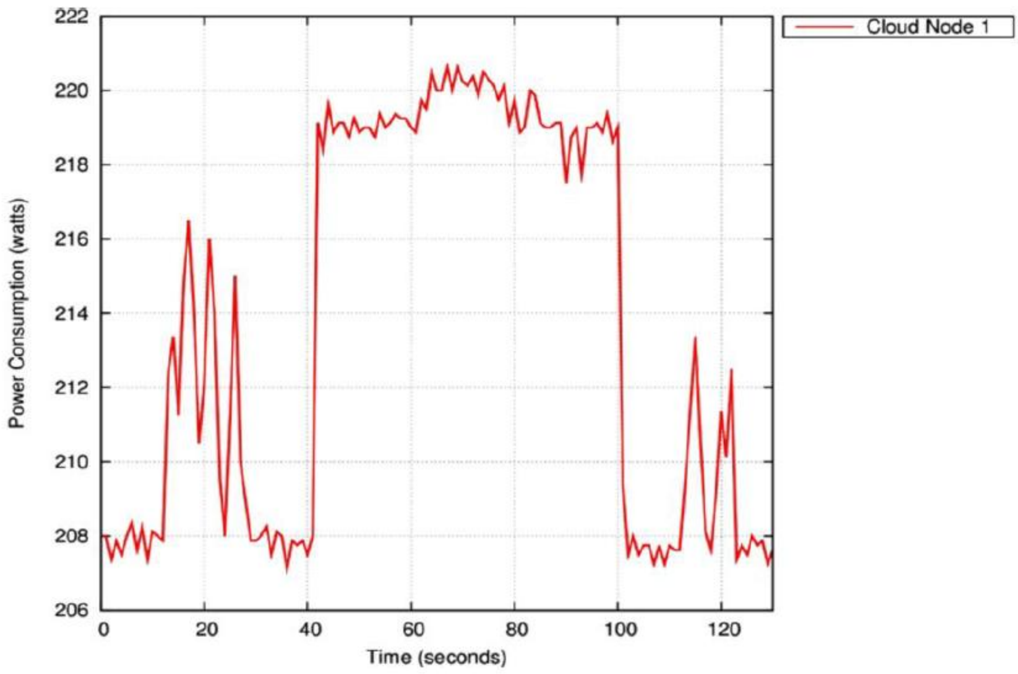


Figure 2.2: Life of a virtual machine: boot, run, and halt [14]

Virtual machine migration, which is used to transfer a VM across physical computers, has served as a main approach to achieve better energy efficiency of Internet Data Centers (IDCs). This is because in doing so, server consolidation via VM migration allows more computers to be turned off. Generally, there are two types of migrations [15]: regular migration and live migration. The former moves a VM from one host to another by pausing the originally used server, copying its memory contents, and then resuming it on the destination. The latter performs the same logical functionality but without the need to pause the server domain for the transition. In general when performing live migrations the domain continues its usual activities and from the user's perspective—the migration should be imperceptible. It shows great potential of using VM and VM migration technology to efficiently manage workload consolidation, and therefore improve the total IDC power efficiency [12]. To deal with power-performance trade-off, it is necessary to understand Green Cloud infrastructure for Cloud computing.

2.2 Green Cloud Framework

With the gained popularity of Cloud computing, power dependency has also increased. It has been estimated that the data centers consume 0.5 percent of the world's total electricity usage [20]. If current demand continues, it is projected to be augmented by 2020. In 2005, the total energy consumption (servers, cooling units) was projected at 1.2% of the total U.S. energy consumption, amplifying every 5 years [21]. Amazon‘s estimates [22] about its data center are shown in Figure 2.3. Major part of the energy used in society nowadays, is generated from fossil fuels which results in harmful CO2 emissions. To manage energy consumption of the Cloud while providing reliable Quality of Service (QoS), Green Cloud architecture should be implemented, which is discussed as below.

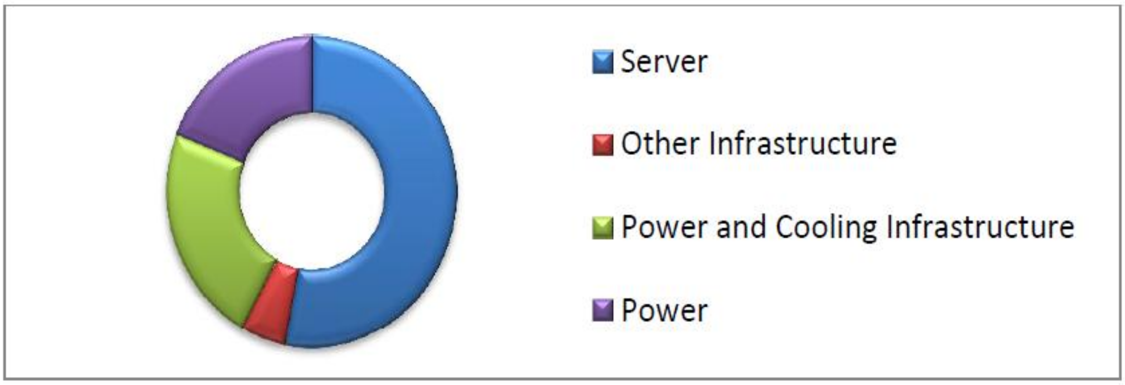


Figure 2.3: Distribution of energy in data center [22]

Green Cloud is an Internet Data Center architecture which aims to reduce data center power consumption, while at the same time guarantee the performance from users‘perspective, leveraging live virtual machine migration technology. A big challenge for Green Cloud is to automatically make the scheduling decision on dynamically migrating/consolidating VMs among physical servers to meet the workload requirements meanwhile saving energy, especially for performance-sensitive applications. Hence, a real-time VM consolidation is needed [16].

Figure 2.4 shows the high-level architecture for supporting energy-efficient service allocation in Green Cloud computing infrastructure [16]. There are basically four main entities involved:

• Consumers/Brokers:

Cloud consumers or their brokers submit service requests from anywhere in the world to the Cloud. It is important to notice that there can be a difference between Cloud consumers and users of deployed services. For instance, a consumer can be a company deploying a Web application, which presents varying workload according to the number of "users" accessing it.

• Green Resource Allocator:

It acts as the interface between the Cloud infrastructure and consumers. It requires the interaction of the following components to support energy-efficient resource management:

- Green Negotiator: Negotiates with the consumers/brokers to finalize the SLA with specified prices and penalties (for violations of SLA) between the Cloud provider and consumer depending on the consumer‘s QoS requirements and energy saving schemes. In case of Web applications, for instance, QoS metric can be 95% of requests being served in less than 3 seconds.
- Service Analyzer: Interprets and analyses the service requirements of a submitted request before deciding whether to accept or reject it. Hence, it needs the latest load and energy information from VM Manager and Energy Monitor respectively.
- Consumer Profiler: Gathers specific characteristics of consumers so that important consumers can be granted special privileges and prioritized over other consumers.
- Pricing: Decides how service requests are charged to manage the supply and demand of computing resources and facilitate in prioritizing service allocations effectively.
- Energy Monitor: Observes and determines which physical machines to power on/off.

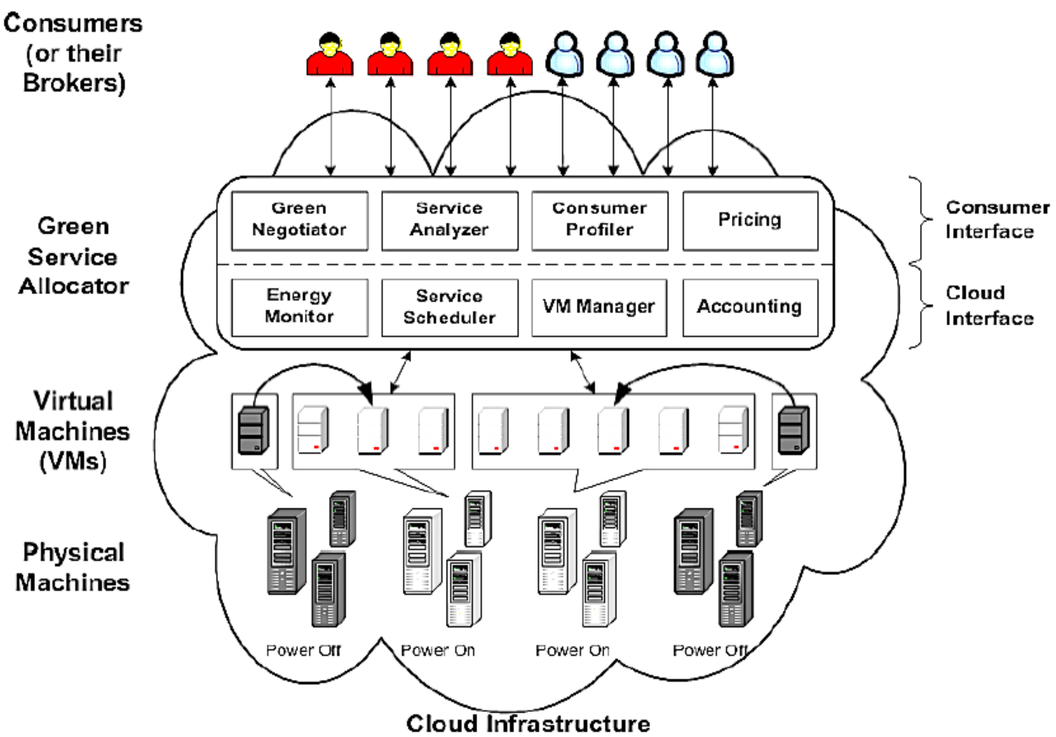


Figure 2.4: High Level Green Cloud Architecture [16]

- Service Scheduler: Assigns requests to VMs and determines resource entitlements for allocated VMs. It also decides when VMs are to be added or removed to meet demand.
- VM Manager: Keeps track of the availability of VMs and their resource entitlements. It is also in charge of migrating VMs across physical machines.
- Accounting: Maintains the actual usage of resources by requests to compute usage costs. Historical usage information can also be used to improve service allocation decisions.
- VMs: Multiple VMs can be dynamically started and stopped on a single physical machine to meet accepted requests, hence providing maximum flexibility to configure various partitions of resources on the same physical machine to different specific requirements of service requests. Multiple VMs can also concurrently run applications based on different operating system environments on a single physical machine. In addition, by dynamically migrating VMs across physical machines, workloads can be consolidated and unused resources can be put on a low-power state, turned off or configured to operate at low-performance levels (e.g., using DVFS) in order to save energy.

- Physical Machines: The underlying physical computing servers provide hardware infrastructure for creating virtualized resources to meet service demands.

Within this framework, there is a major area which can lead to substantial improvement. That is, expansion upon the baseline functioning of virtual machines in a cloud environment. This is achieved by deriving a more efficient scheduling system for VMs. The scheduling system addresses the placement of VMs within the Cloud infrastructure while minimizing the operating costs of the Cloud itself. Due to inherent disposability and mobility of VMs within a semi-homogeneous data center, we can leverage the ability to move and manage VMs resulting in further improvement in efficiency [16]. This concept is elaborated in next section.

2.3 Energy-efficient Virtual Machine Scheduling & Management in Cloud

Various research articles related to energy efficient virtual machine management were reviewed during the study process and then categorized into 3 groups which are:

(i) Cloud based Energy Conservation Models,
(ii) Cloud based Energy Efficiency Analysis,
(iii)Virtual Machine Allocation Techniques.

Breakdown of research articles according to these three categories is shown in Figure 2.5. Based on Figure 2.5, majority of research articles are focused on Virtual Machine Allocation techniques followed by Cloud based Models and Cloud based Energy Efficiency Analysis.

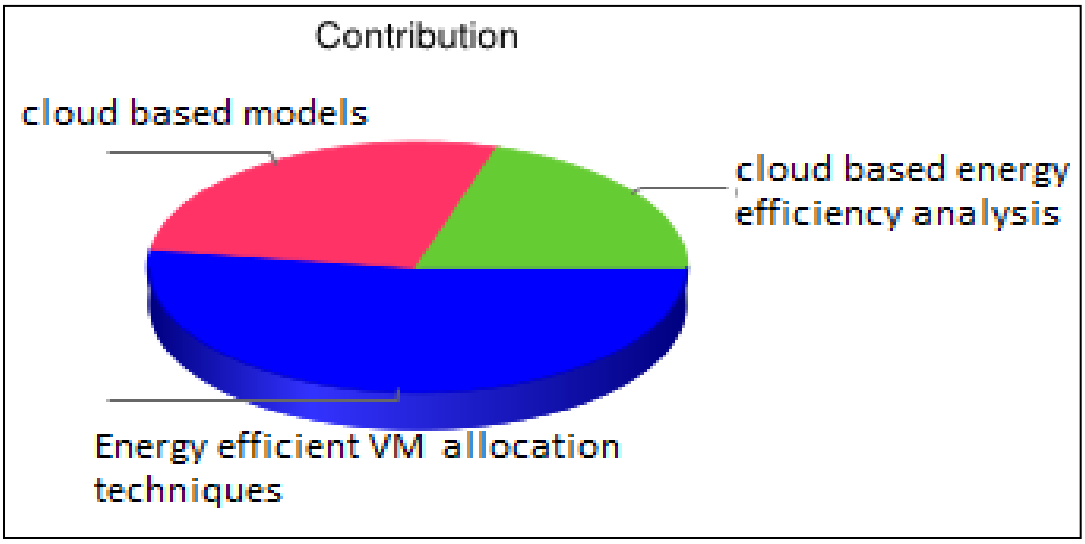


Fig 2.5. Breakdown of Research Paper Category

Number of research publications made per year in above three categories is shown in Figure 2.6. Based on Figure 2.6, research papers on Cloud based energy efficient virtual machine allocation techniques has the highest number of increase in publications as compared to other research paper categories. Overall, research publications on energy efficient virtual machine management in cloud have shown tremendous increase in the period of 2008 to Dec 2012. The following section will discuss findings of the study process for each area as categorized in Figure 2.5

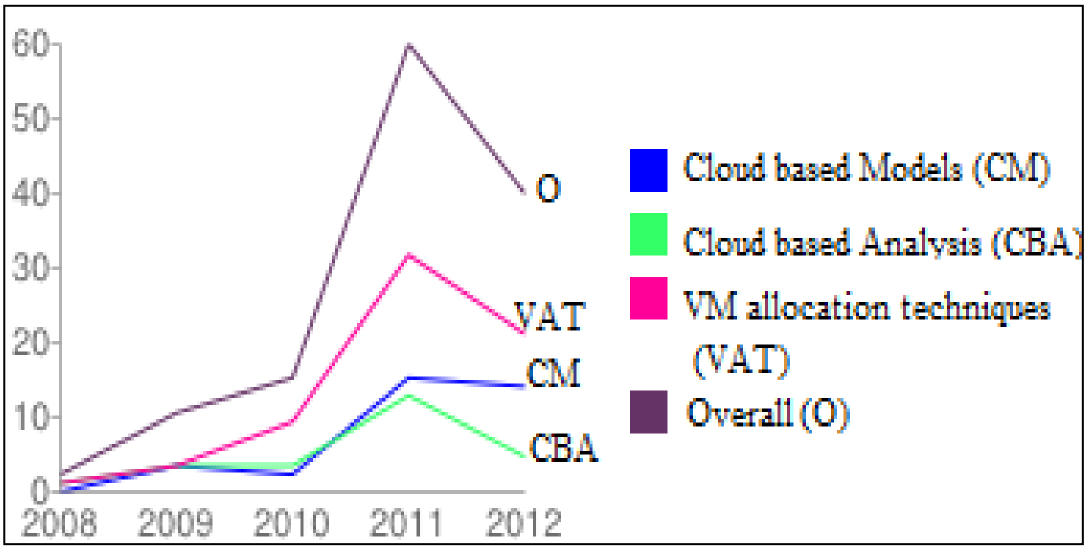


Fig 2.6. Number of Research Publications per year in each category

2.3.1 Cloud Based Energy Conservation Models

Based on research articles gathered, cloud based energy conservation models can be viewed from three different prospective:

- Energy efficient resource management i.e. failure aware resource provisioning,
- Power optimization in data centers,
- Estimation of energy consumption by in-processor activities,
- Energy efficient VM scheduling considering multi-facet parameters.

Various Cloud based energy conservation models are described in Table 2.1. In order to analyze these energy conservation approaches, following questions were defined:

(i) What are the issues addressed for energy conservation in cloud based structure?

(ii) Does cloud based models deal with power-performance trade-off?

Answers to above questions are described in Table 2.1 as mentioned by other authors. Eight authors [45][61][46][36][39][51][60][84] have addressed energy efficient and failure aware resource provisioning issues. Similarly five [58][53][57][71][37] authors dealt with power optimization issues and power estimation approaches are proposed by three authors [23][25][24]. Issues related to power-performance trade-off during VM scheduling are addressed by six authors [106][114][119][125][129][29] and few more authors [59][42][42][28][34][103][105][79] discussed other relevant energy conservation tactics. In order to understand implementation of energy conservation approaches listed in Table 2.1, cloud based energy efficiency analysis reports were collected which are discussed in next section.

Table 2.1. Cloud Based Energy Conservation Models

Category	**Author**	**Year**	**Issues Addressed**
Resource Management Model	Bahman Javadi [45]	2012	Failure aware resource provisioning for hybrid cloud
	Anshul Gandhi [61]	2012	Minimizing data center SLA violations and power consumption
	Fotis Aisopos [46]	2011	Use of knap sack problem model in SaaS
	Rajkumar buyya [36]	2011	QoS expectations and power usage characteristics of devices
	Dang Minh Quan [39]	2011	Heterogeneous hardware and workload , average load rate
	Timothy Wood [51]	2009	Sandpiper: Black-box and gray-box approach
	Tung Nguyen [60]	2012	Opera: Open reputation based Resource selection model
	Yunfa Li [84]	2010	Control on QoS for resource scheduling in multiple VM system
Power Management Model	Huichen [58]	2012	Fine-grained power management using process-level profiling
	Shuyi Chen [52]	2011	Using CPU gradients for energy conservation in multitier systems
	Keqin Li [57]	2011	Optimal power allocation among multiple heterogeneous servers
	Dusit Niyato [71]	2009	Server operation mode selection using constrained Markov decision process
	Luciano Bertini [37]	2009	Power optimization for dynamic configuration in heterogeneous web servers
Energy Estimation Model	Nakkukim [23]	2012	Estimation based on in-processor events generated by virtual machine

	Andreas Berl [25]	2011	Energy consumption of hosts and network within virtualized and ordinary office environments
	Ramon Bertran [24]	2011	PMC based power modeling method
Virtual Machine Scheduling Model	Nguyen Trung Hieu [106]	2011	Supporting real-time applications for U-Life care
	Mohammad Moein Taheri [114]	2011	2-phase optimization of VM scheduling
	Thomas Setzer [119]	2012	Considering migration overhead while VM re-assignment
	Stelios Sotiriadis [125]	2012	Dynamic instantiation of VMs in relation to current job characteristics
	Fei Wu [129]	2012	Deadline partitioning fair dynamic voltage frequency scaling
	B. Daughouty [29]	2011	Auto-scaling VM configuration
Others	Marco Bessi [59]	2012	Dynamic memorization for energy efficiency in financial applications
	Xiaoying Wang [42]	2012	Adaptive model-free resource-power management in multi-tier cloud
	Danilo Ardagna [43]	2012	Dual time-scale distributed capacity allocation and load redirection
	Zhenghua Xue [28]	2011	Characterization of service process of isolated versus multi-cluster system
	A.Kertesz [34]	2012	To ease interoperable service execution in heterogeneous cloud environment
	Hyunjeong Lee [103]	2011	Profile-based building energy saving service
	Cecile Germain Renaud [105]	2011	Data curation approach for green IT
	Haifeng Chen [79]	2012	Virtualization and consolidation analysis engine for large scale data centers

2.3.2 Cloud based Energy Efficiency Analysis

Several authors have analysed cloud with respect to energy consumption, energy aware performance of cloud, impact of green practices in cloud, comparison between available virtualization technologies etc. A range of articles with this respect are described in Table 2.2. On the basis of Table 2.2, it is identified that:

- Shajulin Benedict [50], V.A.Korthi Kanti [55], Robert R. Harmon [66] and T.V.Truong Duy [70] have done energy-performance trade-off analysis of green practices in cloud based architectures.

- Yubin Xia [64], Ying Song [68], Ryan Jansen [96], S Zhuravlev [108], Yichao Jin [115], N Regola [133] and C.R Chang [90] have analyzed various virtual machine allocation techniques and their effects.
- M.Witkowski [49], A Corradi [94] and Y.T Lee [75] have analyzed real life high performance computing applications against power conservation techniques.
- Principled Tech. [69] and Vmware [121] identified that Vmware vsphere is 5.4 times faster than Hyper-V live Migration and how it can reduce power consumption in IT infrastructure.
- Citrix [40] and Dell-Intel [48] discussed server consolidation without compromise and oracle [83] defined an optimized and cost effective data center management strategy.

Table 2.2. Cloud Based Energy Efficiency Analysis

Author	**Year**	**Issues Addressed**
M.Witkowski [49]	2012	Practical power consumption estimation for real life HPC
Shajulin Benedict [50]	2012	Energy aware performance analysis methodologies for HPC architectures
V. A. Korthi Kanti [55]	2011	Energy-performance trade-off analysis of parallel algorithms for shared memory architectures
Yubin Xia [64]	2008	Analysis and enhancement for iterative-oriented VM scheduling
Robert R. Harmon [66]	2009	Assessing the impact of green computing practices
Ying Song [68]	2009	Utility analysis of internet-oriented server consolidation
T. V. Truong Duy [70]	2010	Performance evaluation of green scheduling algorithms for energy savings in cloud computing
Yeng-Ting Lee [75]	2010	Benefits of server consolidation to MMORPG(world of warcraft)
Chao-Rui Chang [90]	2011	Empirical study on memory sharing of VMs for server consolidation
Antonio Corradi [94]	2011	IBM project on power saving through server consolidation
Ryan Jansen [96]	2011	Analysis of real-world VM allocation policies in cloud
Sergey Zhuravlev [108]	2012	Survey of energy-cognizant scheduling techniques
Yichao Jin [115]	2012	Trade-off between energy saving and detrimental effects from server virtualization
Nathan Regola [71]	2010	Recommendations for virtualization technologies in HPC
Principled Tech. [69]	2011	VM migration comparison between Vsphere and Hyper-V
Citrix [40]	2011	Consolidation without compromise
Dell-Intel [48]	2011	Virtualization: consolidation to cloud foundation
Oracle [83]	2011	Optimized and cost effective data center strategy
Vmware [121]	2011	How vmware virtualization right-sizes IT infrastructure to reduce power

		consumption
Pingpeng Yuan [67]	2009	Evaluating dynamic performance of VMM in server consolidation
George Kousiouris [41]	2011	Effects of scheduling, workload type & consolidation on VM performance
Ho-Leung Chan [54]	2011	Energy-throughput trade-off for online deadline scheduling

2.3.3 Energy-aware virtual machine allocation techniques

Allocation of VMs can be divided in two: the first part is admission of new requests for VM provisioning and placement of VMs on hosts, whereas the second part is optimization of current allocation of VMs. As the largest operating cost incurred in Cloud data center is for operating servers, it acts as a motivation behind power- aware VM scheduling in Cloud data centers. In power-aware scheduling [17], jobs are scheduled to nodes in such a way to minimize the server's total power. Figure 2.7 shows a power consumption curve, which illustrates that as the number of processing cores increases, the amount of energy used does not increase proportionally. In fact, change in power consumption decreases. When using only one processing core, change in power consumption incurred by using another processing core is over 20 watts. The change from 7 processing cores to all 8 processing cores results in an increase of only 3.5 watts [17].

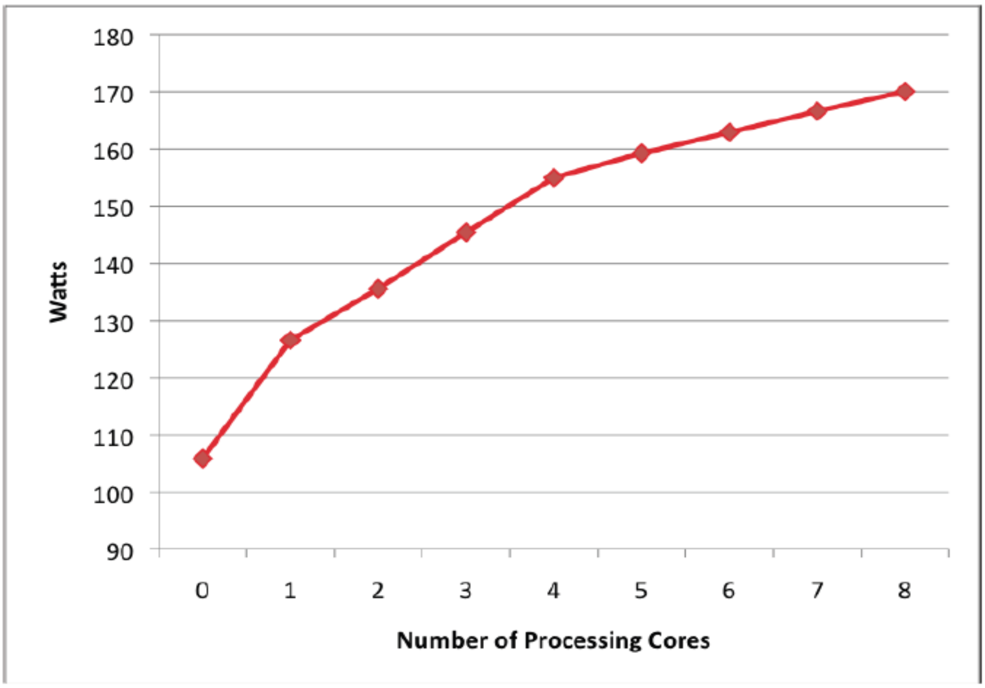


Fig. 2.7. Power consumption curve of an Intel Core i7 920 CPU [17]

Impact of this finding is substantial. In a normal round robin VM scheduling system like the one in Eucalyptus, load of VMs is distributed evenly to all servers within the data center. While this may be a fair scheduler, in practice it is very inefficient because each time the scheduler distributes VMs to a processor, power consumption increases by its greatest potential. In contrast, power-aware scheduling system demonstrates that if the scheduler distributes VMs with the intent to fully utilize all processing cores within each node, power consumption is decreased dramatically [17].

To complete the study of VM scheduling system, major energy efficient virtual machine allocation techniques are evaluated. Energy efficient VM allocation policies addresses following criteria [16]:

(i) First admission of new requests for VM provisioning and placement of VMs on hosts, i.e virtual machine scheduling and

(ii) Second optimization of current allocation of VMs, i.e. virtual machine consolidation.

Table 2.3 describes authors involved, approaches and specifications used for addressing above two criteria while allocation of VMs to new requests. Table 2.3 summaries numerous research articles within year 2008-2012 addressing issues related to VM consolidation i.e. optimization of current allocation of VMs such as load balancing, migration control, bounded migration cost and Bin-Item and Item-Item incompatibility constraints etc, and issues related to virtual machine scheduling i.e. placement of VMs on hosts such as scheduling delays, dynamic task scheduling, slacking non-critical jobs and elastic scheduling.

Table 2.3. Energy-aware virtual machine allocation techniques

Category	**Author**	**Year**	**Issues Addressed**
Virtual Machine Consolidation	Tiago C. Ferreto [27]	2011	Migration control by prioritizing virtual machines with steady capacity
	Mario Fanelli [32]	2012	Power consumption, host resources and networking
	Lanzheng Liu [117]	2012	Dynamic memory scheduling
	Rohit Gupta [63]	2008	Item-Item and Bin-Item incompatibility constraints
	Deshi Ye [65]	2009	Load balancing
	Michael Cardosa[72]	2009	Shares and utilities based power consolidation
	Jae-Wan Jang [74]	2011	Memory aware consolidation
	Kejiang Ye [76]	2010	Improving isolation property of server consolidation
	Zhenhuan Gong [77]	2010	PAC: Pattern driven application consolidation
	Daniel Versick [81]	2010	Load aggregation with dynamic live migration of

			VMs
	S. Kumar Bose [88]	2011	Live migration of VMs across WANs
	M. Marzolla [95]	2011	Gossiping
	Kejiang Ye [97]	2011	Resource reservation while live migration of multiple VMs
	A. G. Guirado [100]	2011	Cache coherence in chip-multiprocessors
	Yufan Ho [107]	2011	Bounded migration cost and performance guarantees
	Ching-Hsien Hsu[110]	2011	Task consolidation
	Shyam K. Doddavula [111]	2011	Implementation of fast vector packing and its application for server consolidation
	Sanghwan Lee [112]	2011	Intra-cluster traffic
	Haikun Liu [26]	2011	Runtime VM mapping under constraint of multi-facet resource consumptions
	Brototi Mondal [62]	2011	Load balancing using stochastic hill climbing-a soft computing approach
	M. Mazzucco [56]	2011	Parameters leading to highest revenues
	Jian Li [123]	2012	Application consolidation with resource prediction
	Kishaloy Halder [127]	2012	Risk aware provisioning and resource aggregation based consolidation
	R. Buyya [128]	2012	Host overload detection
	Ali Pahlavan [131]	2012	Variation aware server placement and chassis consolidation
	Ligang He [33]	2012	Resource consolidation of moldable VM
Virtual Machine Scheduling	Hyunsik Choi [73]	2009	VM scheduling delays in Xen
	Linwei niu [78]	2010	Pattern variation and dynamic slack reclaiming extensions
	Huacai Chen [80]	2010	Affinity aware proportional share scheduling
	Hidekazu Tadokoro [82]	2010	Denial-of-Service attacks
	Stylianos Zikos [35]	2010	Compute intensive jobs with unknown service times
	Wei Yan [47]	2010	Dynamic task scheduling with fuzzy prediction
	Wan Yeon Lee [86]	2012	Periodic real-time tasks scheduling on lightly loaded multicore processors
	Fanxin Kong [85]	2011	Energy efficiency for system with and without timing constraints
	Linwei Niu [87]	2011	Speed determination and task pre-emption management
	Xiao Jun Chen [89]	2011	Time slices adjustment methodology in VMs

	Ching-Hsien Hsu [91]	2011	Contention-free scheduling and dynamic voltage adjustment
	Dong Song Zhang [92]	2011	Global earliest deadline first scheduling for hard real time tasks
	Stefan Videv [93]	2011	Bandwidth-Energy efficiency trade-off with low load
	Ching-Chi Lin [98]	2011	Dynamic round-robin scheduling and consolidation
	Harold Castro [99]	2011	Flexible opportunistic computing with virtualization
	Dong Song Zhang [101]	2011	Scheduling of sporadic real-time tasks
	Wei Liu [102]	2011	Dynamic threshold based task scheduling on homogenous DVS-enabled clusters
	Xiaomin Zhu [104]	2011	Elastic scheduling for aperiodic tasks on heterogeneous DVS-enabled clusters
	Balazs Gerofi [109]	2011	Workload adaptive checkpoint scheduling of VM replication
	Jia Tian [113]	2011	SMP virtual machine scheduling
	R Jeyarania [30]	2011	Adaptive provisioning using swarm intelligence
	Sayan Chakrabarti [116]	2012	Backfill strategy based light weight VM scheduler for dispatching jobs
	Lina Sawalha [118]	2012	Phase-aware scheduling for heterogeneous multicore processors
	Hadi Goudarzi [120]	2012	Aforesaid resource allocation problem
	Qingjia Huang [122]	2012	Slacking non-critical jobs
	Pedro Eugenio Rocha [124]	2012	Non-work conserving disk scheduling guarantying high throughput
	Xiao Ling [126]	2012	Allocating disk resources ensuring specified latency and throughput targets
	Jianjun Li [130]	2012	Non-preemptive scheduling ensuring timing correctness
	Shin-Gyu Kim [132]	2012	Effects of shared on-chip last level cache interference
	Inigo Goiri [31]	2012	Virtualization overheads, SLA violation penalties, support for outsourcing
	Xiaomin Zhu [44]	2012	Adaptive dynamic voltage scaling in heterogeneous clusters

Various VM allocation techniques mentioned in the Table 2.3, manage both homogeneous and heterogeneous workloads but existing virtual machine allocation techniques incur significant SLA violations and VM migrations. Also, existing heterogeneous workload consolidation techniques are energy efficient for the Cloud computing platform to some extent only. In next section, this problem investigated is illustrated and a solution is proposed to deal with power-performance trade off in Cloud infrastructure.

Aim of this thesis is, power-aware scheduling of virtual machines in virtualized heterogeneous systems. This is accomplished by consolidating heterogeneous workloads in an efficient way so that resource utilization is maximized and energy consumption of the data center is minimized resulting in lesser number of carbon footprints and hence assisting in achieving Green Computing.

Chapter 3
Problem Analysis

Previous chapter depicted major energy efficient virtual machine allocation policies in Cloud infrastructure. This chapter focuses on gaps found while literature review of various VM allocation techniques.

3.1 Gap Analysis

Recently emerged Cloud computing paradigm leverages virtualization and provides on-demand resource provisioning over the Internet on a pay-as-you-go basis. This allows enterprises to drop costs of maintenance of their own computing environment and out-source their computational needs to the Cloud. It is essential for Cloud providers to offer reliable Quality of Service (QoS) to customers, which is negotiated in terms of Service Level Agreements (SLA), e.g. throughput, response time. Therefore, to ensure efficient resource management and provide higher utilization of resources, Cloud providers (e.g. Amazon EC2) have to deal with power-performance trade-off, because aggressive consolidation of VMs can lead to performance loss.

Based on the literature survey there are various resource management techniques (as shown in the Table 3.1), which manage virtual machine scheduling in virtualized heterogeneous environment, but these techniques are energy efficient to some extent only as they provide high performance but render significant SLA violations and also do not focus on optimization of current allocation of VMs to minimize energy consumption for the Cloud computing platform.

Table 3.1: VM scheduling policies for virtualized heterogeneous environment

Technique (Year)	Power Saving Technique	System Resources	Workload Type	Findings
(i) Adaptive energy-efficient scheduling for real-time tasks on DVS-enabled heterogeneous clusters (2012)	DVFS	CPU	(i) Aperiodic tasks (ii) Independent real-time tasks	Reduces the voltage levels to conserve energy while maintaining higher guarantee ratios.
(ii) Phase-Aware Scheduling for Heterogeneous	Dynamic application to core mapping	CPU, Memory	(i) HPC jobs	Energy reduction over random scheduling of

Multicore Processors (2012)				programs within a heterogeneous multicore processor.
(iii) Energy-Efficient Elastic Scheduling in Heterogeneous Computing Systems (2011)	DVFS	CPU	(i) Aperiodic tasks (ii) Independent tasks	(i) significantly improves the scheduling quality (ii) Effectively enhances the system elasticity.
(iv) Multifaceted Resource Management for Dealing with Heterogeneous Workloads in Virtualized Data Centers (2010)	VM consolidation	CPU	(i) HPC jobs (ii) Web based applications	(i) Provides 15% benefit to the provider. (ii) Reduces power.
(v) A Dynamic MapReduce Scheduler for Heterogeneous Workloads (2009)	VM consolidation	CPU, Memory	(i) CPU bound (100%utilization) (ii) CPU bound without shuffle) (iii) I/O bound	Improve Hadoop throughput by 30%.
(vi) Managing SLAs of Heterogeneous Workloads using Dynamic Application Placement (2008)	DVFS, VM Consolidation	CPU	(i) Transactional Applications (ii) Long Running Jobs	Maximized performance of heterogeneous Workloads
(vii) Phoenix Cloud: Consolidating Heterogeneous Workloads of Large Organizations on Cloud Computing Platforms (2008)	Server Consolidation	CPU	(i) Parallel Batch Jobs (ii) Web Service	Increased no. of completed jobs
(viii) Server virtualization in Autonomic management of heterogeneous workloads (2007)	Server Consolidation, VM consolidation	CPU	(i) Web Application, (ii) Non interactive workloads	(i) Heterogeneous workloads can be allocated on any server machine. (ii) Used high level performance goals for resource allocation.

3.2 Problem Statement

Workload and resource management are two essential functions provided at the service level of Cloud infrastructure. Virtualization technologies, on which Cloud computing environments heavily rely on, provide the ability to transfer VMs between physical nodes using live or offline migration. This enables dynamic consolidation of VMs to a minimal number of nodes according to current resource requirements. As a result, idle nodes can be switched off or put to a power saving mode (e.g. sleep, hibernate) to reduce total energy consumption by the data center.

The problem is to determine how VMs should be placed on hosts that will provide the most efficient overall usage of resources, while ensuring reliable Quality of Service (QoS) and minimal energy consumption. Current approaches for energy efficient virtual machine scheduling in Cloud data centers do not investigate the problem of scheduling of VMs in heterogeneous infrastructure. Most of the approaches usually focus on one particular workload type and homogeneous infrastructure.

3.2.1 Objectives

Aim of this research work is to schedule VMs in virtualized heterogeneous environment in such a way that:

- Resource utilization is maximized, and
- SLA violations and energy consumption in the data center are minimized, resulting in lesser number of carbon footprints.

In this work, we leverage live migration of VMs and propose a heuristic for dynamic reallocation of VMs according to current resource requirements, while ensuring reliable QoS. The objectives of reallocation are:

- To minimize the number of physical nodes serving current workload, and
- To switch off idle nodes in order to decrease power consumption.

Chapter 4

Proposed VM Scheduling Technique

This chapter discusses about how the problem stated in previous chapter can be solved with the help of the proposed virtual machine scheduling technique. This chapter also describes the design of the proposed technique along with its complexity and architectural details.

4.1 Design of Solution

Solution to the problem (Power-aware VM scheduling in virtualized heterogeneous environment) has been explained in this section along with outline of the proposed technique and its architectural details.

4.1.1 Outline of the Proposed Technique

This section presents background information on various architectural elements that form the basis for implementation of VM scheduling policy in Cloud environment. Figure 4.1 shows the layered design of service-oriented Cloud computing architecture. Physical Cloud resources along with core middleware capabilities form the basis for delivering IaaS. The user-level middleware aims at providing PaaS capabilities. The top layer focuses on application services (SaaS) by making use of services provided by the lower layer services. PaaS/SaaS services are often developed and provided by 3rd party service providers, who are different from IaaS providers [133].

- User-Level Middleware

This layer includes the software frameworks such as Web 2.0 Interfaces (Ajax, IBM Workplace) that help developers in creating rich, cost effecting user-interfaces for browser-based applications. The layer also provides the programming environments and composition tools that ease the creation, deployment, and execution of applications in Clouds [18].

- Core Middleware

This layer implements the platform level services that provide runtime environment enabling Cloud computing capabilities to application services built using User-Level Middlewares. Core services at this layer include Dynamic SLA Management, Accounting, Billing, Execution monitoring and management, and Pricing. The well-known examples of services operating at this layer are Amazon EC2, Google App Engine, and Aneka [18].

- System Level

The computing power in Cloud computing environments is supplied by a collection of data centers, which are typically installed with hundreds to thousands of servers. At the System Level layer there exist massive physical resources (storage servers and application servers) that power the data centers [18]. These servers are transparently managed by the higher level virtualization services and toolkits that allow sharing of their capacity among virtual instances of servers. These VMs are isolated from each other, which aid in achieving fault tolerant behaviour and isolated security context [13].

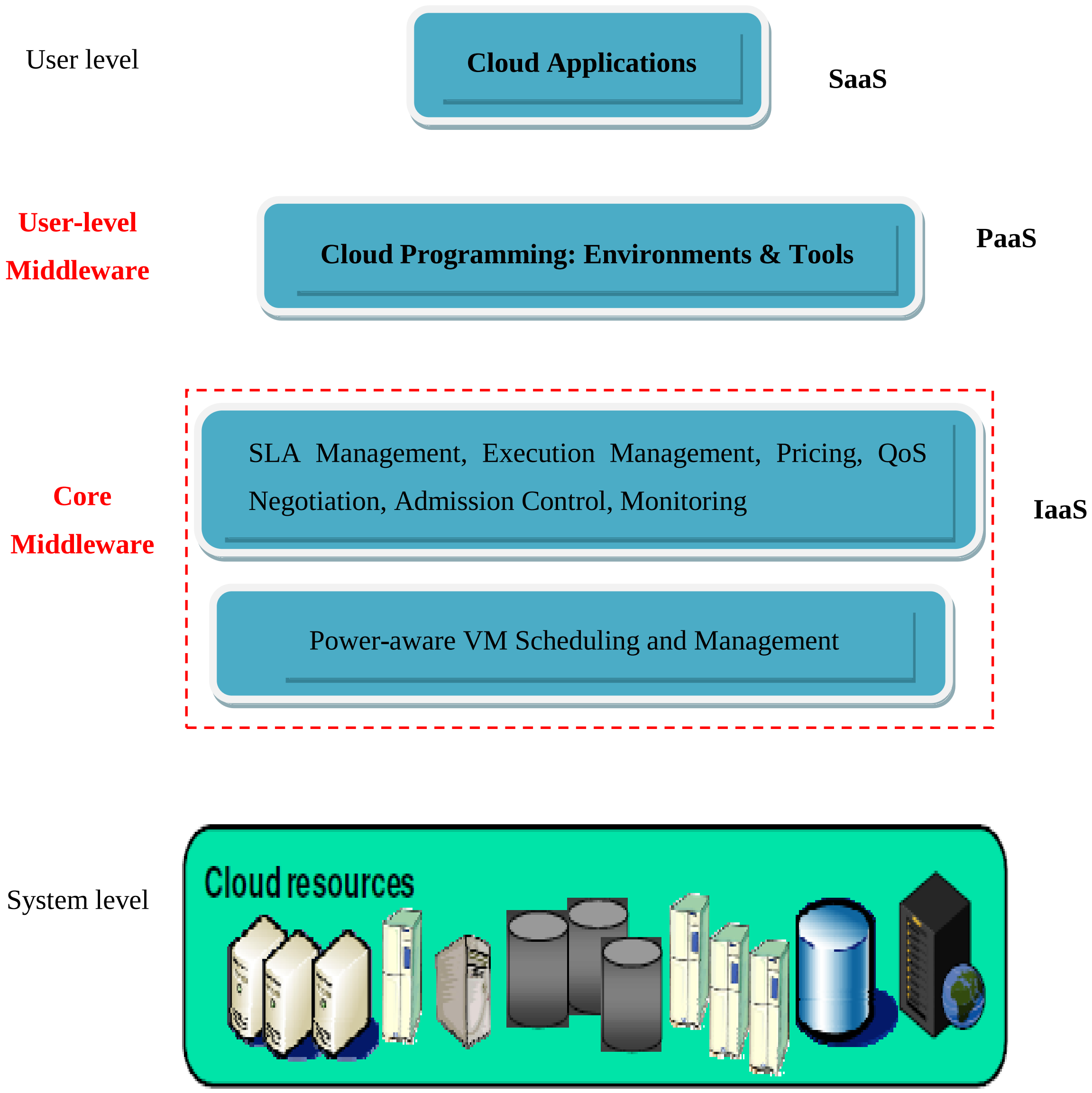


Figure 4.1: Layered Cloud Computing Architecture

From Figure 4.1, it can be concluded that Core Middleware is responsible for management and deployment of VMs. It also calculates the energy consumption of the Cloud data center and also gives the information about SLA violations as VM allocation policies are implemented at this layer. Therefore, the proposed VM scheduling policy should be implemented at this layer.

4.1.2 Design of the Proposed Technique

In particular, the proposed approach is an optimized method for efficient server allocation. In the proposed policy, an existing threshold based VM scheduling policy is optimized in a manner to further reduce SLA violations and energy consumption. In existing VM scheduling policy i.e Single Threshold (ST) policy [36], VM scheduling is carried out by setting upper utilization threshold for hosts and placing VMs on hosts keeping the total utilization of CPU below the stated threshold. The pseudo-code for ST policy is described in Algorithm 1 [36].

Algorithm 1: Existing Threshold based policy

```
Input: hostList, vmList Output: vmAllocationList
foreach h in hostList do
    vmList←h.getVmList()
    hUtil←h.getUtil()
    while hUtil < THRESH_UP do
       foreach vm in vmList do
          if vm.getUtil() + hUtil <  THRESH_UP then
          hUtil←vm.getUtil() + hUtil
      allocatedVm ← vm
       break
  hostList.add(h)
  vmList.remove(allocatedVm)
return vmAllocationList
```

This threshold based VM scheduling policy helps in preventing SLA violation and results in minimum possible energy consumption. But further energy consumption can be reduced by dynamic reallocation of VMs according to current resource requirements, while ensuring reliable QoS.

The proposed VM scheduling policy is based on the idea of dynamic reallocation of VMs to reduce energy consumption and SLA violation. The intuition behind our approach is that, first place VMs on host while keeping total utilization of CPU below defined threshold and then optimizing VM allocation by constrained migration of VMs with host utilization below the lower utilization threshold, leveraging live migration and switching off idle nodes to minimize power consumption.

As excessive migration of VMs can lead to significant SLA violation hampering the required Quality of Service, the proposed approach can assist in correctly allocating server resources in data center by optimal VM allocation and constrained migration of VMs, resulting in least SLA violations and minimum energy consumption. Constrained migration of VMs is achieved in the proposed policy by defining minimum CPU utilization threshold for hosts and migrating only those VMs which are having host utilization below defined minimum CPU utilization threshold.

Architecture for the proposed VM scheduling approach is shown in Figure 4.2. Figure 4.2 elaborates the procedure for:

- Admission of new requests for VM scheduling.
- Placement of VMs on hosts.
- Optimization of current allocation of VMs.
- Conditions for placement of VMs on hosts
- Pre-requisite for optimizing current VM allocation.

In the proposed architecture, VM allocation procedure is carried out in a manner as described below:

(i) First, VMs and hosts are created and initiated for undertaking user's requests. Then user defined workload is identified for VM scheduling and placement of VMs on hosts.

(ii) Then CPU utilization threshold is defined for hosts to prevent SLA violation due to consolidation in cases when resource requirements by VMs increase. All the VMs in VM list will be placed on hosts keeping host utilization below pre-defined CPU utilization threshold for hosts.

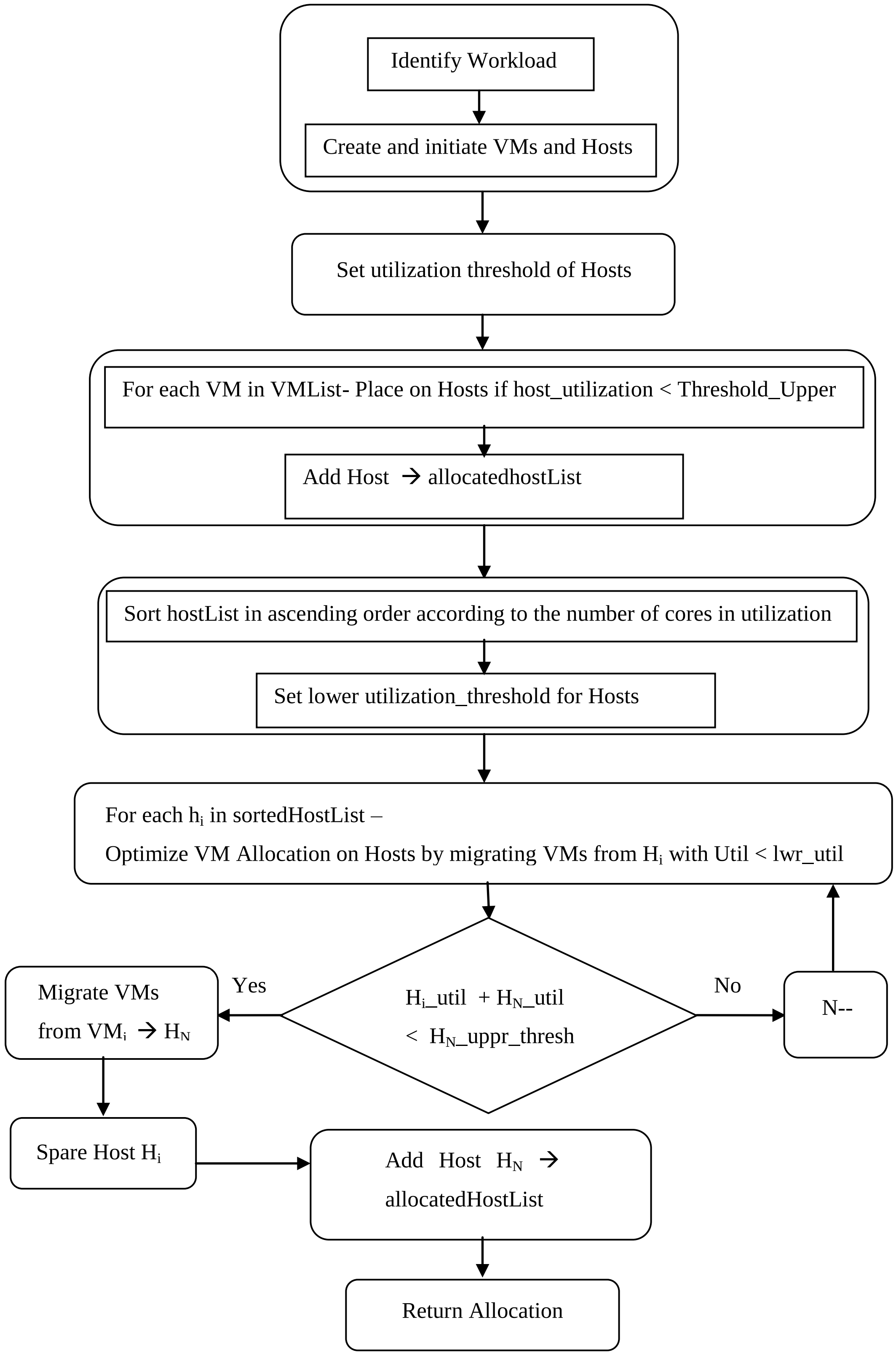


Figure 4.2: Flowchart of Proposed Technique

(iii) When all the VMs are allocated, then the corresponding hosts are added to allocated hostlist. All the hosts in the allocated hostlist are sorted in ascending order according to the CPU utilization level. Hostlist is sorted to minimize optimization overhead.

(iv) After host list is sorted, lower CPU utilization threshold is defined for each host in the sorted hostlist. Lower CPU utilization threshold is set to constrain the VM migration procedure i.e. only those VMs will be migrated which have host utilization below stated CPU utilization threshold.

(v) Accordingly, all the hosts are checked for VM migration as per condition stated above. If VM migration is possible then VMs are migrated to the utmost host in the sorted hostlist keeping upper utilization threshold not violated.

(vi) Hosts from which VMs are migrated are added to list of spare hosts and hosts to which VMs are migrated are added to allocated host list. This procedure repeats till all the possible VM migrations are carried out and ends up by returning optimized VM allocation.

The proposed solution is divided two components, one for placement of VMs on hosts and other for optimization of VM allocation on hosts.

(i) The first component helps in preserving free resources by placing VMs below defined static threshold in order to prevent SLA violation due to consolidation in cases when resource requirements by VMs increase, and

(ii) The second component helps in reducing level of power consumption by migrating VMs from only those hosts which are having total CPU utilization below lower migration threshold and switching off the spare host to eliminate idle power consumption.

The pseudo-code for components of the proposed technique is presented in Algorithm 2and 3. Algorithm 2 sets an upper utilization threshold for hosts. Then, for each VM in the list of VMs, it repeatedly looks through the list of hosts and finds a host with total CPU utilization below static utilization threshold. Then chosen host is set as the allocated host and VM is placed on it. Increase in host power after allocation of VMs is added to the total power consumption.

This procedure is repeated for each VM in the VM list and algorithm stops when all of the VMs are placed on available hosts. Complexity of the algorithm is proportional to the product of the number of VMs that have to be allocated and the number of hosts available.

Algorithm 2: Initial Allocation Procedure

Input: hostList, vmList **Output**: vmAllocationList

hostList.setUpperThresholdUtilization ()

allocatedHost <- NULL

foreach vm *in* vmList **do**

  foreach host *in* hostList **do**

    if hostUtil < Thresh_Upper **then**

      if vmUtil() + hostUtil < Thresh_Upper **then**

      hostUtil←vmUtil() + hostUtil

       allocatedHost <- host

       TotalPower <-allocatedHostPower

      if allocatedHost != NULL **then**

        allocate vm to allocatedHost

return allocation

optimize vm allocation with Algorithm 3

Optimization of current allocation of VMs on hosts is done using Algorithm 3. In algorithm 3, first the list of hosts is sorted in ascending order according to total utilization of CPU by all VMs. Then lower utilization threshold also known as migration threshold is set to constrain migration of VMs. A counter equal to total number of hosts in sorted host list is set to start the optimization procedure. First, for each host *i* in sorted host list, if host utilization lies below migration threshold i.e. host is under-utilized, then VMs on that particular host are located for migration. Migration of VMs is first checked at host *j*, host with highest CPU utilization i.e. last host in sorted host list. Migration to the host *j* is done if total CPU utilization of host after VM consolidation remains below static upper threshold defined in Algorithm 3.

If total CPU utilization of host *j* will go across upper utilization threshold when VM consolidation will be done, then migration of VMs from under-utilized host to this particular host is not possible, therefore preceding host in sorted host list is checked for migration and counter is decreased by one. This procedure is carried out till all the VMs on host *i* are migrated to host *j*. This procedure repeats for all under-utilized hosts till host value *i* become equal to the value of counter i.e. when there is no host left to which migration of VMs of under-utilized hosts can be done. In this way, this approach constrains the migration of VMs, such that SLA violations are minimized and required Quality of service is attained.

Algorithm 3: Optimization procedure

```
Input: vmAllocationList, hostList Output: migrationList
sort hostList according to CPU utilization in ascending order and set Migration threshold
i=1, j=N
foreach h_i in sortedHostlist do
   while h_i.util_rate < Migration_Thresh and i<j do
      if h_i.util_rate + h_j.util_rate < h_j.Thresh_Upper then
         migrate vm_i to h_j
         sparehostList.add(h_i)
         usedhostList.add(h_j)
         break
      else
         j= j-1;
return migrationList
```

Implementation view of pseudo-code for proposed solution is shown in Figure 4.3. Implementation view comprises of following elements:

(i) System Nodes or Hosts

(ii) Monitor Node or Core Middleware

(iii) User Code layer or User Interface

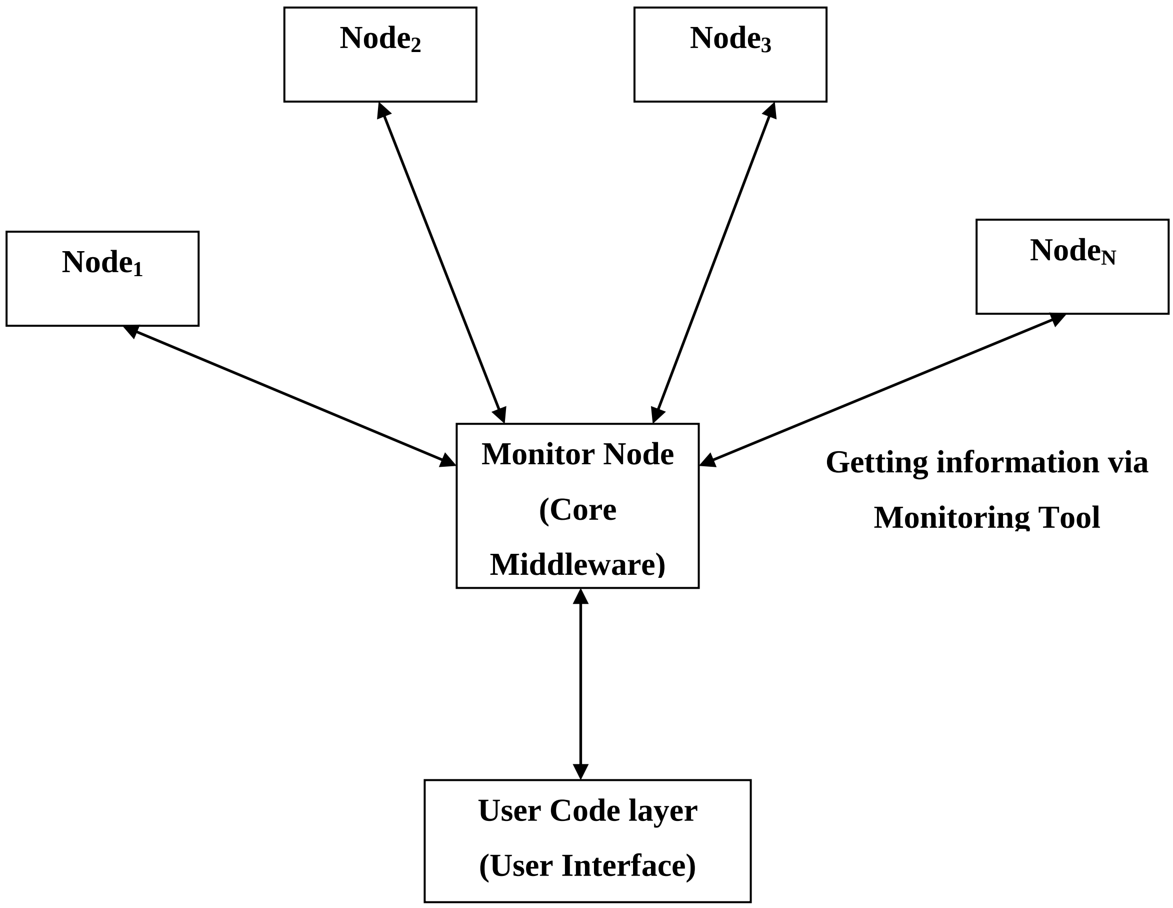


Figure 4.3: Implementation view of Pseudo-code

System nodes get all information regarding VM scheduling and placement of VMs on nodes via Monitoring node i.e. Core Middleware. Core Middleware is responsible for handling all the user's requests for VM scheduling, VM management and placement of VMs on hosts according to defined VM scheduling policy. Core Middleware also helps in power management and SLA violation control. All the conditions for VM allocation are implemented at Core Middleware.

User interacts with Core Middleware via User Code layer i.e. User Interface. User code layer is responsible for admission of new user requests for VM scheduling and placement of VMs on hosts. User Code layer exposes configuration related functionalities for hosts such as number of machines, their specification and so on, and similarly for applications such as number of tasks and their requirements, VMs, number of users and their application types. A Cloud application developer can generate mix of user request distributions, application configurations and Cloud availability scenarios at this layer.

4.1.3 Complexity of the Proposed Technique

Complexity is a measure of the performance of an algorithm in terms of CPU time and memory usage. In this case, computational complexity has been considered as this technique is for Cloud environment.

Complexity of the allocation part of the algorithm is $n \cdot m$, where n is the number of VMs that have to be allocated and m is the number of hosts.

In next section, the proposed heuristic has been evaluated and evaluation results have been presented, validating substantial energy savings with the implementation of proposed VM allocation policy.

Chapter 5

Experimental Results

This chapter focuses on tools for setting up the Cloud environment, implementation of the proposed power-aware VM scheduling policy for heterogeneous infrastructure and experimental results, justifying substantial energy savings by implementation of the proposed approach.

5.1 Tools for setting Cloud Environment

Cloud applications have different composition, configuration and deployment requirements. Quantifying the performance of scheduling and allocation policies in a real Cloud environment for different application and service models under different conditions is extremely challenging. The use of real infrastructures such as Amazon EC2, limits experiments to the scale of the infrastructure, and makes the reproduction of results an extremely difficult undertaking. The main reason for this being the conditions prevailing in the Internet-based environments are beyond the control of developers of resource allocation and application scheduling algorithms.

An alternative is the utilization of simulation tools that open the possibility of evaluating the hypothesis prior to software development in an environment where one can reproduce tests. Specifically in the case of Cloud computing, where access to the infrastructure incurs payments in real currency, simulation-based approaches offer significant benefits to Cloud customers by allowing them to:

(i) Test their services in repeatable and controllable environment free of cost; and

(ii)Tune the performance bottlenecks before deploying on real Clouds.

At the provider side, simulation environments allow evaluation of different kinds of resource leasing scenarios under varying load and pricing distributions. Such studies could aid providers in optimizing the resource access cost with focus on improving profits. In the absence of such simulation platforms, Cloud customers and providers have to rely either on theoretical and imprecise evaluations, or on try and-error approaches that lead to inefficient service performance and revenue generation. Tools required to set up a simulated Cloud environment for implementation of the proposed VM scheduling policy are described as follows:

• CloudSim

CloudSim is an extensible simulation toolkit that enables modelling and simulation of Cloud computing systems and application provisioning environments. The CloudSim toolkit supports both system and behaviour modelling of Cloud system components such as data centers, virtual machines (VMs) and resource provisioning policies. It implements generic application provisioning techniques that can be extended with ease and limited effort. Currently, it supports modelling and simulation of Cloud computing environments consisting of both single and Inter-Networked Clouds (Federation of Clouds). Moreover, it exposes custom interfaces for implementing policies and provisioning techniques for allocation of VMs under Inter-Networked Cloud computing scenarios. CloudSim offers the following novel features [18]:

- Support for modeling and simulation of large-scale Cloud computing environments, including data centers, on a single physical computing node.
- A self-contained platform for modeling Clouds, service brokers, provisioning, and allocation policies.
- Support for simulation of network connections among the simulated system elements.
- Facility for simulation of federated Cloud environment that internetworks resources from both private and public domains, a feature critical for research studies related to Cloud-Bursts and automatic application scaling.
- Availability of a virtualization engine that aids in the creation and management of multiple, independent, and co-hosted virtualized services on a data center node.
- Flexibility to switch between space-shared and time-shared allocation of processing cores to virtualized services.

• CloudSim Architecture

Figure 5.1 shows the multi-layered design of the CloudSim software framework and its architectural components. CloudSim provides novel support for modeling and simulation of virtualized Cloud based data center environments such as dedicated management interfaces for VMs, memory, storage, and bandwidth.

CloudSim layer manages the instantiation and execution of core entities (VMs, hosts, data centers, application) during the simulation period. The fundamental issues such as provisioning of hosts to VMs based on user requests, managing application execution, and dynamic monitoring are handled by this layer. A Cloud provider, who wants to study the efficacy of different policies in allocating its hosts, would need to implement his strategies at this layer by programmatically extending the core VM provisioning functionality.

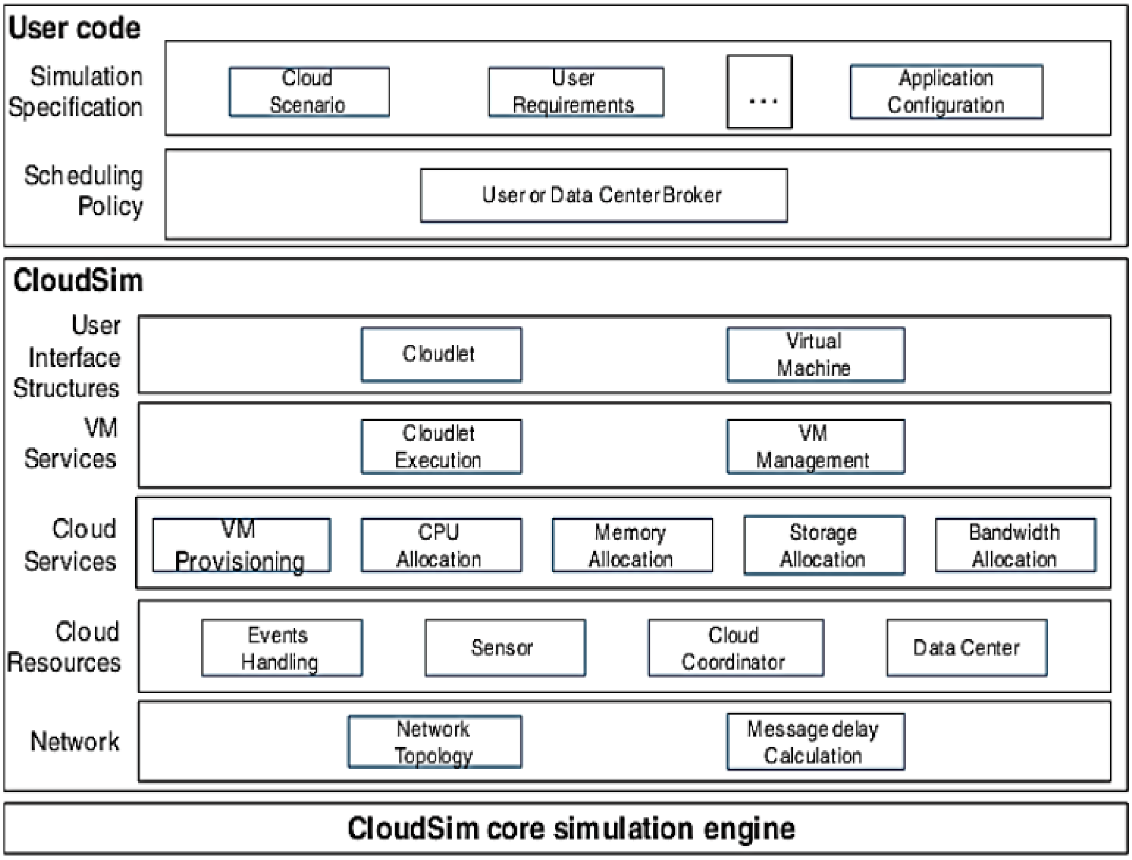


Figure 5.1: Layered CloudSim Architecture [18]

The top layer in the simulation stack is the User Code that exposes configuration related functionalities for hosts (number of machines, their specification and so on), applications (number of tasks and their requirements), VMs, number of users and their application types, and broker scheduling policies. A Cloud application developer can generate:

(i) A mix of user request distributions, application configurations; and

(ii) Cloud availability scenarios at this layer and perform robust tests based on the custom configurations already supported within the CloudSim.

• Eclipse

Eclipse is a multi-language software development environment comprising of an integrated development environment (IDE) and an extensible plug-in system [19]. It is used to develop applications in Java and other programming languages including Ada, C, C++, etc, or by means of various plug-ins, available in this software development environment for execution and development of new applications. The Eclipse SDK (which includes the Java development tools) is meant for Java developers. Users can extend its abilities by installing plug-ins written for the Eclipse Platform, such as development toolkits for other programming languages, and can write and contribute their own plug-in modules. The Eclipse SDK includes the Eclipse Java development tools (JDT), offering an IDE with a built-in incremental Java compiler and a full model of the Java source files. As the CloudSim Simulation toolkit is developed on Java Platform, so Eclipse SDK is used to create simulated Cloud environment.

5.2 Implementation and Evaluation Work

The proposed virtual machine scheduling approach has been implemented in CloudSim toolkit by programmatically extending the core VM scheduling functionality provided at core CloudSim layer of CloudSim simulation engine. For implementation of new approach, some changes are also done at the top layer i.e. User Code layer of CloudSim simulation engine. For execution of this simulation engine, Eclipse SDK is used.

5.2.1 Implementation Details

Implementation of any approach in CloudSim software framework is accomplished by modifying the core functionalities available at:

(i) The top layer i.e. User Code layer, and
(ii)The core layer i.e. CloudSim layer

Implementation of power-aware virtual machine scheduling technique for Cloud is explained with the help of snapshots as shown below:

(i) Implementation at User Code Layer

The simulated data center comprises of 200 heterogeneous physical nodes. Each node is modelled to have dual CPU core with performance equivalent to 1860 or 2660 MIPS, 4 GB of RAM and 1 TB of storage. Users submit request for scheduling of 310 heterogeneous VMs that fills the full capacity of the data center.

At first, host data center is created for provisioning of users' requests. Figure 5.2 and Figure 5.3 illustrate step by step creation of host data center, along with its architectural details.

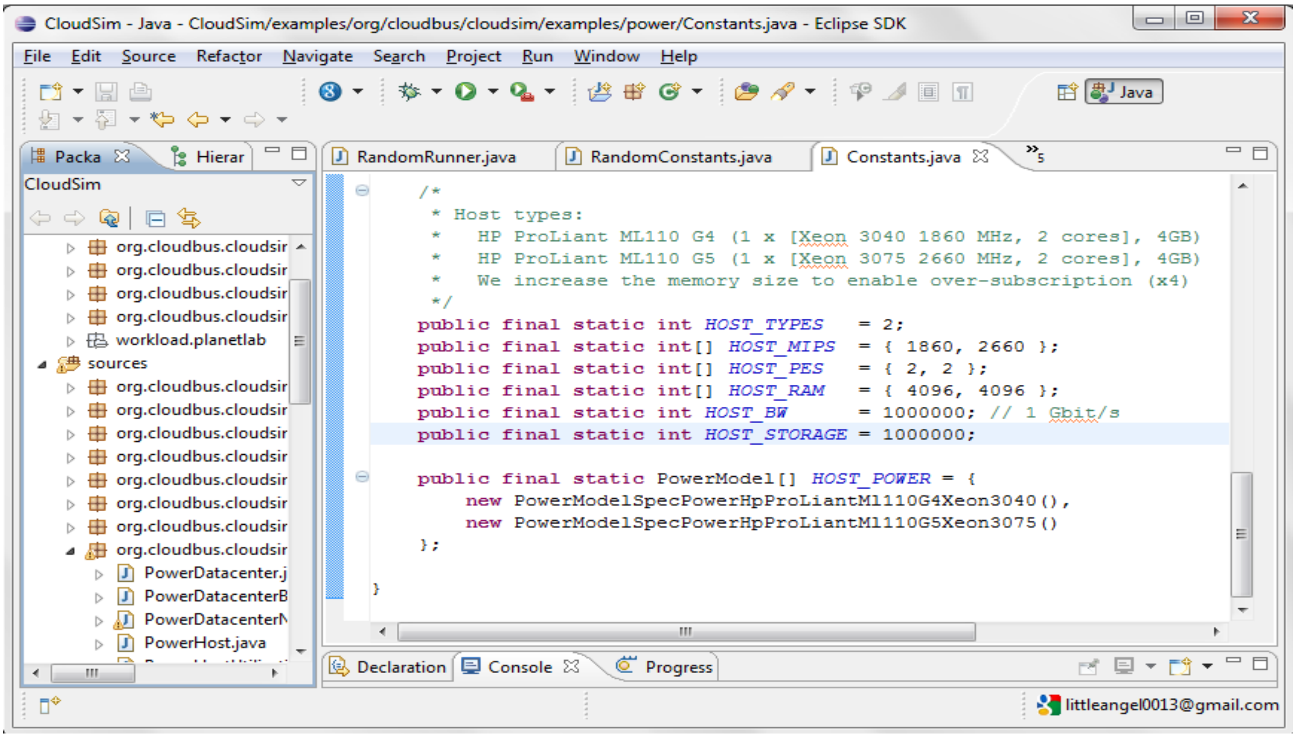


Figure 5.2: Creation of host data center

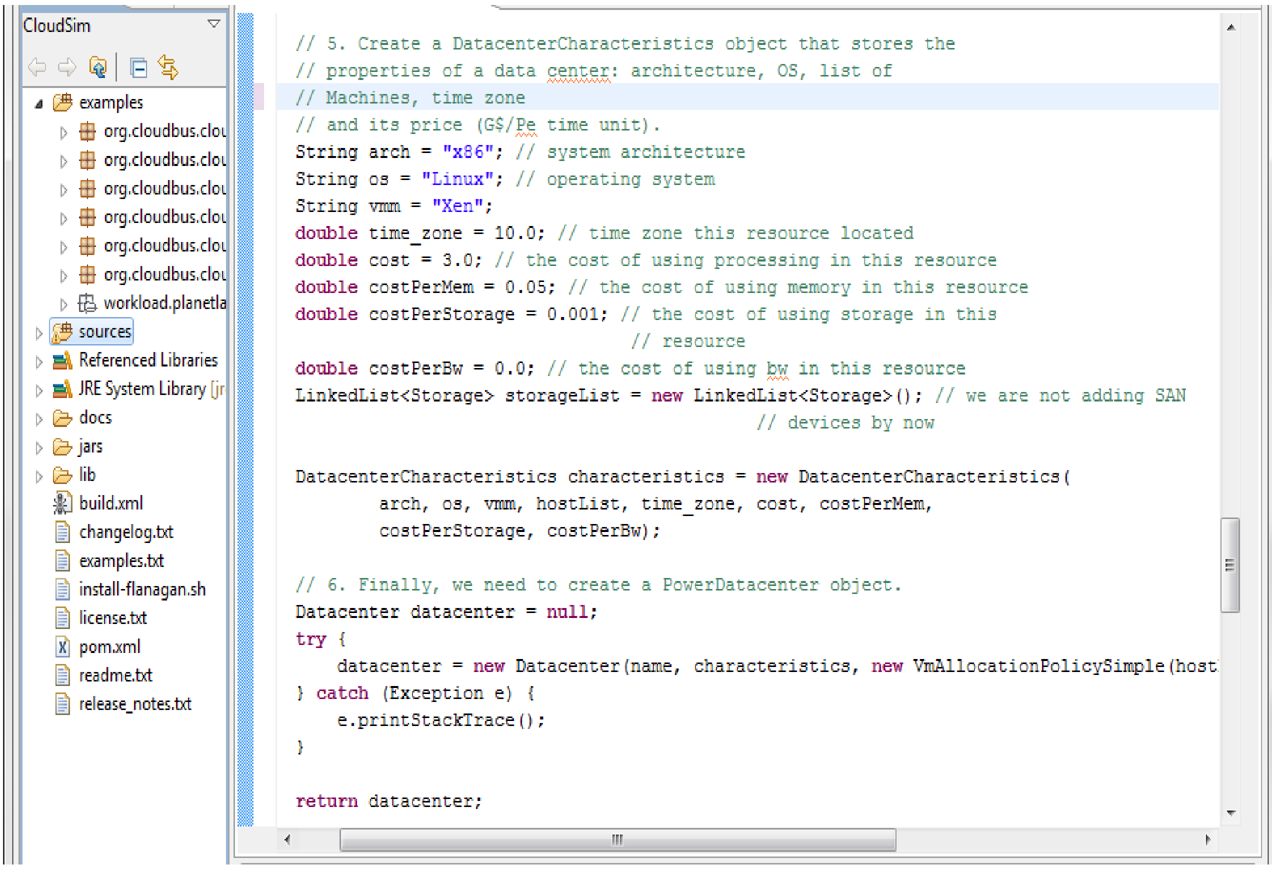


Figure 5.3: Host data center architectural details

Next step is to create Virtual Machines (VMs) to reduce the amount of hardware in use and to define user's file properties. Figure 5.4 shows the process of creation of virtual machine and its architectural details along with user's file properties. After host data center and VMs are created for provisioning of users' requests, the simulated Cloud environment created by CloudSim engine is verified. In Figure 5.5, the simulated Cloud environment is given a trial run for verification and the console window of CloudSim showing successful running of the simulated Cloud is shown in Figure 5.6.

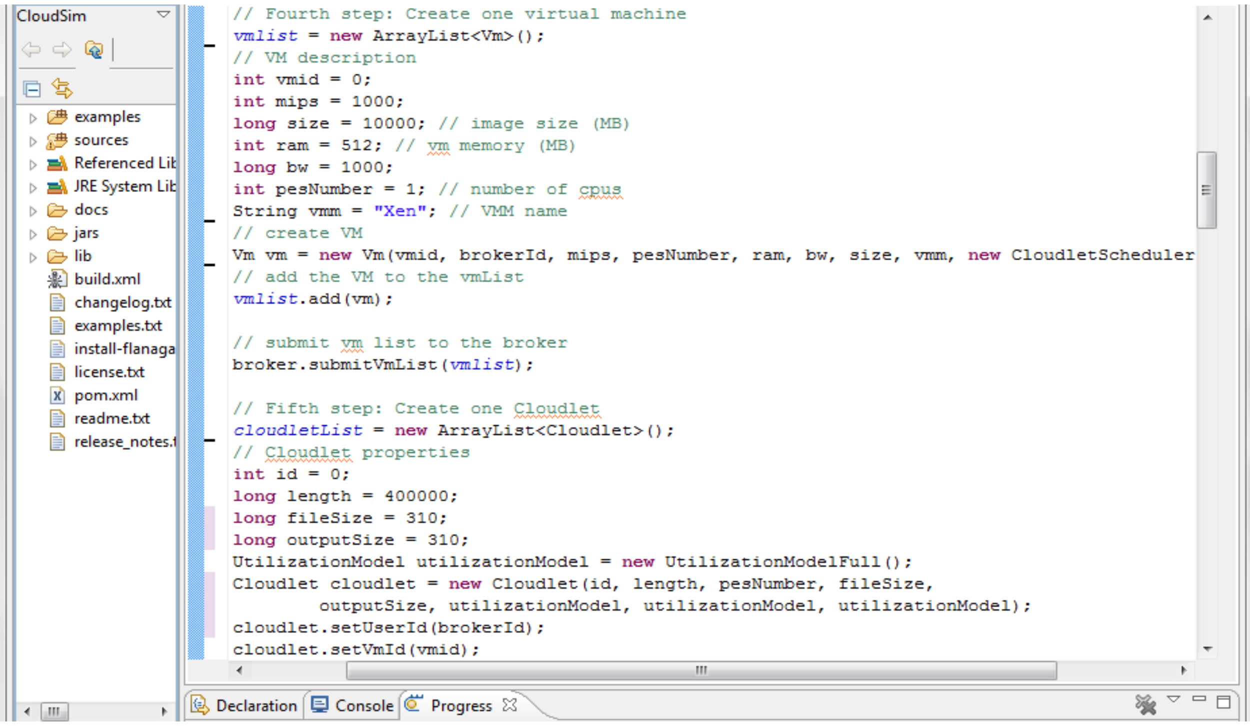


Figure 5.4: Creation of VMs, its architectural details and user file properties

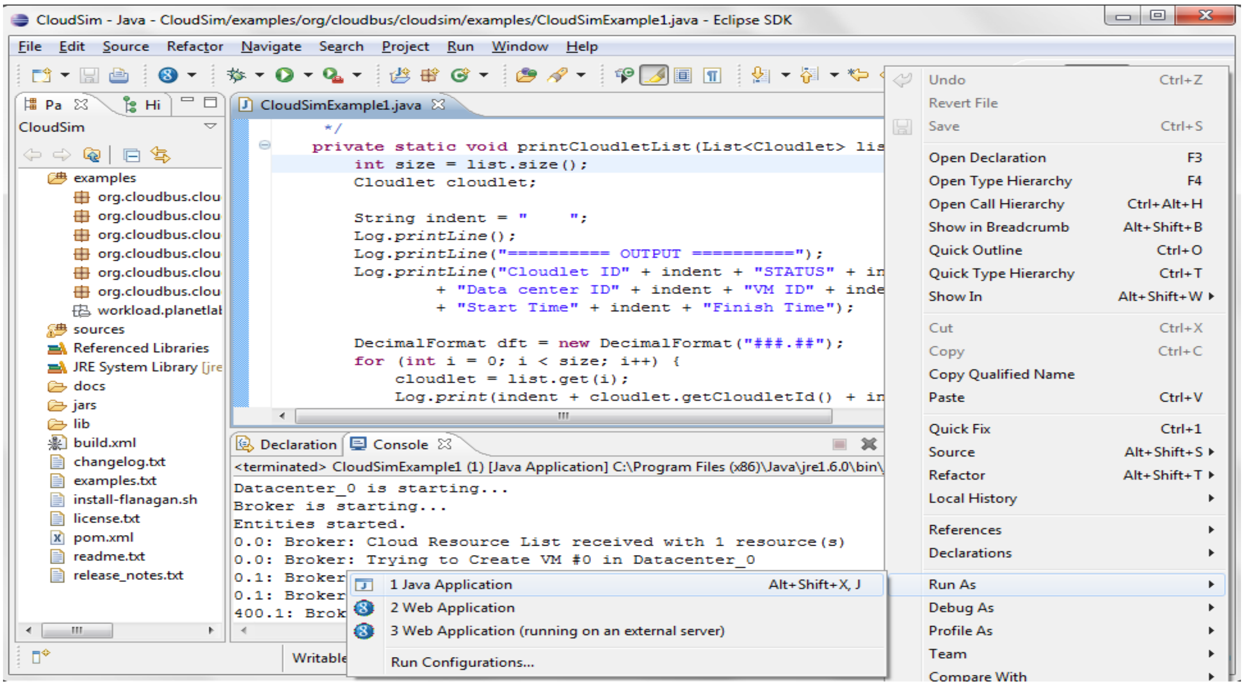


Figure 5.5: Execution of the Simulated Cloud

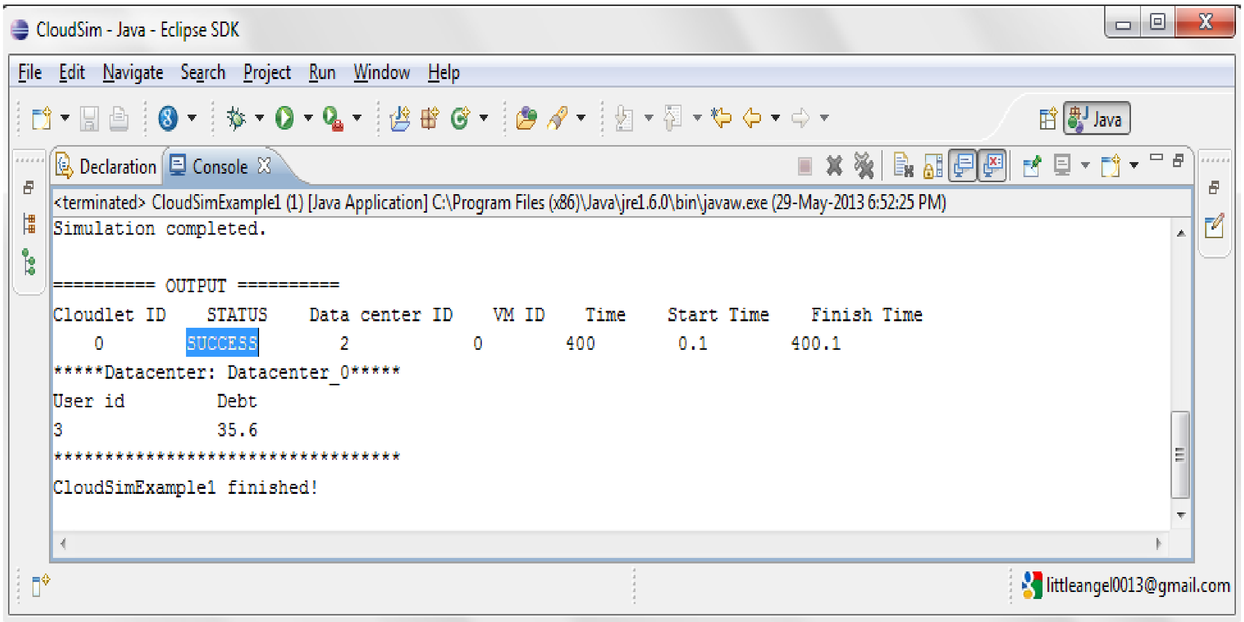


Figure 5.6: Console window showing successful execution of the Cloud

(ii) Implementation at core layer i.e. CloudSim layer

CloudSim layer manages the instantiation and execution of core entities (VMs, hosts, data centers, application) which are created at User Code layer during the simulation period. Efficacy of proposed VM scheduling policy can be studied by programmatically extending the core VM scheduling functionality at core CloudSim layer. Therefore, power-aware VM scheduling technique for Cloud is implemented at core layer.

In Figure 5.7, core VM scheduling policy is shown. As described earlier that, VM scheduling is divided into two parts: the first part is admission of new requests for VM allocation and placement on hosts, and the second part is optimization of current allocation of VMs. In core VM scheduling policy, the first part is implemented by extending a base class named, *PowerVmAllocationPolicyAbstract* class, and for implementing the second part, a public function is created within the Main class as shown in Figure 5.7.

In core scheduling scheme, optimization of current VM allocation of VMs is not carried out. Therefore, public function created for optimization of VM allocation in this scheme is returning null and makes the VM scheduling policy less energy efficient. Figure 5.8 shows the first part of core VM scheduling policy i.e. *PowerVmAllocationPolicyAbstract* class and Figure 5.9 shows the second part.

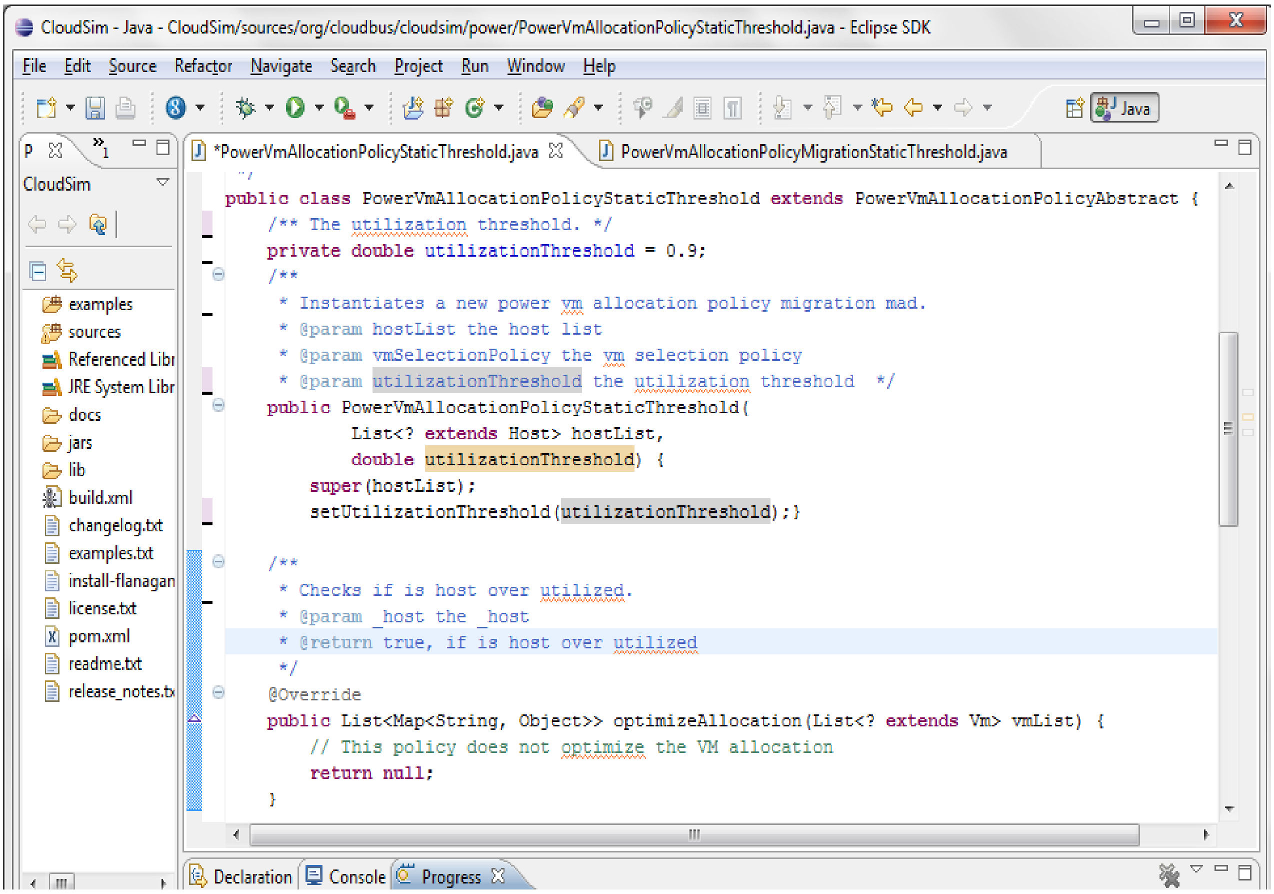


Figure 5.7: Core VM scheduling policy

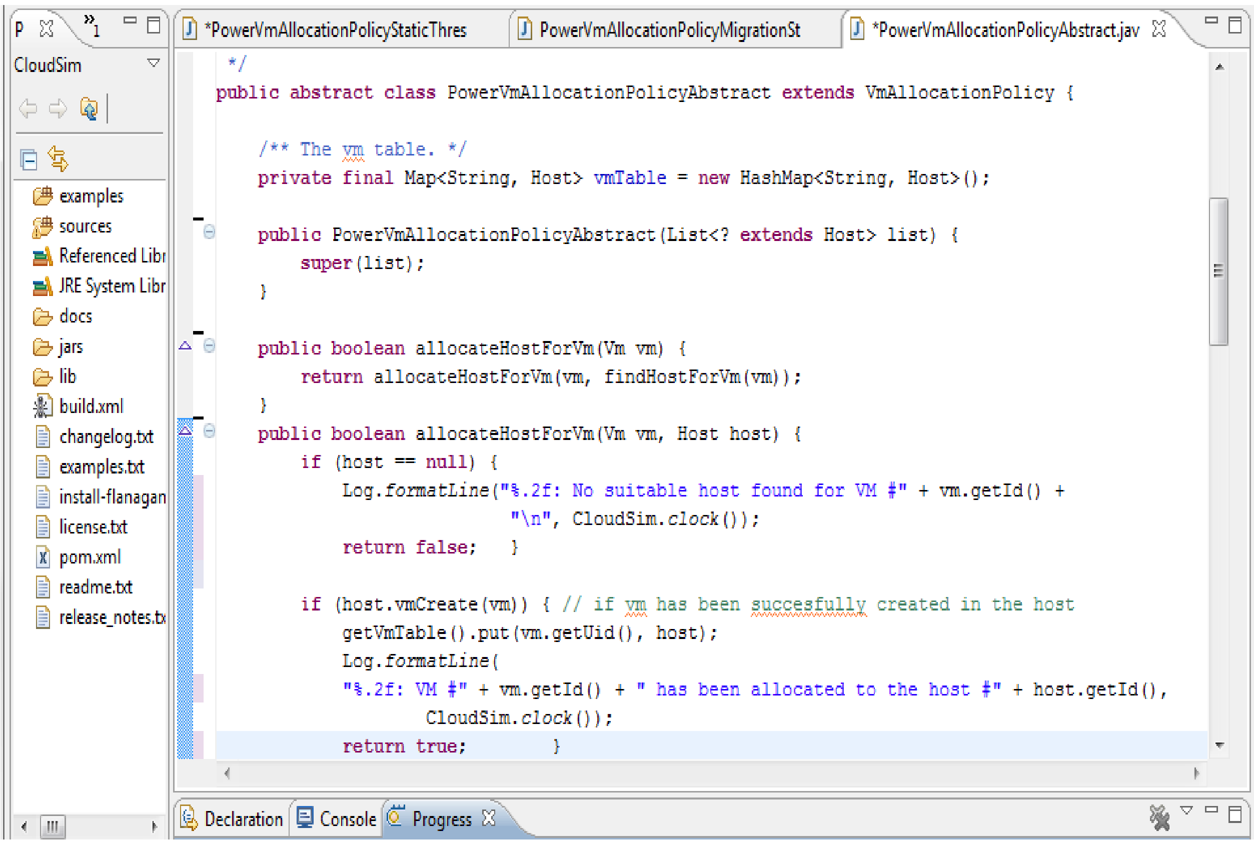


Figure 5.8: VM allocation class of core scheduling policy

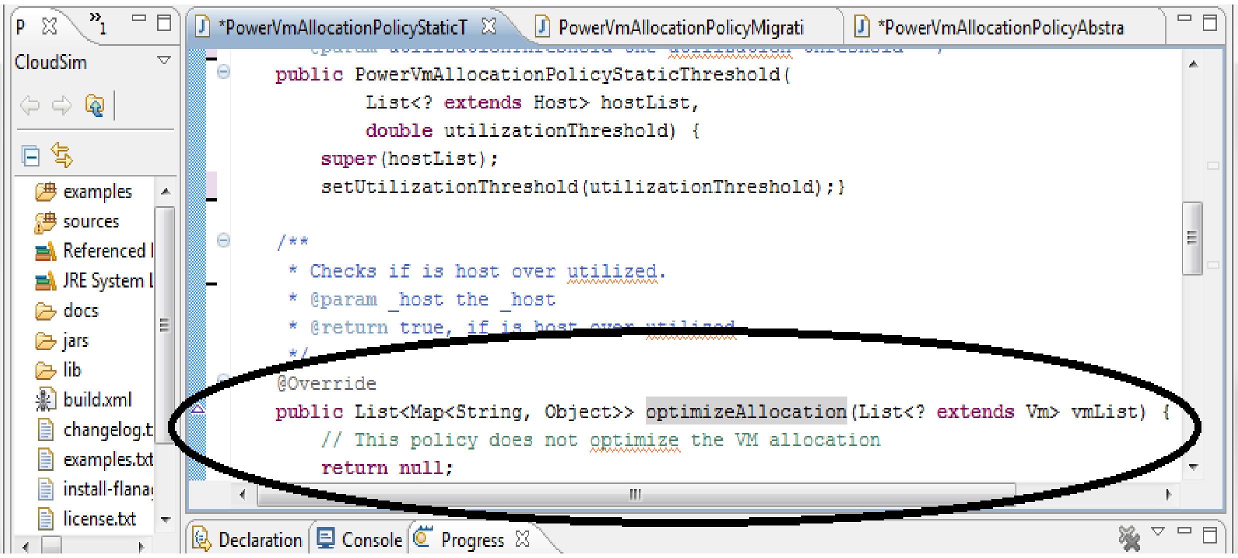


Figure 5.9: Optimization function of Core scheduling policy returning null

To improve the energy efficiency, proposed power-aware VM scheduling approach is implemented by making following changes in the core VM scheduling approach:

(i) By updating the Main class with proposed power-aware scheduling method.

(ii) By extending migration supportive base class i.e. *PowerVmAllocationMigrationAbstract* class instead of non-migration supportive base class i.e. *PowerVmAllocationAbstract* class.

(iii)By overriding Optimization allocation function with proposed optimization method.

Figure 5.10 shows the Main class of proposed power-aware VM scheduling policy. The first part of VM scheduling i.e. admission of new requests for VM scheduling and placement of VMs on hosts is partially implemented in the Main class and is partially implemented in *PowerVmAllocationMigrationAbstract* class.

In *PowerVmAllocationMigrationAbstract* class, methods for VM allocation and optimization of current VM allocation are implemented as per proposed power-aware VM scheduling mechanism. Functionality of this class is implemented by extending it in the Main class of the new VM scheduling policy as shown in Figure 5.10.

Figure 5.11 shows the base class i.e. *PowerVmAllocationMigrationAbstract* class of power-aware VM scheduling technique, which is used for implementing new allocation and optimization mechanism for VMs.

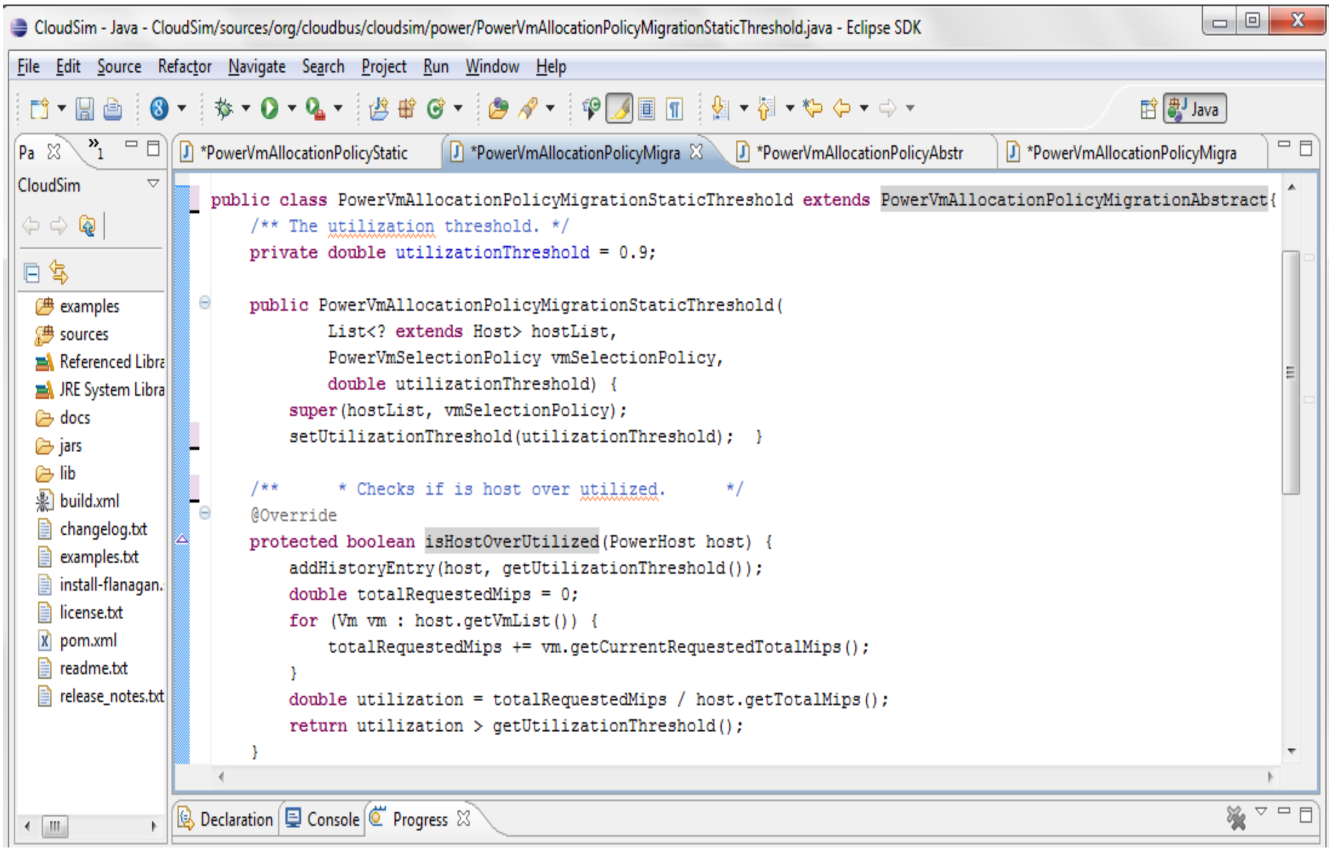


Figure 5.10: Main class of power-aware VM scheduling technique

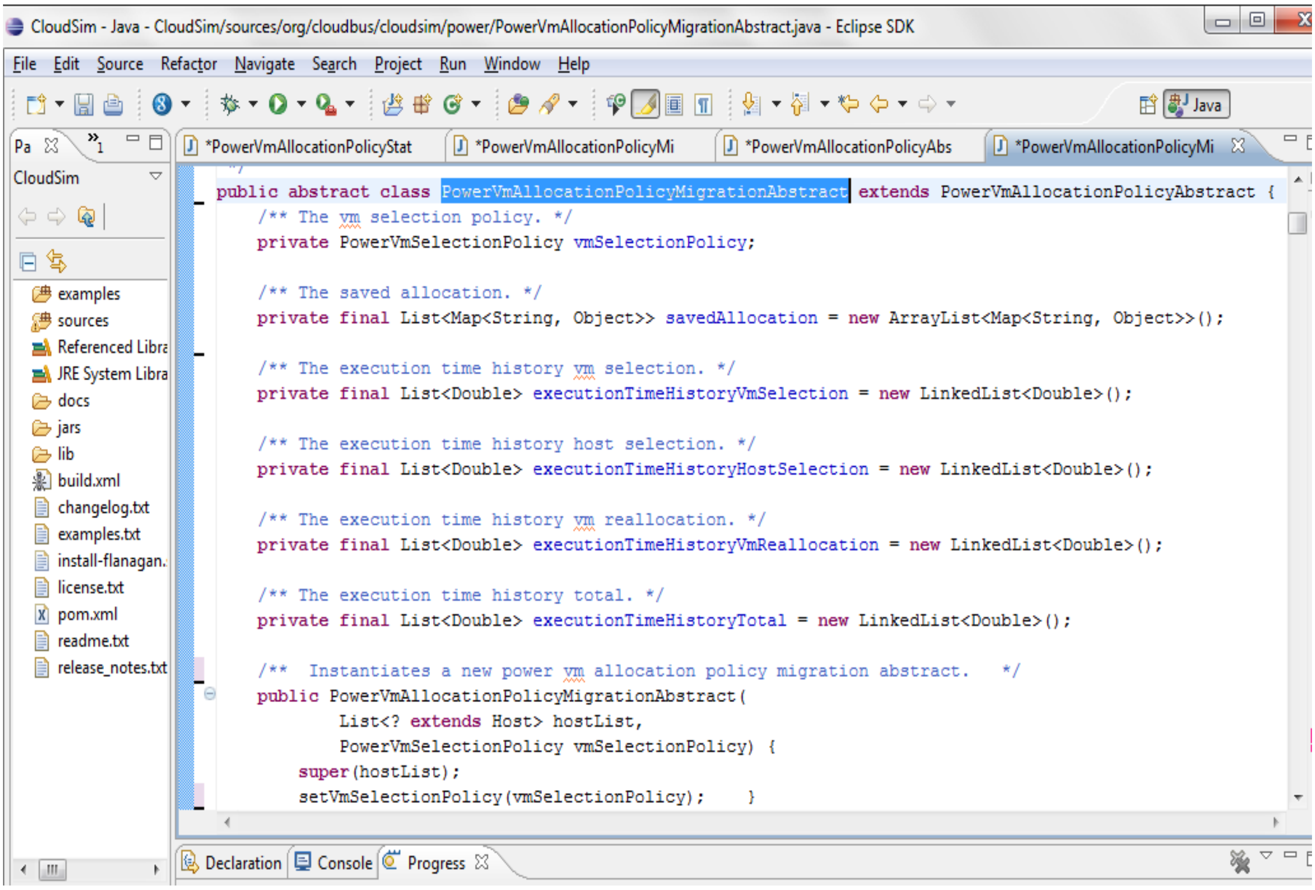


Figure 5.11: Base class for implementing new allocation and optimization mechanism

Figure 5.12 shows the Base class code for implementing power-aware VM allocation mechanism. Then step by step optimization of current VM allocation is done by implementation of proposed optimization mechanism. Figure 5.13 shows optimization function of power-aware VM scheduling policy. This optimization function calls other methods:

- For getting under-utilized hosts.
- For creation of migration map between under-utilized hosts and hosts above migration threshold.
- For new VM placement on the basis of hosts extracted from migration map.
- For saving final VM allocation.
- For switching off idle nodes and returning Power after final allocation.

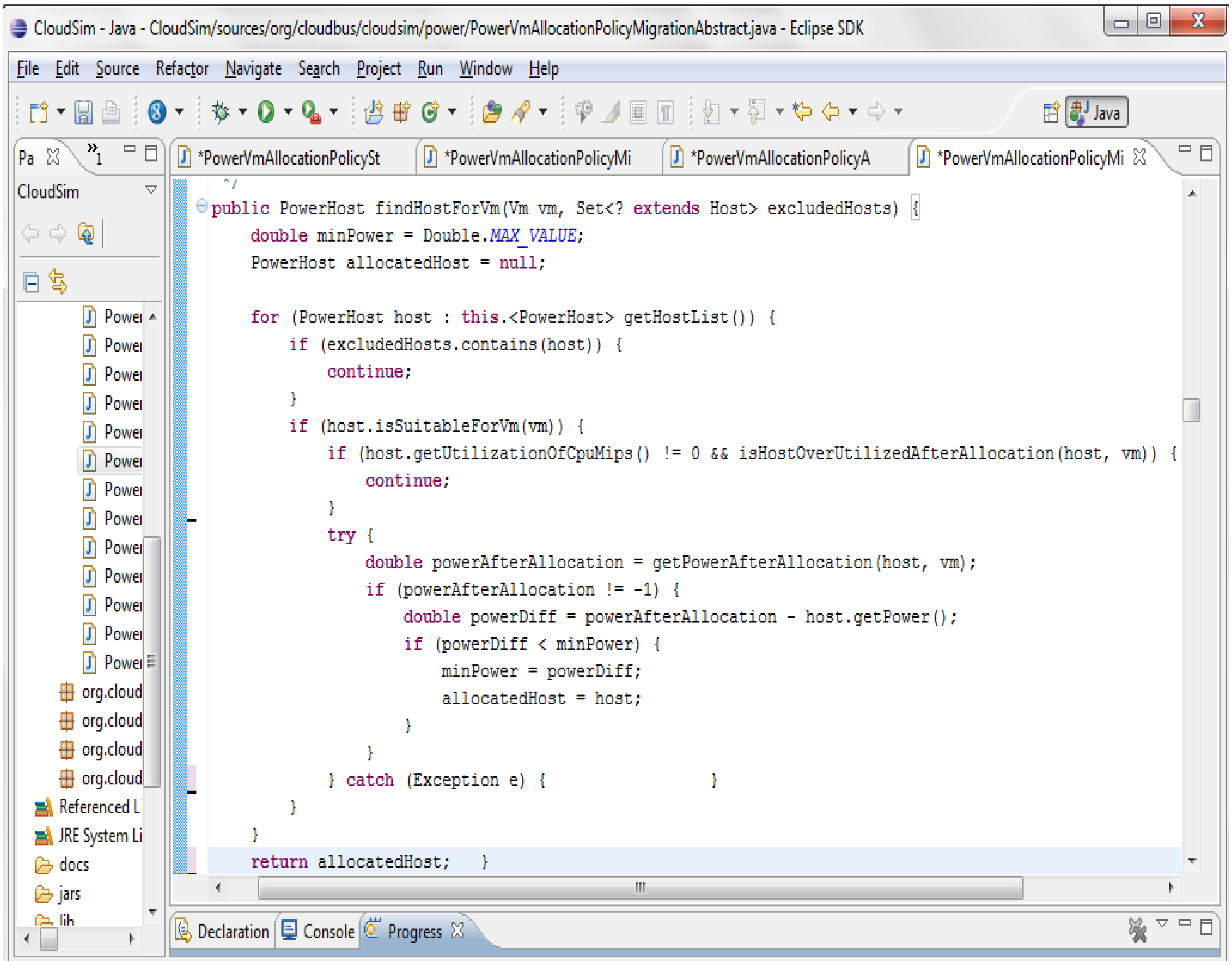


Figure 5.12: Power-aware VM allocation method implemented in Base class

Figure 5.13 shows the optimization function for optimizing current VM allocation, and Figures 5.14, 5.15, 5.16, 5.17, 5.18 and 5.19 show other methods which are called by optimization function for the said purpose.

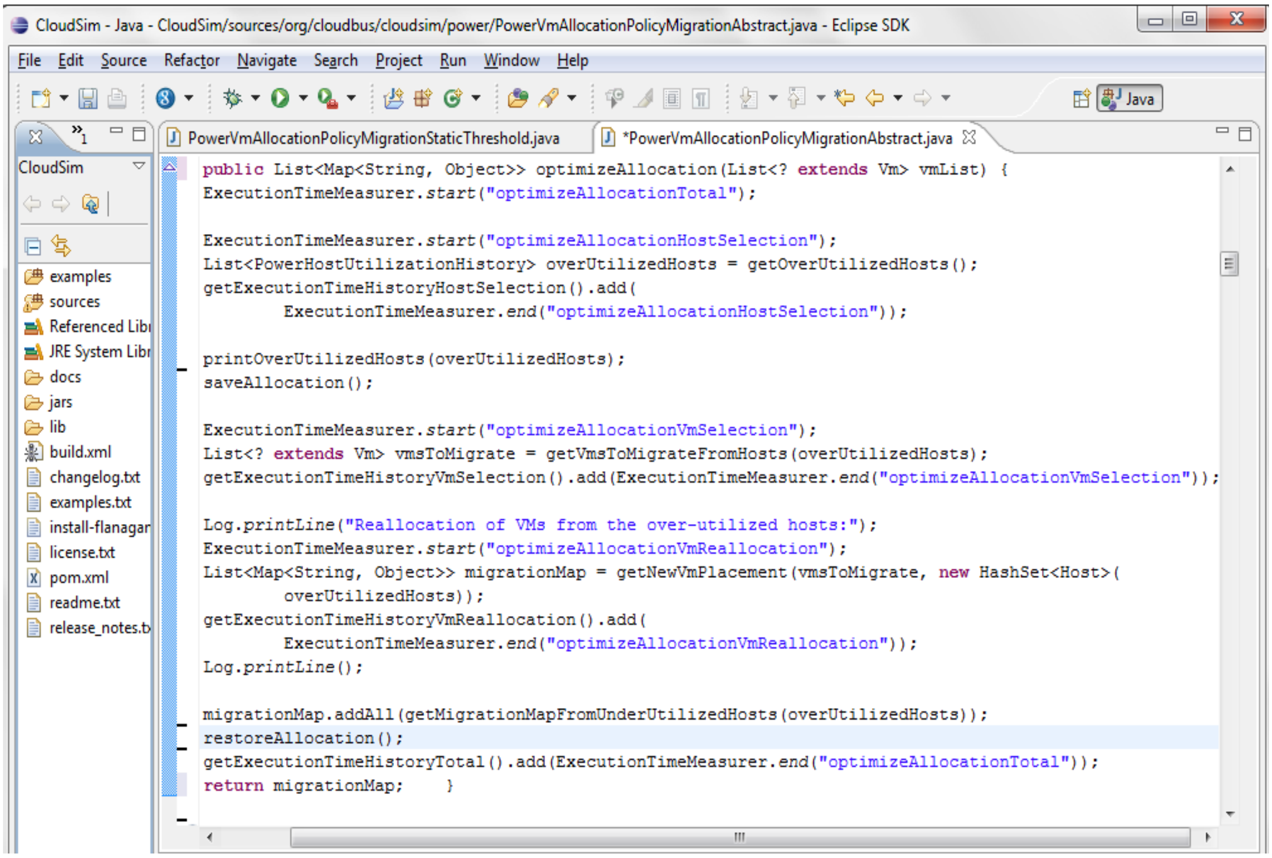


Figure 5.13: Optimization function for current VM allocation

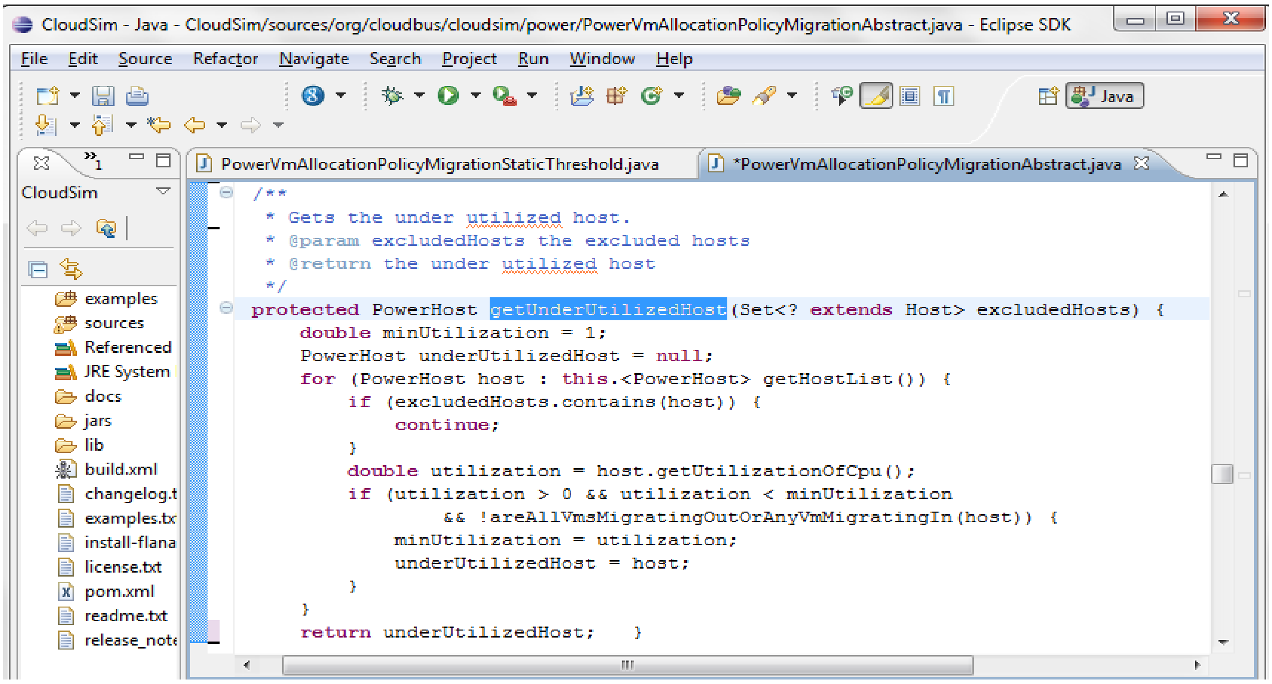


Figure 5.14: Method for getting under-utilized hosts

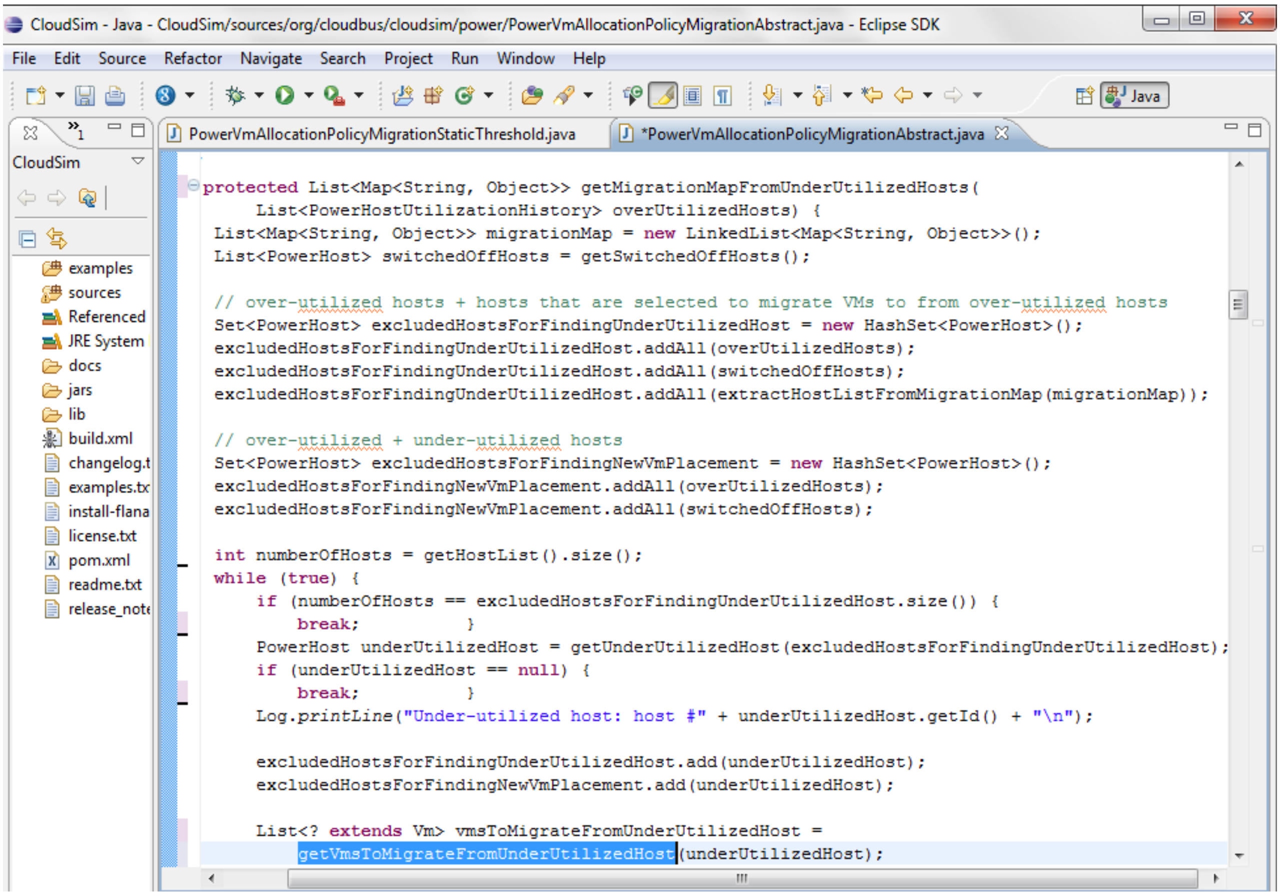


Figure 5.15: Method for creating migration map of underutilized hosts

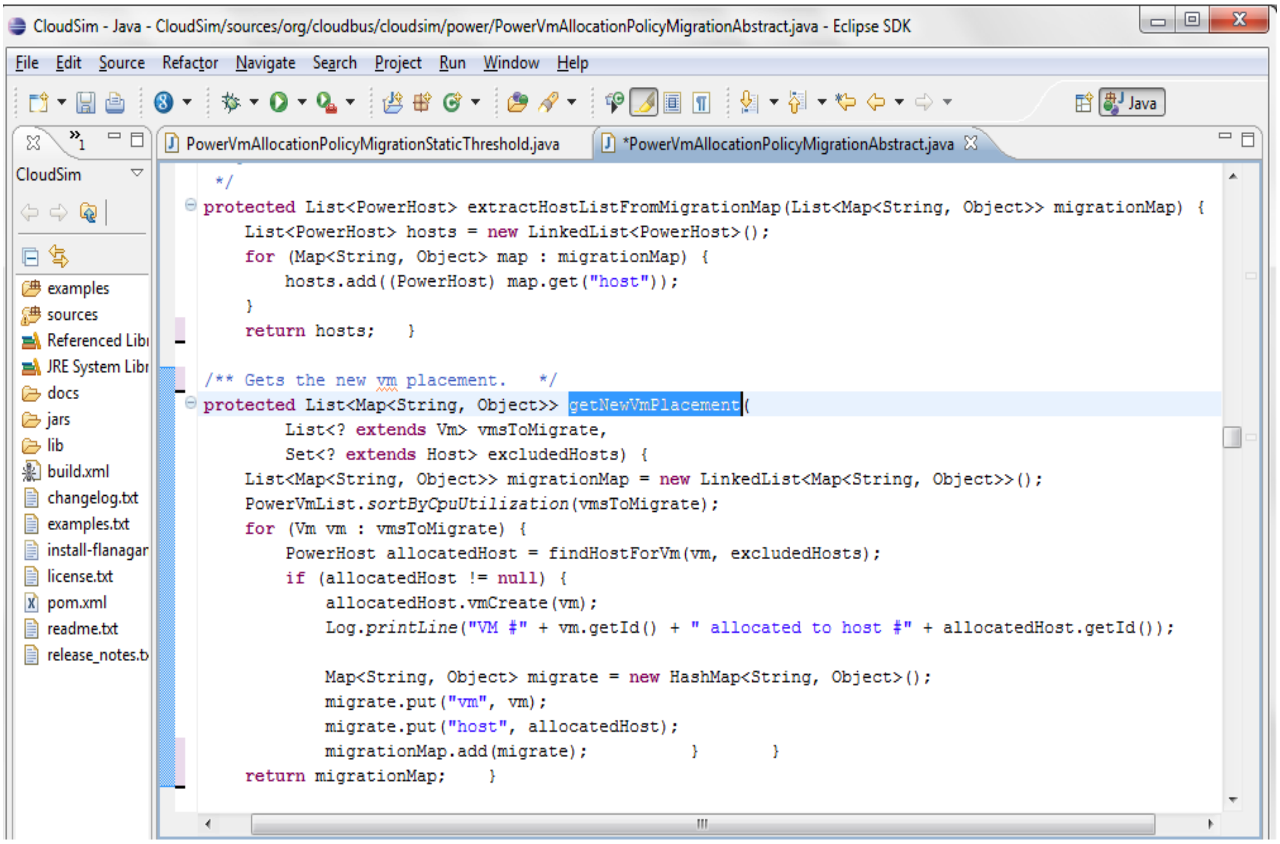


Figure 5.16: Method for getting new VM placement on the basis of hostlist extracted from MigrationMap

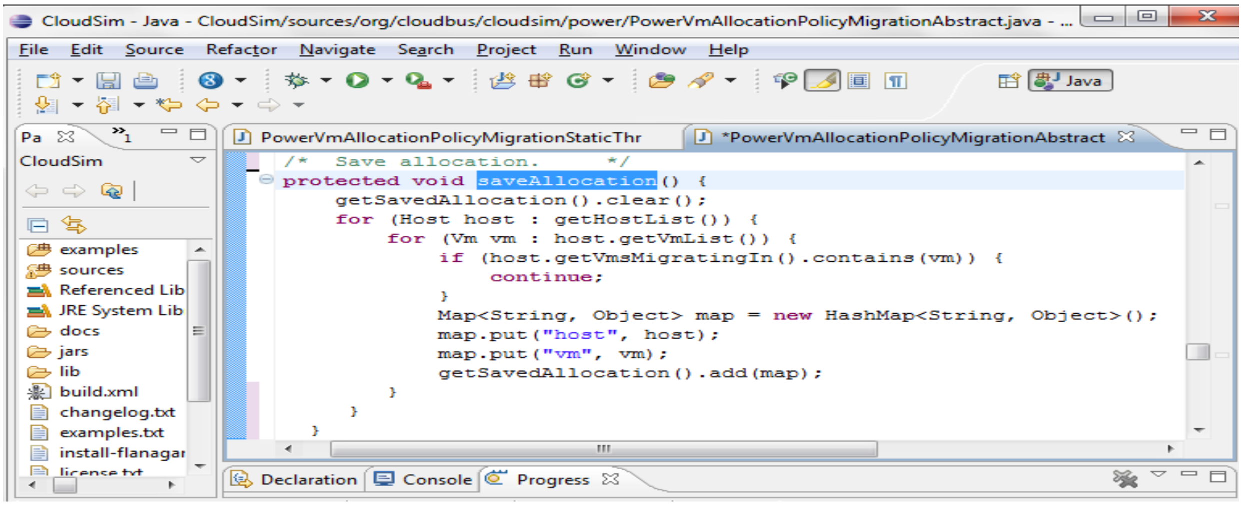


Figure 5.17: Method for saving final VM allocation

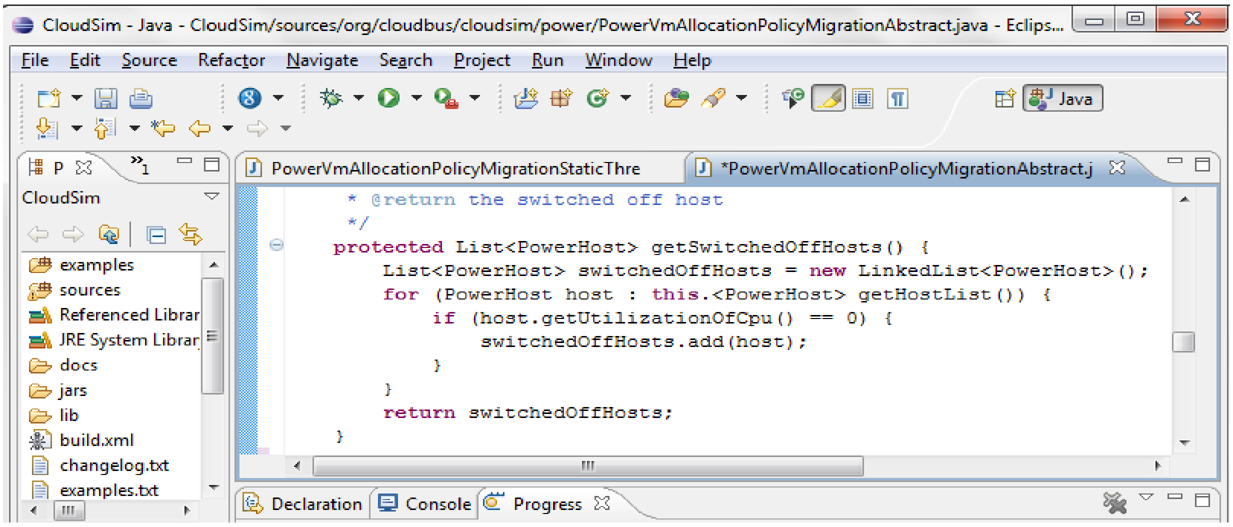


Figure 5.18: Method for getting switched off hosts

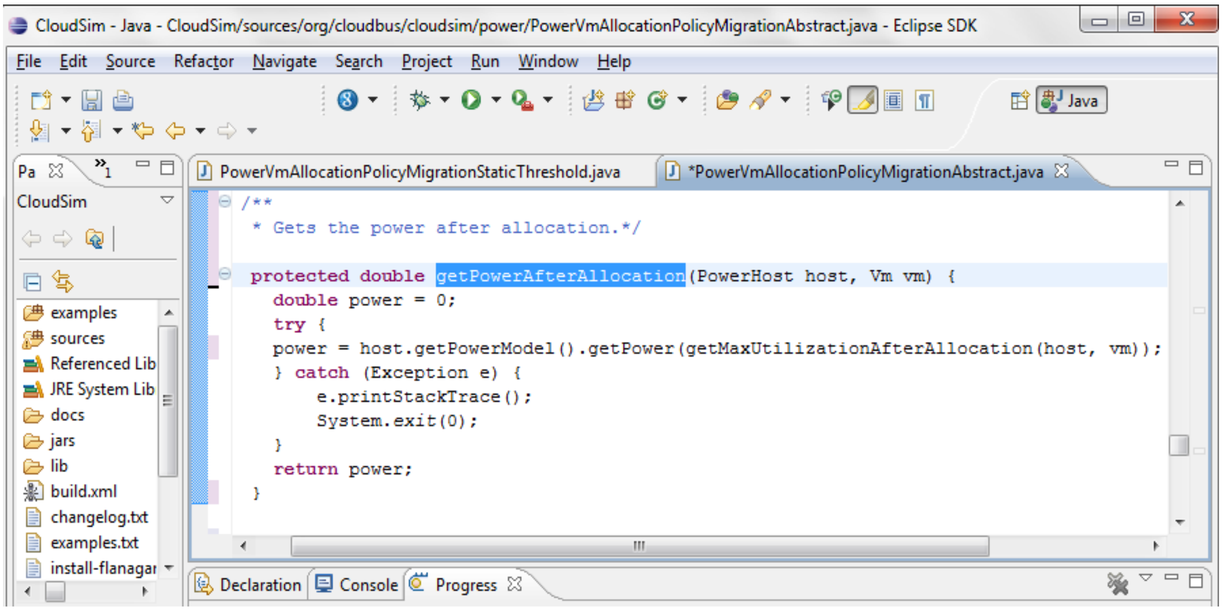


Figure 5.19: Method for calculating power consumption after final VM allocation

5.3 Evaluation Results

The simulated data center comprising of 200 physical nodes is taken into consideration as shown in Figure 5.20. For the benchmark policies, we simulated DVFS policy that adjusts the frequency and voltage of CPU according to current utilization and a Non Power Aware policy (NPA) along with Power-aware scheduling polices i.e. Static Threshold policies for Power-Aware VM scheduling in virtualized heterogeneous systems.

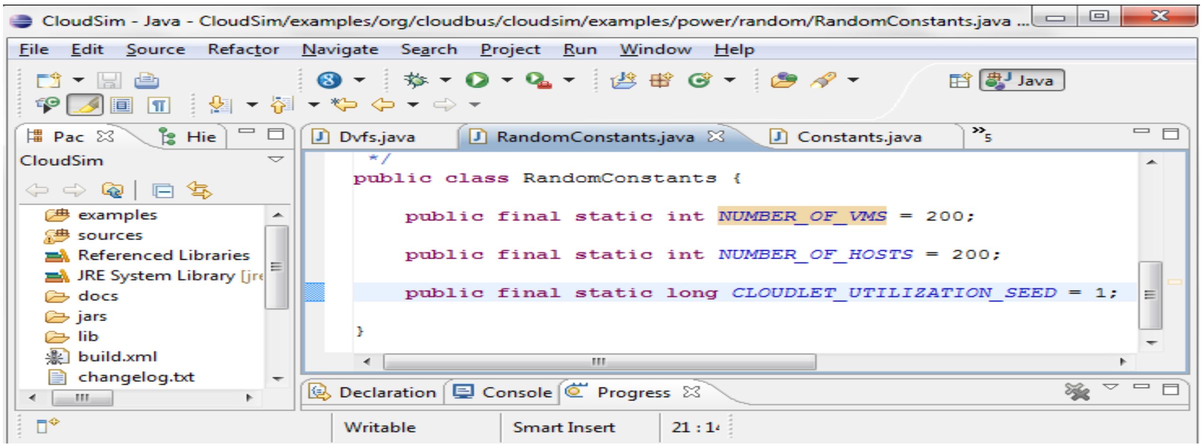


Figure 5.20: Setting number of physical nodes

Figures 5.21 and 5.22 show simulation results and power consumed by NPA (Non Power Aware) policy for given workload.

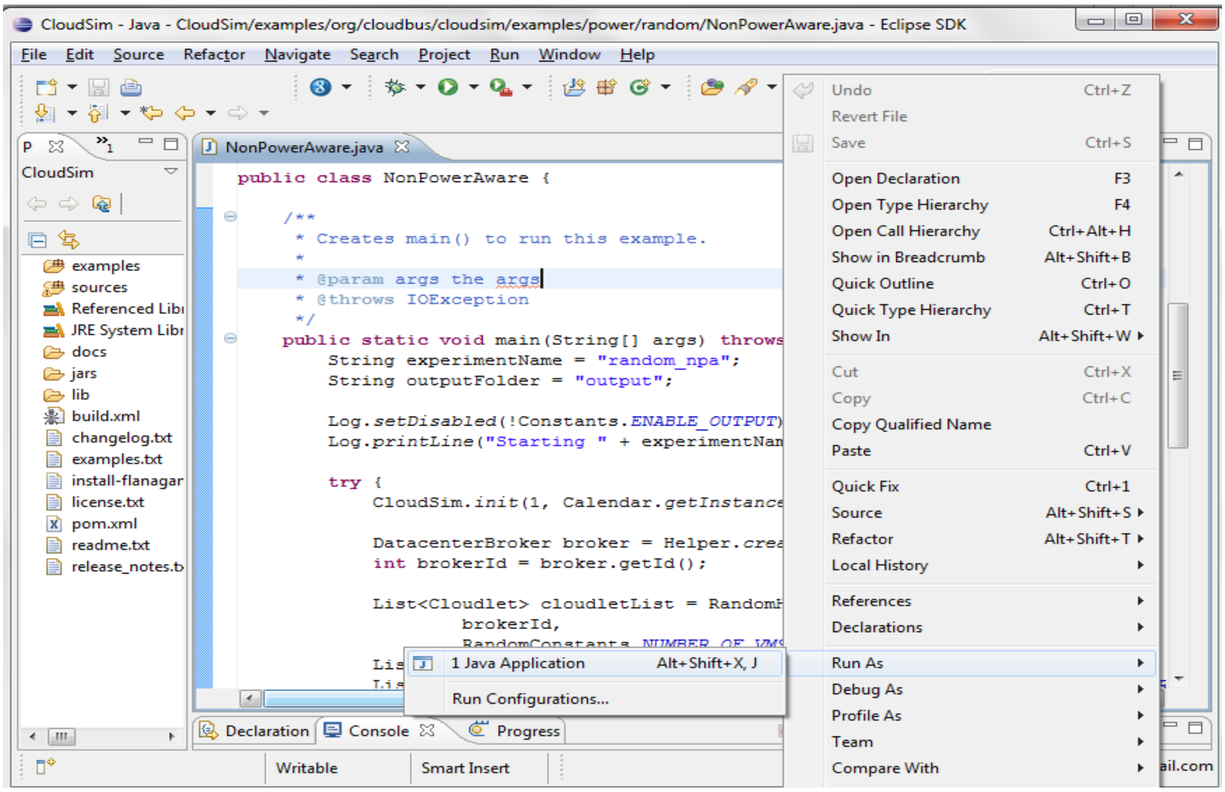


Figure 5.21: Running NPA policy

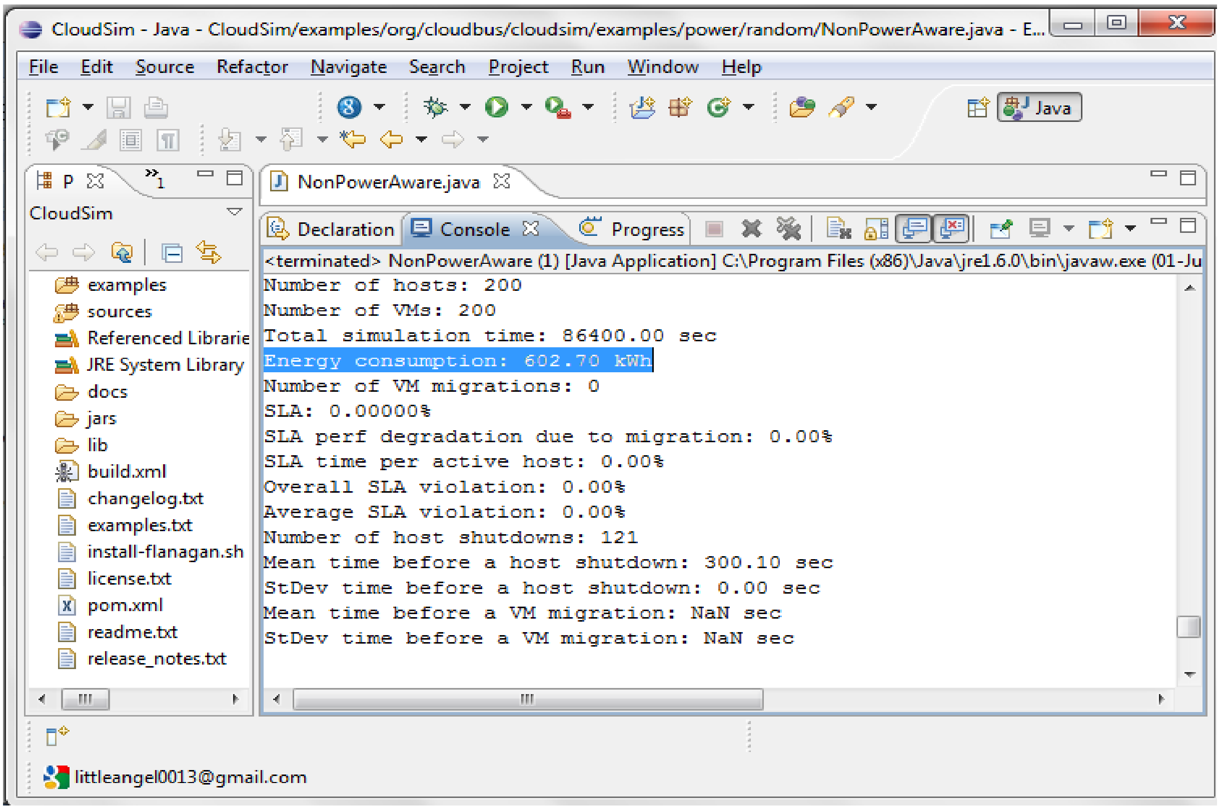


Figure 5.22: Simulation Results of NPA policy

Figures 5.23 and 5.24 show simulation results and power consumed by DVFS policy for given workload.

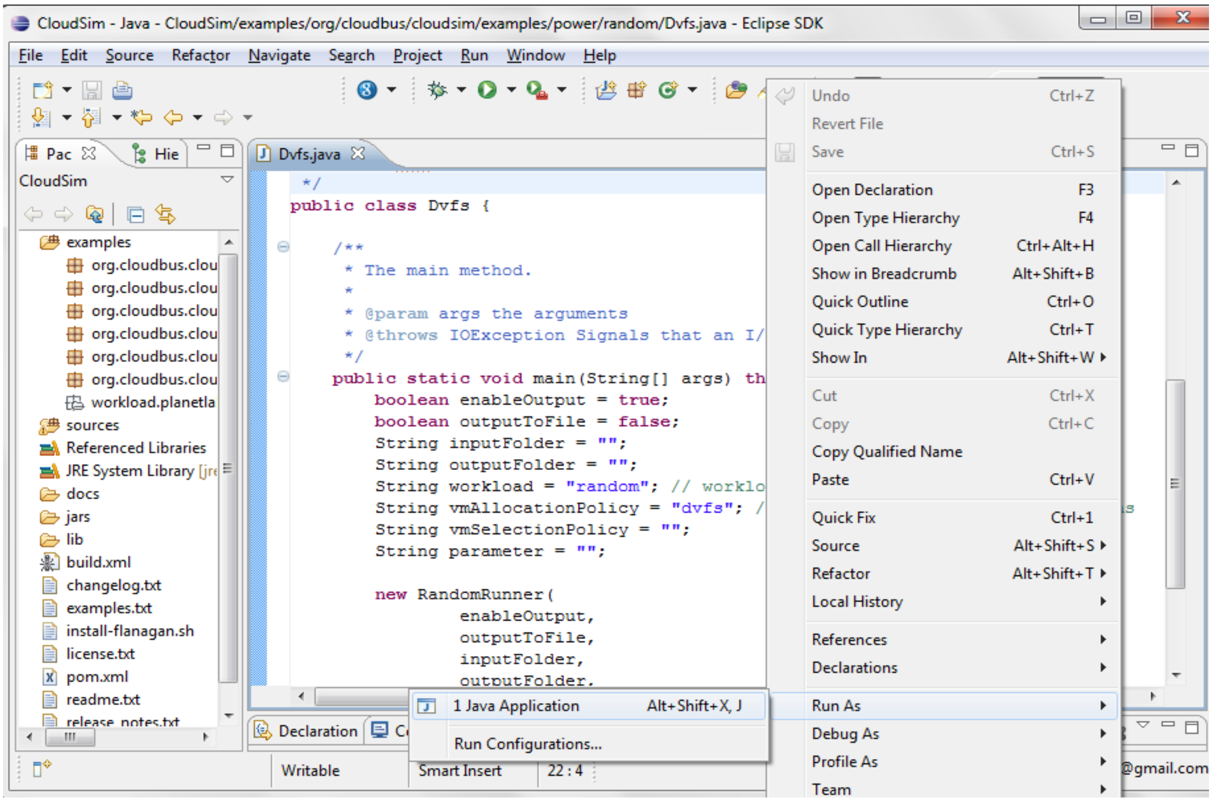


Figure 5.23: Running DVFS policy

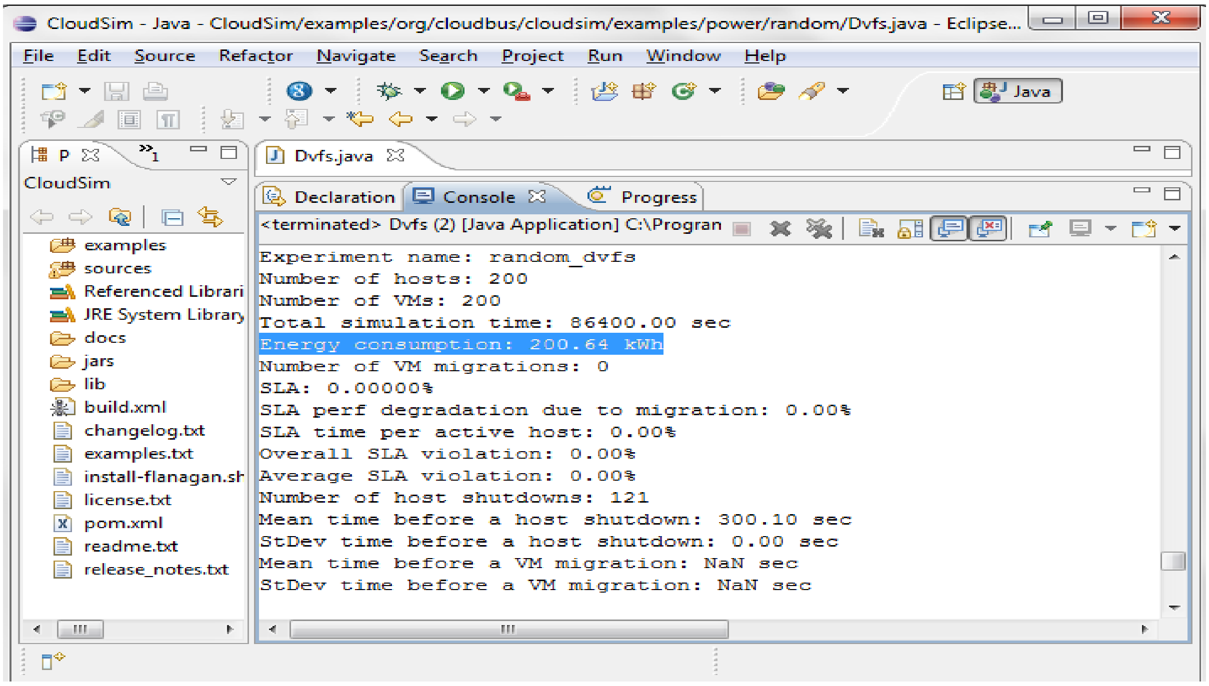


Figure 5.24: Simulation Results of DVFS policy

Figures 5.25 and 5.26 show simulation results and power consumed by Single/Static Threshold (ST) policy for given workload and 90% CPU utilization threshold for hosts.

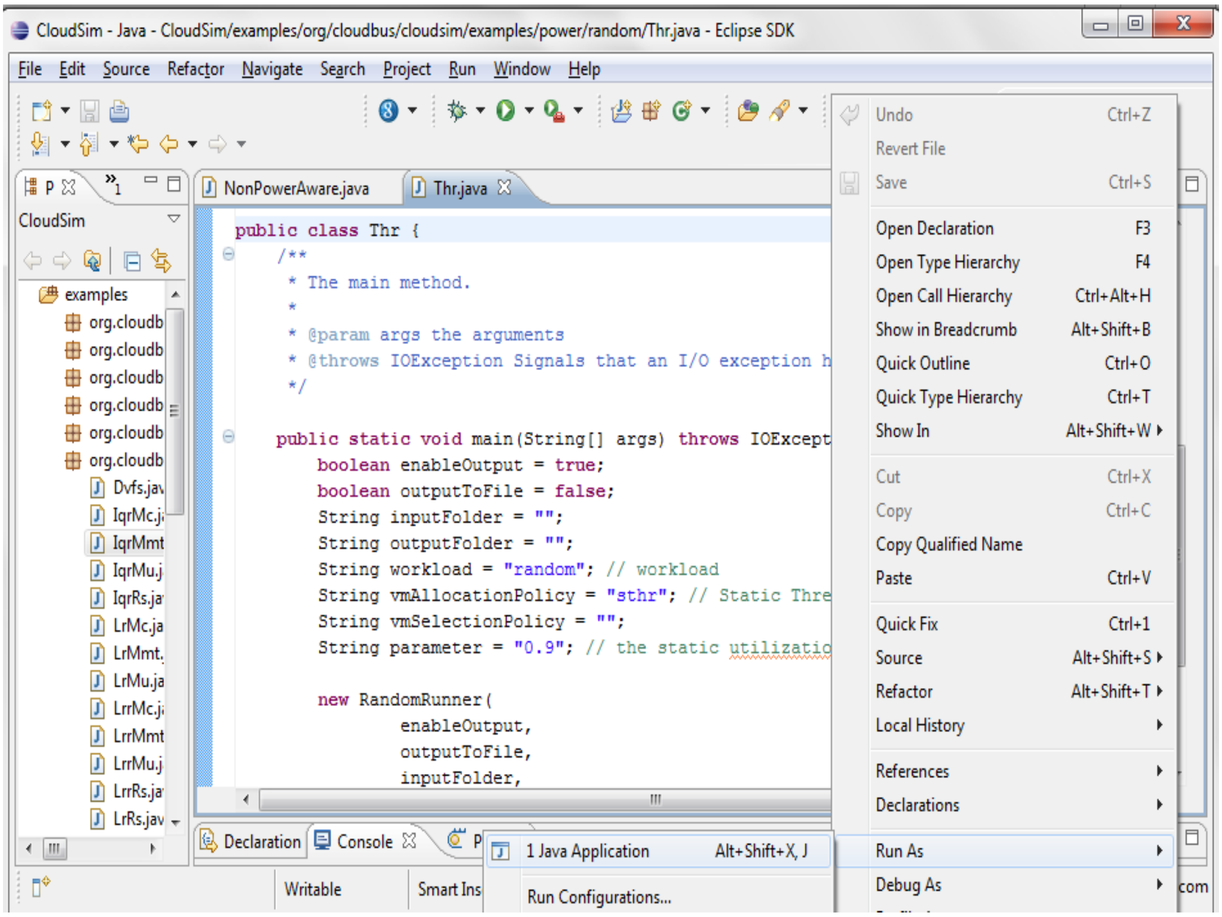


Figure 5.25: Running Single/Static Threshold (ST) Policy

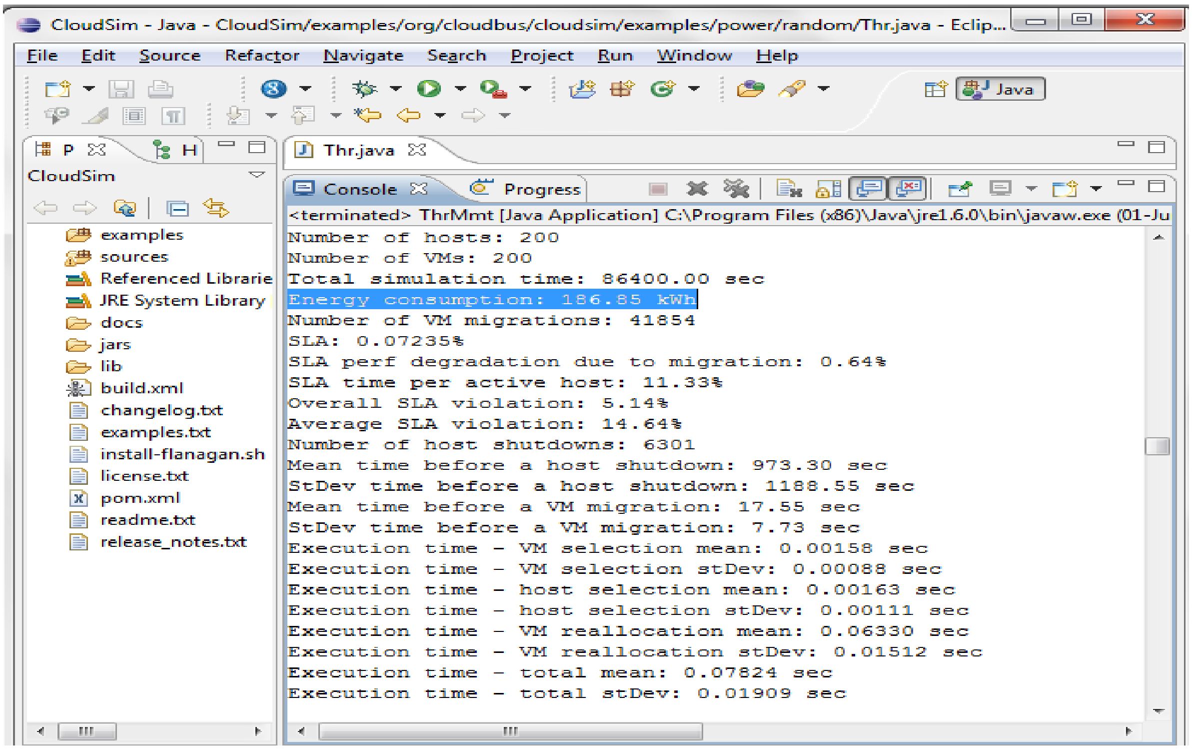


Figure 5.26: Simulation Results of Single/Static Threshold (ST) Policy

Figures 5.27 and 5.28 show simulation results and power consumed by optimized Static Threshold Policy i.e. Static Threshold Minimum Migration Time (ST-MMT) policy for given workload and 90%-20% CPU utilization threshold for hosts.

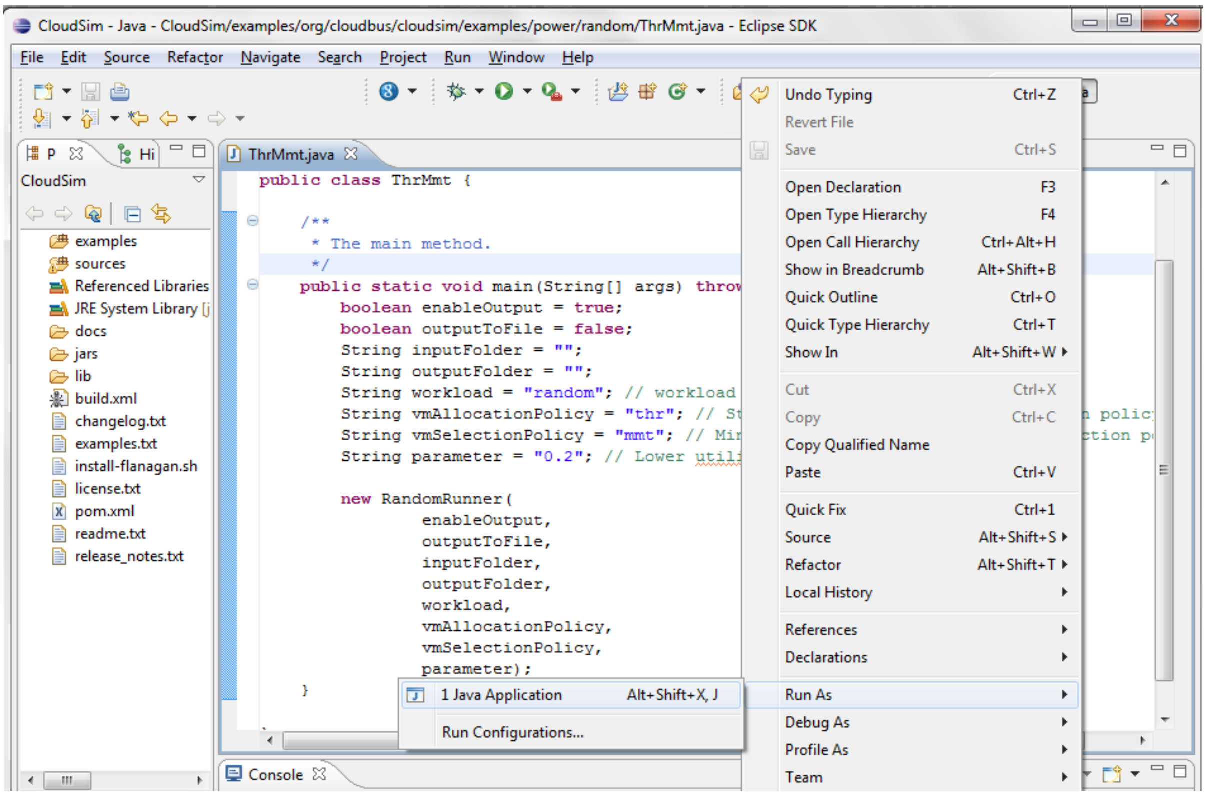


Figure 5.27: Running optimized Static Threshold (ST-MMT) Policy

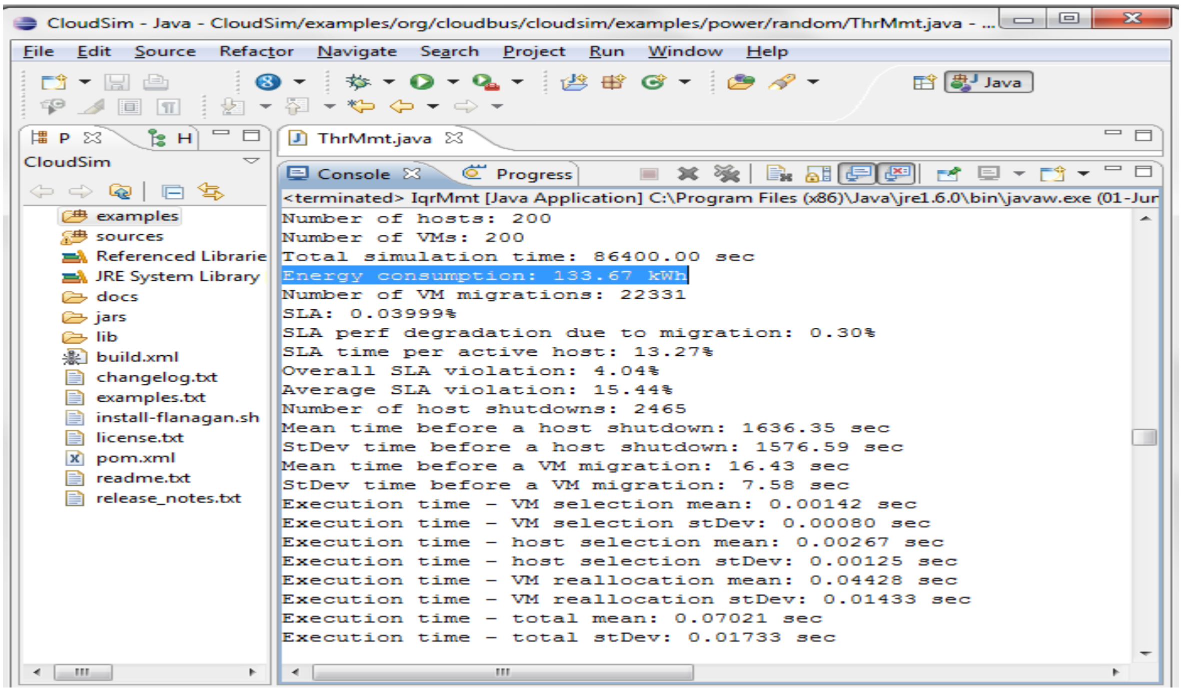


Figure 5.28: Simulation Results of optimized Static Threshold (ST-MMT) Policy

Table 5.1 presents evaluation results of the benchmark policies i.e. DVFS policy and Non Power-Aware (NPA) policy, and Power-Aware scheduling policies i.e. Static Threshold (ST) policy and optimized Static Threshold (ST-MMT) policy. Policies have been evaluated at different threshold values.

Table 5.1: Evaluation Results of VM Scheduling Policies

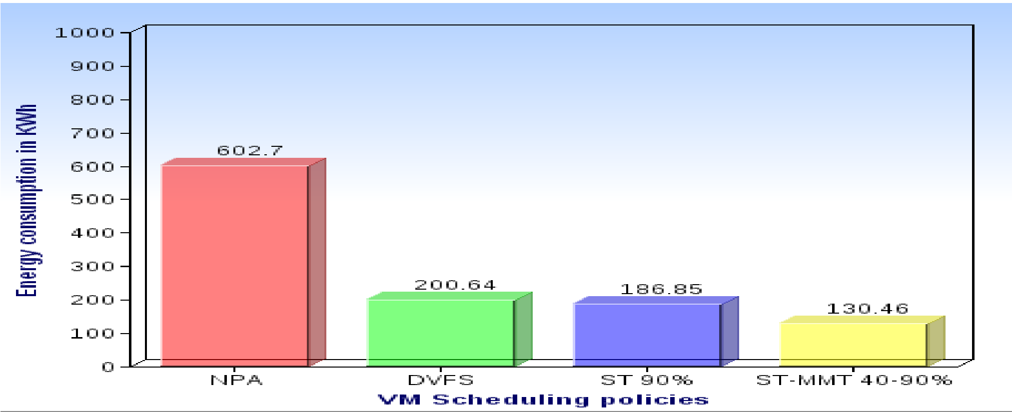
Category	**Policy**	**Energy**	**SLA Violation**	**Total VM Migration**	**Avg. SLA Violation**
Benchmark Scheduling Policies	NPA	602.70 KWh	-	-	-
	DVFS	200.64 KWh	-	-	-
Power-Aware Scheduling Policies	ST 80%	187.78 KWh	3.41%	44376	11.05%
	ST 90%	186.85 KWh	5.14%	41854	14.64%
	ST-MMT 20%-90%	133.65 KWh	4.04%	22331	15.44%
	ST-MMT 30%-90%	133.05 KWh	2.72%	20971	14.56%
	ST-MMT 40%-90%	130.46 KWh	2.29%	18656	14.12%
	ST-MMT 50%-90%	129.96 KWh	3.01%	17885	14.92%

Comparing evaluation results of all the policies shown in Table 5.1, it can be concluded that:

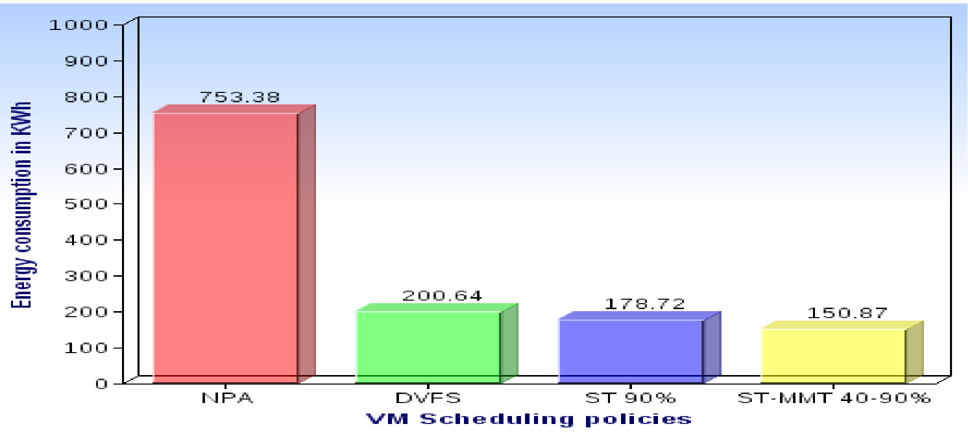
- Though Power-aware scheduling policies result in SLA violation but the ratio of SLA violation is very small in compare to energy savings brought by them. Power-Aware scheduling policies bring significant energy savings in compare to benchmark policies i.e. NPA and DVFS.
- Moreover, Optimized Power-aware scheduling policy not only brings further energy savings but also reduces level of SLA violation. Thus, it can handle strict QoS requirements.

Further, reliability of the power-aware scheduling policy is examined by changing the ratio of number of physical nodes to number of virtual machines. Five different cases have been evaluated as shown below, for checking level of energy consumption by the data center when VM scheduling policies are used at varying number of physical and virtual machines. Evaluation results are presented in the form of barcharts in figure 5.29.

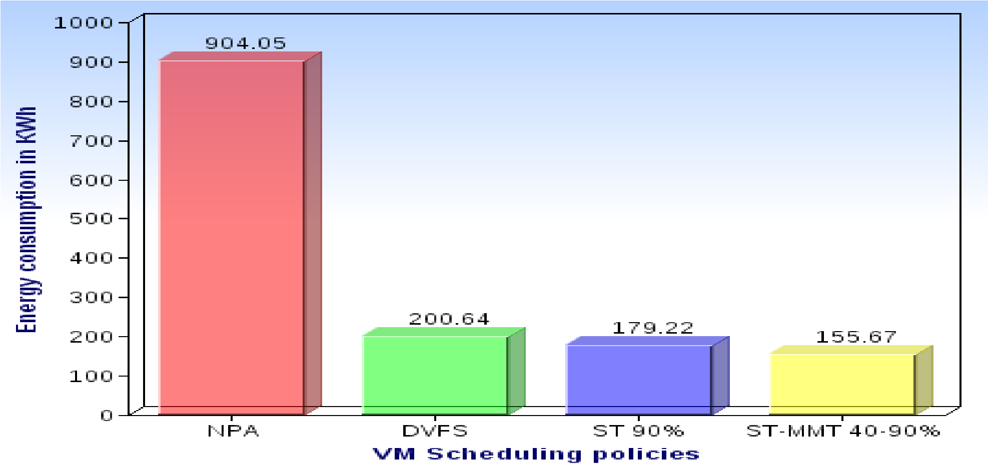
Case 1: When Number of Hosts: 200 and Number of VMs: 200



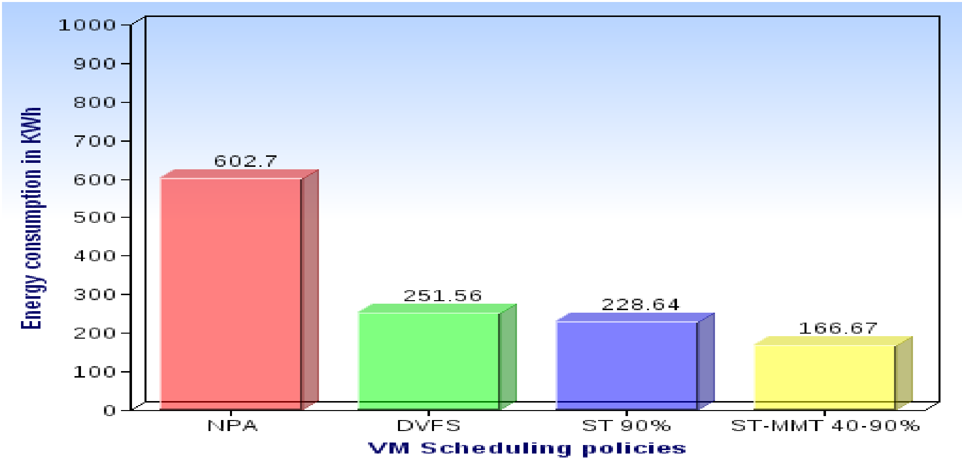
Case 2: When Number of Hosts: 250 and Number of VMs: 200



Case 3: When Number of Hosts: 300 and Number of VMs: 200



Case 4: When Number of Hosts: 200 and Number of VMs: 250



Case 5: When Number of Hosts: 200 and Number of VMs: 300

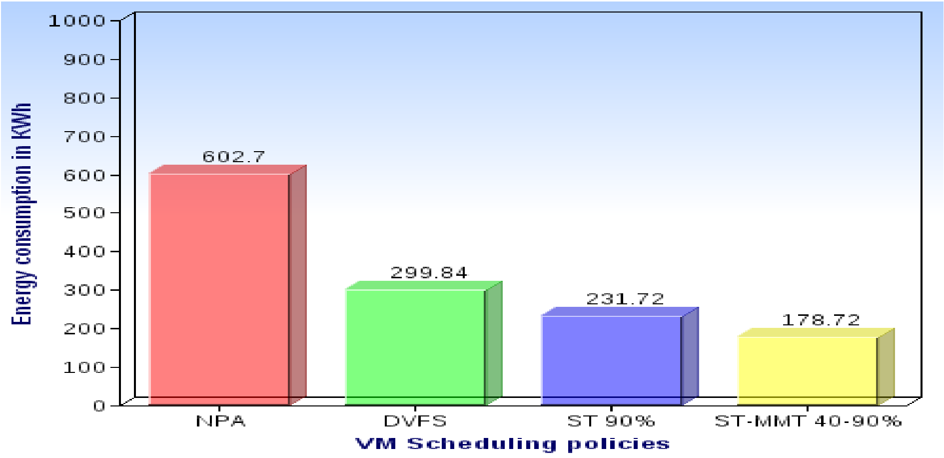


Fig 5.29: Energy consumption level of VM scheduling policies at varying Host-VM rate

From evaluation results of VM scheduling policies at varying rate of host and virtual machines, it can be concluded that:

- NPA Policy gives the highest energy consumption rate in compare to other policies. It results in increase in energy consumption as ratio of Hosts to VMs increases, whereas energy consumption level remains almost equal when ratio of Hosts to VMs decreases.
- DVFS Policy incurs standard level of energy consumption either ratio of Hosts to VMs increases or decreases.
- In contrast to benchmark policies, power-aware scheduling policies i.e. ST and ST-MMT result in minimum energy consumption in all cases. This justifies reliability of power-aware scheduling policies in virtualized heterogeneous environment and thus, validates further development of stated policy.

Chapter 6

Conclusion and Future Scope

This chapter concludes the work presented in this thesis and suggests future research direction which can be taken further.

6.1 Conclusion

This thesis gives introduction of Cloud computing and background of various virtual machine management techniques in Cloud data centers. In this work, gaps in existing virtual machine scheduling techniques have been analyzed and on the basis of gap analysis, a power-aware virtual machine scheduling technique for virtualized heterogeneous systems has been proposed. The proposed technique has been implemented in simulated cloud environment with the help of CloudSim toolkit. This technique has been compared with benchmark policies and experimental results have been gathered.

6.2 Thesis Contribution

a) In this thesis, existing virtual machine management techniques have been analyzed and compared according to their features.

b) This thesis work presents and evaluates an energy-aware resource allocation approach making use of dynamic consolidation of VMs.

c) Experimental results have revealed that this approach results in considerable reduction of energy consumption in Cloud data center relatively to former techniques.

d) This work precedes the Cloud computing field in two ways:

- First, it has a vital role in the reduction of energy consumption costs in Cloud data centers, and thus helps in developing a strong and competitive Cloud computing industry.
- Second, reduction in energy consumption leads to reduction in green house gas emissions and hence assists in achieving green computing.

6.3 Future Scope

a) This work optimizes virtual machine allocation in Cloud data centers by constrained consolidation of VMs according to CPU utilization threshold of hosts and switching off idle nodes to minimize power consumption. For future work, it is projected to investigate setting of utilization thresholds dynamically according to current allocation of VMs to a host and checking its influence on multi-core architectures. Also, decentralization of optimization algorithms can be considered for improving scalability and fault tolerance of Cloud data centers.

b) Further interesting directions for the future work are investigation of effects of network interface and disk storage on reallocation decisions of virtual machines, as both significantly contribute to the overall energy consumption.

References

[1] D. C. Wyld, "Moving to the Cloud: An Introduction to Cloud Computing in Government," IBM center for The Business of Government e- Government Series, 2009.

[2] B. Alexander, "Web 2.0: A New Wave of Innovation for Teaching and Learning?" Learning, vol. 41, no. 2, pp. 32-44, 2006.

[3] R. Buyya, C. Yeo, and S. Venugopal, "Market-oriented cloud computing: Vision, hype, and reality for delivering it services as computing utilities," in proceedings of the 10th IEEE International Conference on High Performance Computing and Communications (HPCC-08, IEEE CS Press, Los Alamitos, CA, USA), 2008.

[4] S. Zhang, S. Zhang, X. Chen and X. Huo, "The Comparison Between Cloud Computing and Grid Computing", International Conference on Computer Application and System Modeling, 2010.

[5] P. Mell, and T. Grance, "The NIST Definition of Cloud computing," National Institute of Standards and Technology, 2009.

[6] K. Jeffery, B. Neidecker-Lutz, "The Future of Cloud Computing. Opportunities for European Cloud Computing Beyond 2010", Expert Group Report, European Commission, January 2010.

[7] V. R. Srinivasa, N. K. R. Nageswara and E. K. Kumari, "Cloud Computing: An Overview," Journal of Theoretical and Applied Information Technology, 2009.

[8] S. Garg and R. Buyya, "Green Cloud Computing and Environmental Sustainability", Harnessing Green IT: Principles and Practices, S. Murugesan and G. Gangadharan (eds), Wiley Press, UK, in press, accepted on April 2, 2011.

[9] T. Dillon, C. Wu, and E. Chang, "Cloud Computing: Issues and Challenges", in 24th IEEE International Conference on Advanced Information Networking and Applications, pp. 27-33, 2010.

[10] V. K. Reddy, B. T. Rao, L. S. S. Reddy and P.S. Kiran, "Research issues in Cloud Computing," Global Journal Computer Science & Technology Vol. 11, Issue no. 11, pp. 70-76, June 2011.

[11] J. Hamilton, "Cooperative expendable micro-slice servers (CEMS): low cost, low power servers for Internet-scale services," In the Proceeding of CIDR, 2009.

[12] W. Forrest, "How to cut data centre carbon emissions?" December 2008. [Online]. Available:

http://www.computerweekly.com/Articles/2008/12/05/233748/how-to-cut-data-centre-carbon-emissions.htm

[13] Aasys, "Virtualization Basics," Vol. 6, Issue 9, September 2008.

[14] L. Lefevre and A. C. Orgerie, "Designing and evaluating an energy efficient Cloud," The Journal of Supercomputing, Springer, Vol. 51, Issue no. 3, pp. 352-373, March 2010.

[15] "Xen User Manual," http:// bits.xensource.com/Xen/docs/user.pdf

[16] R. Buyya, A. Beloglazov and J. Abawajy, "Energy-efficient management of data center resources for cloud computing: a vision, architectural elements, and open challenges," In: International conference on parallel and distributed processing techniques and applications (PDPTA), Las Vegas, USA, 2010.

[17] C. Hsu and W. Feng, "A power-aware run-time system for high performance computing," in Proceedings of 2005 ACM/IEEE conference on Supercomputing. IEEE Computer Society Washington, DC, USA, 2005.

[18] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. F. D. Rose, and R. Buyya, "CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," Software: Practice and Experience, Wiley Press, New York, USA, 2010.

[19] "Eclipse," Available at: http://en.wikipedia.org/wiki/Eclipse_(software).

[20] I. Foster, Y. Zhao, I. Raicu, and S. Lu, "Cloud Computing and Grid Computing 360-Degree Compared," in Grid Computing Environments Workshop, 200B, GCE'OB, pp. 1-10, 2008.

[21] J. Koomey, "Estimating total power consumption by servers in the US and the world," Final report, vol. 15, February 2007.

[22] A. Berl, E. Gelenbe, M. D. Girolamo, G. Giuliani, H. D. Meer, M. Q. Dang and K. Pentikousis, "Energy-Efficient Cloud Computing," The Computer Journal, Vol. 53, Issue no. 7, pp. 1045-1051, 2009.

[23] N. Kim, J. Cho and E. Seo, "Energy-credit scheduler: An energy-aware virtual machine scheduler for cloud systems," Elsevier Journal of Future Generation Computer Systems (2012), doi:10.1016/j.future.2012.05.019.

[24] R. Bertran and Y. Becerra, et al., "Energy accounting for shared virtualized environments under DVFS using PMC-based power models," Elsevier Journal of Future Generation Computer Systems, vol. 28, pp. 457–468, March 2011.

[25] A. Berl and H. D. Meer, “An energy consumption model for virtualized office environments,” Elsevier Journal of Future Generation Compute Systems, vol. 27, pp. 1047–1055, April 2011.

[26] X. Liao, H. Jin and H. Liu, “Towards a green cluster through dynamic remapping of virtual machines,” Elsevier Journal of Future Generation Computer Systems, vol. 28, pp. 469–477, May 2011.

[27] T. C. Ferreto, et al., “Server consolidation with migration control for virtualized data centers,” Elsevier Journal of Future Generation Computer Systems, vol. 27, pp. 1027–1034, May 2011.

[28] Z. Xuea, et al., “A performance and energy optimization mechanism for cooperation-oriented multiple server clusters,” Elsevier Journal of Future Generation Computer Systems, vol. 28, pp. 801–810, May 2011.

[29] B. Dougherty, J. White and D. C. Schmidt, “Model-driven auto-scaling of green cloud computing infrastructure,” Elsevier Journal of Future Generation Computer Systems, vol. 28, pp. 371–378, May 2011.

[30] R. Jeyarani, N. Nagaveni and R. V. Ram, “Design and implementation of adaptive power-aware virtual machine provisioner (APA-VMP) using swarm intelligence,” Elsevier Journal of Future Generation Computer Systems, vol. 28, pp. 811–821, July 2011.

[31] I. Goiri, et al., “Energy-efficient and multifaceted resource management for profit-driven virtualized data centers,” Elsevier Journal of Future Generation Computer Systems, vol. 28, pp. 718–731, January 2012.

[32] A. Corradi, M. Fanelli and L. Foschini, “VM consolidation: A real case based on OpenStack Cloud,” Elsevier Journal of Future Generation Computer Systems (2012), doi: 10. 1016/j.future.2012.05.012.

[33] L. Hea, et al., “Developing resource consolidation frameworks for moldable virtual machines in clouds,” Elsevier Journal of Future Generation Computer Systems (2012), doi:10.1016/j.future.2012.05.015.

[34] A. Kertesz, G. Kecskemeti and I. Brandic, “An interoperable and self-adaptive approach for SLA-based service virtualization in heterogeneous Cloud environments” Elsevier Journal of Future Generation Computer Systems (2012), doi:10.1016/j.future.2012.05.016.

[35] S. Zikos and H. D. Karatza, “Performance and energy aware cluster-level scheduling of compute-intensive jobs with unknown service times,” Elsevier Journal of Simulation Modelling Practice and Theory, vol. 19, pp. 239–250, June 2010.

[36] A. Beloglazov, J. Abawajy and R. Buyya, “Energy-aware resource allocation heuristics for efficient management of data centers for Cloud computing,” Elsevier Journal of Future Generation Computer Systems, vol. 28, pp. 755–768, May 2011.

[37] L. Bertini, J. C.B. Leite and D. Mosse, “Power optimization for dynamic configuration in heterogeneous web server clusters,” Elsevier Journal of Systems and Software, vol. 83, pp. 585–598, Nov 2009.

[38] T. V. Do and C. Rotter, “Comparison of scheduling schemes for on-demand IaaS requests,” Elsevier Journal of Systems and Software, vol. 85, pp. 1400– 1408, Feb. 2012.

[39] D. M. Quan, et al., “T-Alloc: A practical energy efficient resource allocation algorithm for traditional data centers,” Elsevier Journal of Future Generation Computer Systems vol. 28, pp. 791–800, May 2011.

[40] Citrix, “Consolidation without Compromise.” Citrix Whitepaper. Available at www.citrix.com/ [Last Accessed: December 2012].

[41] G. Kousiouris, T. Cucinotta and T. Varvarigou, “The effects of scheduling, workload type and consolidation scenarios on virtual machine performance and their prediction through optimized artificial neural networks,” Elsevier Journal of Systems and Software, vol. 84, pp. 1270– 1291, April 2011.

[42] X. Wang, Z. Du and Y. Chen, “An adaptive model-free resource and power management approach for multi-tier cloud environments,” Elsevier Journal of Systems and Software, vol. 85, pp. 1135– 1146, January 2012.

[43] D. Ardagna, et al., “Dual time-scale distributed capacity allocation and load redirect algorithms for cloud systems,” Elsevier J. Parallel Distrib. Comput., vol. 72, pp. 796–808, March 2012.

[44] X. Zhua, et al., “Adaptive energy-efficient scheduling for real-time tasks on DVS-enabled heterogeneous clusters,” Elsevier J. Parallel Distrib. Comput., vol. 72, pp. 751–763, March 2012.

[45] B. Javadi, J. Abawajy and R. Buyya, “Failure-aware resource provisioning for hybrid Cloud infrastructure,” Elsevier J. Parallel Distrib. Comput., vol. 72, pp. 1318–1331, July 2012.

[46] F. Aisopos, K. Tserpes and T. Varvarigou, “Resource management in software as a service using the knapsack problem model,” Elsevier J. Production Economics (2011), doi:10.1016/j.ijpe.2011.12.011.

[47] X. Kong, et al., “Efficient dynamic task scheduling in virtualized data centers with fuzzy prediction,” Elsevier Journal of Network and Computer Applications, vol. 34, pp. 1068–1077, June 2010.

[48] Dell, Intel, “Virtualization: Consolidation to Cloud Foundation.” Dell-Intel Whitepaper. [Last Accessed: December 2012].

[49] M. Witkowskia, et al., “Practical power consumption estimation for real life HPC applications,” Elsevier Journal of Future Generation Computer Systems, vol. 29, pp. 208–217, June 2012.

[50] S. Benedict, “Energy-aware performance analysis methodologies for HPC architectures— An exploratory study,” Elsevier Journal of Journal of Network and Computer Applications, vol. 35, pp. 1709–1719, Aug. 2012.

[51] T. Wood, et al., “Sandpiper: Black-box and gray-box resource management for virtual machines,” Elsevier Journal of Computer Networks, vol. 53, pp. 2923–2938, July 2009.

[52] W. Fang, et al., “VMPlanner: Optimizing virtual machine placement and traffic flow routing to reduce network power costs in cloud data centers,” Computer Networks (2012), http://dx.doi.org/10.1016/j.comnet.2012.09.008.

[53] S. Chen, et al., “Using CPU gradients for performance-aware energy conservation in multitier systems,” Elsevier Journal of Sustainable Computing: Informatics and Systems, vol. 1, pp. 113–133, Feb. 2011.

[54] H. L. Chan, T. W. Lam and R. Li, “Tradeoff between energy and throughput for online deadline scheduling,” Elsevier Journal of Sustainable Computing: Informatics and Systems, vol. 1, pp. 189–195, May 2011.

[55] V. A. Korthikanti and G. Agha, “Energy-performance trade-off analysis of parallel algorithms for shared memory architectures,” Elsevier Journal of Sustainable Computing: Informatics and Systems, vol. 1, pp. 167–176, May 2011.

[56] M. Mazzucco and D. Dyachuk, “Optimizing Cloud providers revenues via energy efficient server allocation,” Elsevier Journal of Sustainable Computing: Informatics and Systems, vol. 2, pp. 1– 12, Nov. 2011.

[57] K. Li, “Optimal power allocation among multiple heterogeneous servers in a data center,” Elsevier Journal of Sustainable Computing: Informatics and Systems, vol. 2, pp. 13– 22, 2011.

[58] H. Chen, Y. Li and W. Shi, “Fine-grained power management using process-level profiling,” Elsevier Journal of Sustainable Computing: Informatics and Systems, vol. 2, pp. 33– 42, Jan. 2012.

[59] M. Bessi, et al., “Automatic memoization for energy efficiency in financial applications,” Elsevier Journal of Sustainable Computing: Informatics and Systems, vol. 2, pp. 105– 115, 2012.

[60] T. Nguyen, W. Shi, “Improving resource efficiency in data centers using reputation-based resource selection,” Elsevier Journal of Sustainable Computing: Informatics and Systems, vol. 2, pp. 138– 150, March 2012.

[61] A. Gandhi, et al., “Hybrid resource provisioning for minimizing data center SLA violations and power consumption,” Elsevier Journal of Sustainable Computing: Informatics and Systems, vol. 2, pp. 91– 104, Jan. 2012.

[62] B. Mondal, K. Dasgupta and P. Dutta, “Load Balancing in Cloud Computing using Stochastic Hill Climbing-A Soft Computing Approach,” Elsevier Journal of Procedia Technology, vol. 4, pp. 783 – 789, 2011.

[63] R. Gupta, et al., “A Two Stage Heuristic Algorithm for Solving the Server Consolidation Problem with Item-Item and Bin-Item Incompatibility Constraints,” in Proc. IEEE International Conference on Services Computing, 2008, pp. 39-46.

[64] Y. Xia, et al., “Analysis and Enhancement for Interactive-Oriented Virtual Machine Scheduling,” in Proc. IEEE/IFIP International Conference on Embedded and Ubiquitous Computing, 2008, pp. 393-398.

[65] D. Ye, H. Chen and Q. He,“Load Balancing in Server Consolidation,” in Proc. IEEE International Symposium on Parallel and Distributed Processing with Applications, 2009, pp.170-174.

[66] R. R. Harmon and N. Auseklis, “Sustainable IT Services: Assessing the Impact of Green Computing Practices,” in Proc. IEEE PICMET’09, Aug. 2-6, 2009, pp. 1707-1717.

[67] P. Yuan, et al., “Evaluating Dynamic Performance of VMM in Server Consolidation,” IEEE’09, pp. 81-85.

[68] Y. Song, et al., “Utility Analysis for Internet-Oriented Server Consolidation in VM-Based Data Centers” IEEE’09, pp. 1-10.

[69] Principled Technologies, “Virtual Machine Migration Comparison: Vmware Vsphere vs. Microsoft Hyper-V.” Principled Technologies Test Report. 2012, Available at www.principledTechnologies.com .

[70] T. V. T. Duy, Y. Sato and Y. Inoguchi, “Performance Evaluation of a Green Scheduling Algorithm for Energy Savings in Cloud Computing,” IEEE’10, pp. 1-8.

[71] N. Regola, J. C. Ducom, “Recommendations for Virtualization Technologies in High Performance Computing,” in Proc. 2nd IEEE International Conference on Cloud Computing Technology and Science, 2010, pp. 409-416.

[72] M. Cardosa, M. R. Korupolu, A. Singh, “Shares and Utilities based Power Consolidation in Virtualized Server Environments,” in Proc. IFIP/IEEE International Symposium on Integrated Network Management (IM 2009), 2009, pp. 327-334.

[73] H. Choi, et al., “A CPU Provision Scheme Considering Virtual Machine Scheduling Delays in Xen Virtualized Environment,” IEEE’09, pp. 1-6.

[74] J. W. Jang, et al., “Energy Reduction in Consolidated Servers through Memory-Aware Virtual Machine Scheduling,” IEEE Transactions on Computers, vol. 60, pp. 552-564, April 2011.

[75] Y. T. Lee and K. T. Chen, “Is Server Consolidation Beneficial to MMORPG? A Case Study of World of Warcraft,” in Proc. IEEE 3rd International Conference on Cloud Computing, 2010, pp. 435-442.

[76] K. Ye, et al., “Two Optimization Mechanisms to Improve the Isolation Property of Server Consolidation in Virtualized Multi-Core Server,” in Proc. 12th IEEE International Conference on High Performance Computing and Communications, 2010, pp. 281-288.

[77] Z. Gong, X. Gu, “PAC: Pattern-driven Application Consolidation for Efficient Cloud Computing,” in Proc. 18th Annual IEEE/ACM International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems, 2010, pp. 24-33.

[78] L. Niu, “Energy Efficient Scheduling for Real-Time Embedded Systems with QoS Guarantee,” in Proc. The Sixteenth IEEE International Conference on Embedded and Real-Time Computing Systems and Applications, 2010, pp. 163-172.

[79] H. Chen, et al., “VCAE: a Virtualization and Consolidation Analysis Engine for Large Scale Data Centers,” in Proc. Fourth IEEE International Conference on Self-Adaptive and Self-Organizing Systems, 2010, pp. 1-10.

[80] H. Chen, H. Jin, K. Hu, "Affinity-aware Proportional Share Scheduling for Virtual Machine System," in Proc. Ninth IEEE International Conference on Grid and Cloud Computing, 2010, pp. 75-80.

[81] D. Versick, D. Tavangarian, "Reducing Energy Consumption by Load Aggregation with an Optimized Dynamic Live Migration of Virtual Machines," in Proc. IEEE International Conference on P2P, Parallel, Grid, Cloud and Internet Computing, 2010, pp. 164-170.

[82] H. Tadokoro, K. Kourai, S. Chiba, "A Secure System-wide Process Scheduler across Virtual Machines," in Proc. IEEE Pacific Rim International Symposium on Dependable Computing, 2010, pp. 27-36.

[83] L. Kelly, "Building an optimised and cost-effective datacenter strategy." Oracle Whitepaper.[Last Accessed: December 2012].

[84] Y. Li, et al., "A Control Mechanism about Quality of Service for Resource Scheduling in Multiple Virtual Machine System," in Proc. The 11th IEEE International Conference on Parallel and Distributed Computing, Applications and Technologies, 2010, pp. 518-523.

[85] F. Kong, W. Yi, Q. Deng, "Energy-Efficient Scheduling of Real-Time Tasks on Cluster-Based Multicores," EDAA'10, pp. 1-6.

[86] W. Y. Lee, "Energy-Efficient Scheduling of Periodic Real-Time Tasks on Lightly Loaded Multicore Processors," IEEE Transactions on Parallel and Distributed Systems, vol. 23, pp. 530-537, March 2012.

[87] L. Niu, "System-Level Energy-Efficient Scheduling for Hard Real-Time Embedded Systems," EDAA'11, pp. 1-4.

[88] S. K. Bose, et al., "CloudSpider: Combining Replication with Scheduling for Optimizing Live Migration of Virtual Machines Across Wide Area Networks," in Proc. 11th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, 2011, pp. 13-22.

[89] X. J. Chen, J. Zhang, "Virtual Machines Scheduling Algorithm Oriented Load Forecast," in Proc. IEEE International Conference on Network Computing and Information Security, 2011, pp.113-117.

[90] C. R. Chang, J. J. Wu, P. Liu, "An Empirical Study on Memory Sharing of Virtual Machines for Server Consolidation," in Proc. Ninth IEEE International Symposium on Parallel and Distributed Processing with Applications, 2011, pp. 244-249.

[91] C. H. Hsu, S. C. Chen and C. T. Yang, “Dynamic Voltage Adjustment for Energy Efficient Scheduling on Multi-Core Systems,” in Proc. Ninth IEEE International Symposium on Parallel and Distributed Processing with Applications Workshops, 2011, pp. 197-202.

[92] D. Zhang, F. Chen, S. Jin, “Global EDF-based Online, Energy-efficient Real-time Scheduling in Multi-core Platform,” IEEE’11, pp. 666-670.

[93] S. Videv and H. Haas, “Energy-Efficient Scheduling and Bandwidth–Energy Efficiency Trade-Off with Low Load” in Proc. IEEE ICC, 2011, pp. 1-5.

[94] A. Corradi, M. Fanelli, L. Foschini, “Increasing Cloud Power Efficiency through Consolidation Techniques,” IEEE’11, pp. 129-134.

[95] M. Marzolla, O. Babaoglu, F. Panzieri, “Server Consolidation in Clouds through Gossiping,” IEEE’11, pp. 1-6.

[96] R. Jansen, P. R. Brenner, “Energy Efficient Virtual Machine Allocation in the Cloud,” IEEE’11, pp. 1-8.

[97] K. Ye, et al., “Live Migration of Multiple Virtual Machines with Resource Reservation in Cloud Computing Environments,” in Proc. IEEE 4th International Conference on Cloud Computing, 2011, pp. 267-274.

[98] C. C. Lin, P. Liu, J. J. Wu, “Energy-Aware Virtual Machine Dynamic Provision and Scheduling for Cloud Computing,” in Proc. IEEE 4th International Conference on Cloud Computing, 2011, pp. 736-737.

[99] H. Castro, et al., “Green Flexible Opportunistic Computing With Virtualization,” in Proc. 11th IEEE International Conference on Computer and Information Technology, 2011, pp. 629-634.

[100] A. G. Guirado, et al., “Energy-Efficient Cache Coherence Protocols in Chip-Multiprocessors for Server Consolidation,” in Proc. IEEE International Conference on Parallel Processing, 2011, pp. 51-62.

[101] D. S. Zhang, et al., “An Energy-Efficient Scheduling Algorithm for Sporadic Real-Time Tasks in Multiprocessor Systems,” in Proc. IEEE International Conference on High Performance Computing and Communications, 2011, pp. 187-194.

[102] W. Liu, H. Yin, W. Du, “Dynamic Threshold-Based Energy Efficient Scheduling Algorithm for Parallel Tasks on Homogeneous DVS-Enabled Clusters,” in Proc. IEEE International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, 2011, pp. 321-328.

[103] H. Lee, et al., “Profile-based Building Energy Saving Service in Green Computing Environment,” in Proc. IEEE ICTC, 2011, pp. 768-769.

[104] X. Zhu, C. He, and J. Wang, “Energy-Efficient Elastic Scheduling in Heterogeneous Computing Systems,” IEEE’11, pp. 1-8.

[105] C. G. Renaud, et al., “The Green Computing Observatory: a data curation approach for green IT,” in Proc. Ninth IEEE International Conference on Dependable, Autonomic and Secure Computing, 2011, pp. 798-799.

[106] N. T. Hieu, et al., “A Framework for Scheduling Virtual Machines to Support Real-time Services for U-Life Care,” in Proc. 12th IEEE International Conference on Parallel and Distributed Computing, Applications and Technologies, 2011, pp. 128-133.

[107] Y. Ho, P. Liu, J. J. Wu, “Server Consolidation Algorithms with Bounded Migration Cost and Performance Guarantees in Cloud Computing,” in Proc. Fourth IEEE International Conference on Utility and Cloud Computing, 2011, pp. 154-161.

[108] S. Zhuravlev, et al., “Survey of Energy-Cognizant Scheduling Techniques,” IEEE Transactions on Parallel and Distributed Systems, pp. 1-19, 2012.

[109] B. Gerofi and Y. Ishikawa, “Workload Adaptive Checkpoint Scheduling of Virtual Machine Replication,” in Proc. 17th IEEE Pacific Rim International Symposium on Dependable Computing, 2011, pp. 204-213.

[110] C. H. Hsu, et al., “Energy-Aware Task Consolidation Technique for Cloud Computing,” in Proc. Third IEEE International Conference on Coud Computing Technology and Science, 2011, pp. 115-121.

[111] S. K. Doddavula, M. Kaushik, A. Jain, “Implementation of a Fast Vector Packing Algorithm and its application for server consolidation,” in Proc. Third IEEE International Conference on Coud Computing Technology and Science, 2011, pp. 333-339.

[112] S. Lee, S. Sahu, “Efficient Server Consolidation Considering Intra-Cluster Traffic,” in Proc. IEEE Globecom, 2011, pp. 1-6.

[113] J. Tian, Y. Du, H. Yu, “Characterizing SMP Virtual Machine Scheduling in Virtualization Environment,” in Proc. IEEE International Conferences on Internet of Things, and Cyber, Physical and Social Computing, 2011, pp. 402-408.

[114] M. M. Taheri and K. Zamanifar, “2-Phase Optimization Method for Energy Aware Scheduling of Virtual Machines in Cloud Data Centers,” in Proc. Sixth IEEE

International Conference on Internet Technology and Secured Transactions, 2011, pp. 525-530.

[115] Y. Jin, Y. Wen and Q. Chen, "Energy Efficiency and Server Virtualization in Data Centers: An Empirical Investigation," First IEEE INFOCOM Workshop on Communications and Control for Sustainable Energy Systems: Green Networking and Smart Grids, 2012, pp. 133-138.

[116] K. Dutta, et al., "A Smart Job Scheduling System for Cloud Computing Service Providers and Users: Modeling and Simulation," in Proc. 1st IEEE Int'l Conf. on Recent Advances in Information Technology, 2012, pp. 1-6.

[117] L. Liu, et al., "DMSS: A Dynamic Memory Scheduling System in Server Consolidation Environments," in Proc. IEEE 15th International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing Workshops, 2012, pp. 70-75.

[118] L. Sawalha and R. D. Barnes, "Energy-Efficient Phase-Aware Scheduling for Heterogeneous Multicore Processors," IEEE'12, pp. 1-6.

[119] T. Setzer and A. Wolke, "Virtual Machine Re-Assignment Considering Migration Overhead," in Proc. IEEE Network Operations and Management Symposium (NOMS), 2012, pp. 631-634.

[120] H. Goudarzi, M. Ghasemazar, and M. Pedram, "SLA-based Optimization of Power and Migration Cost in Cloud Computing," in Proc. 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, 2012, pp. 172-179.

[121] Vmware, "How VMware Virtualization Right-sizes IT Infrastructure to Reduce Power Consumption." Vmware Whitepaper. Available at http://www.vmware.com/solutions/consolidation/green/pge.html/ [Last Accessed: December 2012].

[122] Q. Huang, et al., "Enhanced Energy-efficient Scheduling for Parallel Applications in Cloud," in Proc. 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, 2012, pp. 781-786.

[123] J. Li, et al., "Reducing Operational Costs through Consolidation with Resource Prediction in the Cloud," in Proc. 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, 2012, pp. 793-798.

[124] P. E. Rocha, L. C. E. Bona, "A QoS Aware Non-work-conserving Disk Scheduler," IEEE'12, pp. 1-5.

[125] S. Sotiriadis, et al., “Cloud virtual machine scheduling: Modelling the cloud virtual machine instantiation,” in Proc. Sixth IEEE International Conference on Complex, Intelligent, and Software Intensive Systems, 2012, pp. 233-240.

[126] X. Ling, et al., “Efficient Disk I/O Scheduling with QoS Guarantee for Xen-based Hosting Platforms,” in Proc. 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, 2012, pp. 81-89.

[127] K. Halder, U. Bellur and P. Kulkarni, “Risk Aware Provisioning and Resource Aggregation based Consolidation of Virtual Machines,” in Proc. IEEE Fifth International Conference on Cloud Computing, 2012, pp. 598-605.

[128] A. Beloglazov and R. Buyya, “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, pp. 1-14, 2012.

[129] F. Wu, S. Jin, Y. Wang, “A Simple Model for the Energy-Efficient Optimal Real-Time Multiprocessor Scheduling,” IEEE’12, pp. 18-21.

[130] J. Li, et al., “Energy-Efficient Scheduling in Nonpreemptive Systems With Real-Time Constraints,” IEEE Transactions on Systems, Man, and Cybernetics, IEEE’12, pp. 1-13.

[131] A. Pahlavan, M. Momtazpour, M. Goudarzi, “Data Center Power Reduction by Heuristic Variation-Aware Server Placement and Chassis Consolidation,” in Proc. 16th CSI International Symposium on Computer Architecture and Digital Systems, 2012, pp. 150-155.

[132] S. G. Kim, H. Eom, H. Y. Yeom, “Virtual Machine Scheduling for Multicores Considering Effects of Shared On-chip Last Level Cache Interference,” IEEE’12, pp. 1-6.

[133] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, “Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for Delivering Computing as the 5th Utility,” Future Generation Computer Systems, vol. 6, pp. 599-616, Elsevier Science, Amsterdam, The Netherlands, June 2009.

List of Papers

Accepted

[1] Taranpreet Kaur and Inderveer Chana, "Power-Aware Virtual Machine Scheduling-policy for Virtualized Heterogeneous Multicore Systems," Second International Conference on Advances In Electronics, Electrical And Computer Engineering (EEC-2013), Dehradun, 22-23 June, 2013.

Communicated

[2] Taranpreet Kaur and Inderveer Chana, "Energy Efficient Virtual Machine Management in Cloud: An Exploratory Study," International Journal of Computational Intelligence and Electronic Systems, Vol. 3, August, 2013.