



Improving Energy Efficiency in Cloud Computing Data centres Using Intelligent Mobile Agents

Ogechukwu M. Okonor

Submitted in partial fulfilment of the requirements for the degree of

Doctor of Philosophy (PhD)

School of Computing

University of Portsmouth

January 2021

@Okonor, 2021

Copyright ©2021 Ogechukwu M. Okonor

All rights reserved. No part of this Oge-thesis may be reproduced in any form or by any means, electronic or mechanical printing, photoprint, recorded or any part of the information stored/retrieved written without permission from the author except as permitted by law.

ABSTRACT

Cloud computing has become a profound name and also a solid bedrock for new emerging technology. Cloud technology fully supports the dynamic provisioning of computing resources as a utility service on a pay-as-you-go approach. Its numerous benefits (such as rapid elasticity, flexibility, network resource pooling) empower small, medium, and large enterprises to use other technologies through the Internet's flexibility. Despite the benefits cloud technology offers, the challenge of the high power consumption rate it incurs as it leverages its promised attributes remains a major concern. Academic research has shown that a typical 500 square meter data centre consumes about 27,048 kilowatts per hour of power per day regardless of whether it is active or not. Over the years, the most dominant energy efficient techniques for managing data centre has been Dynamic voltage frequency scaling and virtual machine consolidation, which has had a significant setback due to its inability to manage systems on overload state. Therefore, a novel paradigm based on an intelligent mobile agent approach has been proposed. This proposed approach is highly intelligent can easily detect underutilised and overloaded components of the data centre due to its unique feature. Agent technique has successfully shown it can prevent and manage overloading issues due to change in workloads and achieve a more efficient load balancing with a low power consumption rate. The mobile agent was embedded into servers and switches to regulate their activities and then shut down underutilised components. Mobile agent (Java agent) is the first of its kind used in a cloud environment. This research proposal saves a significant amount of energy and improves the entire system performance.

In this thesis, the intelligent-based Agent approach is used to address energy efficiency and cost-aware related problems. Agent approach helps facilitate resource management, allocation of cloud data centre components with a significant reduction in energy usage rate with a more efficient system performance while maintaining a highly reliable system as promised by service providers.

DECLARATION

This is to certify that

1. The thesis comprises only of my original work toward the award of PhD
2. Adequate referencing has been made in the text to all other published authors used while producing this research work.
3. This thesis is more than 30,000 words in length according to the PhD thesis requirement, excluding tables, figures, bibliographies and appendixes.

ACKNOWLEDGEMENT

Oh, how time flies and my journey of four years now looks like yesterday. I still remember the first day I came to the University of Portsmouth faculty building to do my 1st-year registration with my one-year-old daughter. She was screaming on top of her voice, having seen so many new faces suddenly.

I am very grateful to Almighty God, who, through His grace, help, provisions and direction, made this day a reality. He is my source of strength and inspiration during my winning and failing moments.

I am thankful to my supervisory team, Dr Mo Adda, my first supervisor and Dr Alex Geogov, my second supervisor. I sincerely want to say a special thank you to Dr Mo Adda for the opportunity he gave me to turn my dream into a reality today. He gave me the support, insight, expert advice, and encouragement through my PhD journey and the financial struggles that came with it.

I also acknowledge the head of the department of computing Dr Nick for assisting PhD students with adequate funds to develop and enhance the research knowledge through paid training, workshops and conferences.

My deepest gratitude goes to my husband for the considerable financial, moral and emotional support he gave me, then to my kids (Victoria, Emmanuel and Annabelle), who tolerated my constant absence from my daily routine because of this PhD work. Your sacrifices and support have given me the voice to say, "I am here now boldly".

Finally, to every other individual who has contributed in one way or the other during this PhD journey to making me come to this final part of this academic journey, I say a big thank you. I appreciate your incredible effort.

CONTENTS

ABSTRACT	iii
DECLARATION	iv
ACKNOWLEDGEMENT	v
CONTENTS	vi
LIST OF FIGURES	viii
LIST OF TABLES	x
Chapter1	1
Introduction.....	1
1.1 Research Overview	1
1.2 Motivations	8
1.3 Research Questions and Objectives.....	11
1.4 Methodology.....	13
1.5 Research Contributions and Publications	14
1.6 Thesis Structure	17
Chapter2	18
Literature Review.....	18
2.1 Introduction.....	18
2.2 Background.....	20
2.3 Article Selection Justification.....	24
2.4 Review and Taxonomy of cloud power Management System.	25
2.5 Intelligent Mobile Agent Approach.....	45
2.6 Summary.....	48
Chapter3	49
Methodology	49
3.1 Introduction	49
3.2 Research Methodology	49
3.3 Description of Algorithm.....	65
3.4 Development Environment.....	66
3.5 Simulation data Source	73
3.6 Evaluation	76
3.7 Summary.....	77
Chapter4	78

Results and Evaluation.....	78
4.1 Introduction	78
4.2 Virtual Machine Migration Result Discussion	78
4.4 Cost of managing cloud data centre's system using mobile agent technique	90
4.5 Service level agreement violation result	92
4.6 Summary	93
Chapter5	95
Semantic Knowledge Representation	95
5.1 Introduction	95
5.2 Overview of Semantic Ontology	95
5.3 Related Work.....	96
5.4 Semantic knowledge representation of power management techniques.....	97
5.5 Semantic ontology of agent-based system.....	99
5.6 Semantic knowledge representation of an agent-based system with cloud data centre solutions	101
5.7 Summary	103
Chapter6.....	104
Conclusion and Future Work	104
6.1 Introduction.....	104
6.2 Summary and Contributions	104
6.3 Future Work.....	106
6.4 Summary	108
References.....	110
Appendix.....	118

LIST OF FIGURES

Figure 1.1: Cloud Computing networks diagram	3
Figure 1.2: Power consumption rate (TWh) of the data centres from 2002 to 2025	5
Figure 1.3: The methodology framework for this research work	13
Figure 1.4: The Thesis Structure	16
Figure 2.1: Taxonomy of a data centre	21
Figure 2.2: Systematic Chart representation of DC energy breakdown	22
Figure 2.3: Taxonomy of power management technique	24
Figure 2.4: Comprehensive DPM techniques	26
Figure 2.5: power consumption at the various levels of the cloud computing system	28
Figure 2.6: Precise critical points of areas to look out for when working on a cloud system	29
Figure 2.7: The application-level was divided into the cloud management system (CMS) and the appliances sections	36
Figure 2.8: Related works on Various Task scheduling	40
Figure 2.9: Resource scheduling in cloud system	43
Figure 3.1: System model design	50
Figure 3.2: The mobile agent performance lifecycle on the cloud data centre system	61
Figure 3.3: Mobile agent deliverables to Cloud Data centre	62
Figure 3.4: Flowchart for system power optimisation process of Cloud DC	63
Figure 3.5: An intelligent technique for migrating VM for power saving	64
Figure 3.6: Usable classes in Cloudsim 3.3.0	68
Figure 3.7: AgentCloudSim Framework	70
Figure 3.8: Data centre structure	71
Figure 3.9: Flowchart for implementing VM allocation using agent technique	73
Figure 4.1: Amount of energy saved at each virtual machine migration period	77
Figure 4.2: Impact of mobile agent technique on cloud data centre system	78
Figure 4.3: Power usage rate during migration	79
Figure 4.4: Host power simulation behaviour under VM migration	80
Figure 4.5: Comparison analysis of three different VM threshold	82

Figure 4.6: Comparing agent policy to other policies in terms of energy efficiency metric	83
Figure 4.7: Resultant display of the power consumption level of the two agents types	84
Figure 4.8: Performance evaluation of all three agent types on the server	85
Figure 4.9: Shows the power consumption of all the different agent and non-agent scenario.....	85
Figure 4.10: Compares the performances of traditional DC power usage rate with Mobile agent-based DC	86
Figure 4.11: Comparing mobile agent policy to other existing policies	87
Figure 4.12: The throughput percentage Performance of the system at all stages	89
Figure 5.1: Semantic web representation of power management techniques	95
Figure 5.2: Taxonomy of agent-based system	97
Figure 5.3: Semantic display of agent characteristic	98
Figure 5.4: Cloud data centre components under agent influence	99
Figure 5.5: Agent-types used in this research work	99

LIST OF TABLES

Table 1.1: Breakdown of data centre past and predicted energy consumption rate in a data centre	6
Table 2.2: Related work on energy efficiency in a cloud wired and wireless system	31
Table 2.3: Tabulate Literature review on the cloud network domain	35
Table 2.4: A tabulated review of virtual machines related works	39
Table 2.5: Literature survey on task scheduler algorithms	42
Table 3.1: Significant energy-efficient cloud network research investigators	49
Table 3.2: Server's specification	65
Table 3.3: Review of selected simulation tools	66
Table 4.1: Workload based characteristics of the CPU utilisation	78
Table 4.2: Data configuration component and its power output on different agent scenario	83
Table 4.3: Total and split host and switch power consumption	84
Table 4.4: Average (SR) violation encounter by a different technique in the real workload	90

Glossary of Terms

ACPI	Advanced Configuration and Power Interface
APM	Advanced Power Management
AWS	Amazon Web Service
CSCI	Climate Saver Computing Initiative
DC	Data Centre
DCD	Dynamic Component Deactivation
DFS	Dynamic Frequency Scaling
DPM	Dynamic Power Management
DPS	Dynamic Performance Scaling
DVFS	Dynamic Voltage Frequency Scaling
DVS	Dynamic Voltage Scaling
EPA	Environmental Protection Agency
GCIO	Green Computing Impact Organisation
ICT	Information and Communications Technology
IoT	Internet of Things
IQR	Interquartile Range
LLC	limited Lookahead Control
LR	Linear Regression
MAD	Mean Absolute Deviation
OCF	Open Compute Project
ODM	Original Design Manufacturer
OSI	Operation System Interconnection
OWL	Web Ontology Language
PODs	Performance Optimised Data Centre Modules
QoS	Quality-on-Service
RPC	Remote Procedure Call
SR	System reliability
SNMP	Simple Network Management Protocol
SPM	Static Power Management
THR	Threshold Ratio
UML	Unified Modelling Language
VM	Virtual Machines

Chapter1

Introduction

1.1 Research Overview

Cloud computing has become the new norm for interconnection and communication in the Information Technology (IT) world. Cloud technology's recent growth can be attributed to the massive increase in Internet-based activities and Internet users. Cloud computing has firmly established that it is the most cost optimisation IT innovation for enterprise use. It leverages computing resources to the small, medium and struggling businesses, giving them the tremendous opportunity to strive and compete fervently with other big enterprises. It is a technology that aims for development with little or no constraints due to its flexibility of usage via virtualisation and service-oriented software. Cloud Technology brought about three dominant differing approaches to the way computing resources are used with zero-level stress and the upfront cost of purchasing infrastructure, buildings, and IT supply management. These approaches are: - pay-as-you-go, elasticity and on-demand provisioning. Cloud computing innovation provides IT resources in the form of utility such as our daily use of water and gas. Cloud computing gives access to a great wealth of computing elements, storage, networks and software from a personal or public local server to required specific client hosted by Facebook, Amazon Web Service (AWS), Microsoft Azure, Google and others. Cloud innovation has now gained significant attention across the industry, business, research and academic sections due to its performance attributes (usage flexibility, rapid resource pooling, the ubiquity of the network, etc.) and its fastest-growing segment IT spending.

Cloud computing can be defined as delivering computing services through the Internet (simply referred to as the cloud). However, there is a standard definition of cloud computing based on the National Institute of Standards and Technology (NIST) standards which says that "cloud computing is a model that enables ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort by the service provider" (*Mell & Grance*). These

services include software, database, servers, storage, networks, etc. and can be deployed through any of these three cloud service levels models to users as- (i) Infrastructure-as-a-Service (IaaS), which provides the computing, networking and storage resources (ii) Platform-as-a-Service (PaaS) which provides the valuable tool that facilitates clouds application deployment, and (iii) Software-as-a-Service (SaaS) which gives the user access to provider's available software on cloud networks. These different service levels depict the summary of all the different services provided by the cloud. IaaS is the most common cloud computing services. With IaaS, you can rent necessary IT infrastructure (e.g. servers, virtual machines, networks, operating systems, etc.) on a pay-as-you-go basis. PaaS provides an on-demand platform for developing, testing, and managing software. This makes it easy for developers to do their job without worrying about the underlying infrastructure, which can cost a fortune to achieve. Software-as-a-service is a type of service that allows cloud customers to use any type of software provided by their chosen cloud providers. Thus, the cloud customer does not need to worry about any subscription, maintenance, security etc., apart from paying for the services they used. Various cloud platforms have been built with substantial infrastructural provisions to support vast applications worldwide, assuring resources are scalable, reliable and available to clients when needed. This gives the cloud users the choice to use computing resources based on their business requirements and when the need arises.

Cloud computing has established a robust network communication flow classified into two main parts; the cloud-to-user flow and the intra-cloud flow. Compare this traffic flow to the era of fewer cloud activities when legacy local computers were the norm that holds all the data and software for the user. Now, correspondences are received on request from the cloud data centre. The volume of data processing per second globally is getting so massive, which corresponds to Cisco's whitepaper report that network communication traffic flow is the fastest-growing data centre component, rising to 4.3 zebibytes (ZiB) in 2016 with a combined annual growth rate of 44% (Cisco, 2013).

Therefore, the data centre is the backbone of cloud activities because cloud service providers use data centres to deploy their services in different geographical locations. However, it is worth noting that cloud computing is not the same as a data centre. While cloud stores data on the Internet, data centres do the same but within the enterprise's local network. A typical data centre is comprised of a plethora of smaller networks that contain processing servers and storage devices. These networks are interconnected through many network switches to form an

overarching data centre (DC) architecture. The design of DC architecture must provide a means to being efficient, dynamic, resilient and function 24/7, which it sometimes fails to achieve. This brings me to explaining the component contentment of DC. A typical data centre consists of the following components: core router, processing servers, layered network switches, and Virtual Machines (VM), each strategically positioned and connected to meet clients' needs based on the DC requirements. Currently, the cloud data centre contains tens of thousands of servers connected to the storage and then service the entire globe. These connection processes put massive pressure on the data centre networks to maintain a certain low cost, high bandwidth and low power usage level. To ensure that the cloud data centre system operates with a minimal service cost without violating the promised system reliability, service providers acquire specialised server boards and networking equipment built by Original Design Manufacturer (ODM) with specific workload requirements. For instance, Facebook now uses Open Compute Project (OCP) with better optimises server while Microsoft uses Performance Optimised Data Centre modules (PODs). Figure1.1 below shows how cloud platforms accommodate many computing components, applications, and deliveries to many customers and enterprises worldwide.

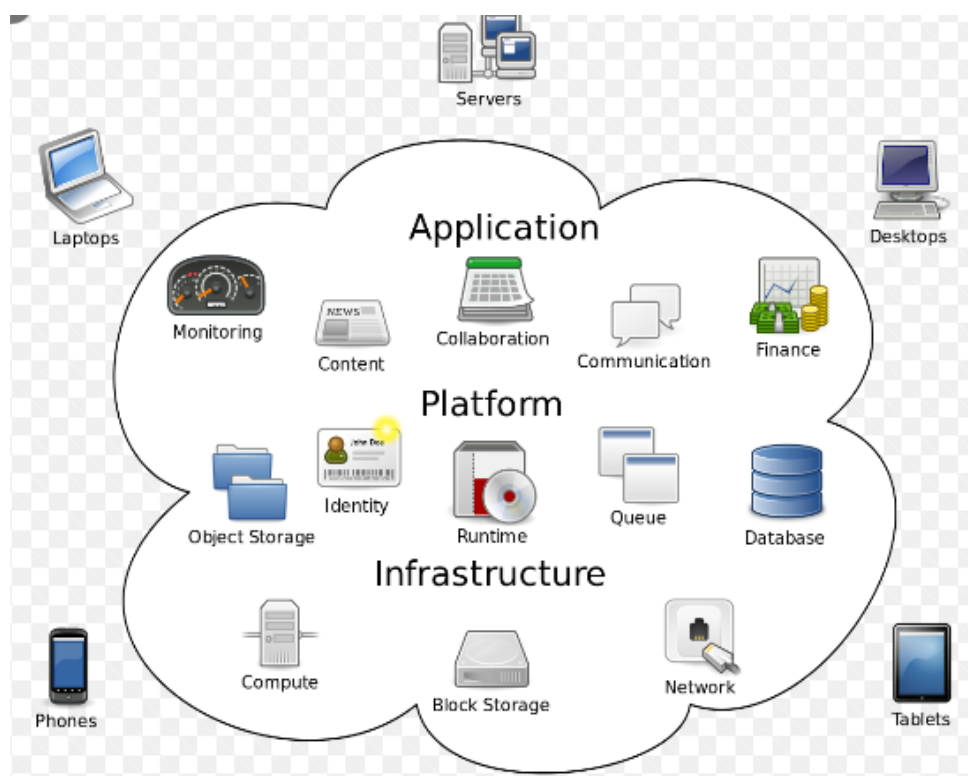


Figure 1.1: Cloud Computing networks diagram (researchgate.net)

Despite the immense efforts put by service providers to ensure that cloud technology delivers on its promise while maintaining a high level of quality-on-service (QoS), it still faces impending challenges, which are of significant concern to providers and researchers for the sake of sustainability. A case study by (Mittal, Sharma & Huang, 2012) showed that more than 100 million videos are watched via YouTube per day. At the same time, Facebook has more than 400 million active users and 3 billion photos uploaded every month. These activities have exponentially increased due to the pandemic that has left many people in lockdown situation in their home with minimal outdoor activities based on the restrictive government rules. According to (Linux foundation training, 2019) article, more than 850,000 people purchase smartphone daily, and 700,000 smart TVs are sold every day. Data centres power all these activities through their computation-intensive software programs, which underscore the amount of energy usage. This leaves researchers wondering what happens to this cloud innovation if nothing is done about its current high Power consumption rate challenge. Figure 1.2 illustrated the past and predicted energy consumption rate in the cloud data centre between 2006 and 2025. The data gathered from the figure showed exponential growth between 2006 to 2013 and is still growing to date, not on an exponential curve as predicted. Still, far more than the expected growth level has been recorded, which bring about an immediate concern. Data centres are global infrastructure, which means it is located in several regions with different availability zones. The US data centre region alone consumed 100 billion kilowatt-hours kWh in 2015, which is enough to supply electricity to six underdeveloped countries for 3 years (Okonor, Adda and Gegov, 2019). The author (Bawden, 2016) forecast that the electricity consumption of the data centres in the US region would double to at least 150 billion (kWh) by 2022 (Pompili, Hajisami and Tran, 2016), which will be 50% increase on its power consumption rate. Google also acknowledges that each Internet data research per second takes about 0.003 kWh of power, which can power a 60 watts (W) light bulb for 17 seconds (Google, 2009). Putting this figure in a more perspective, knowledge represents the number of people who access the Internet data at a given time, especially in this case of the world having a pandemic and being in a state of national lockdown.

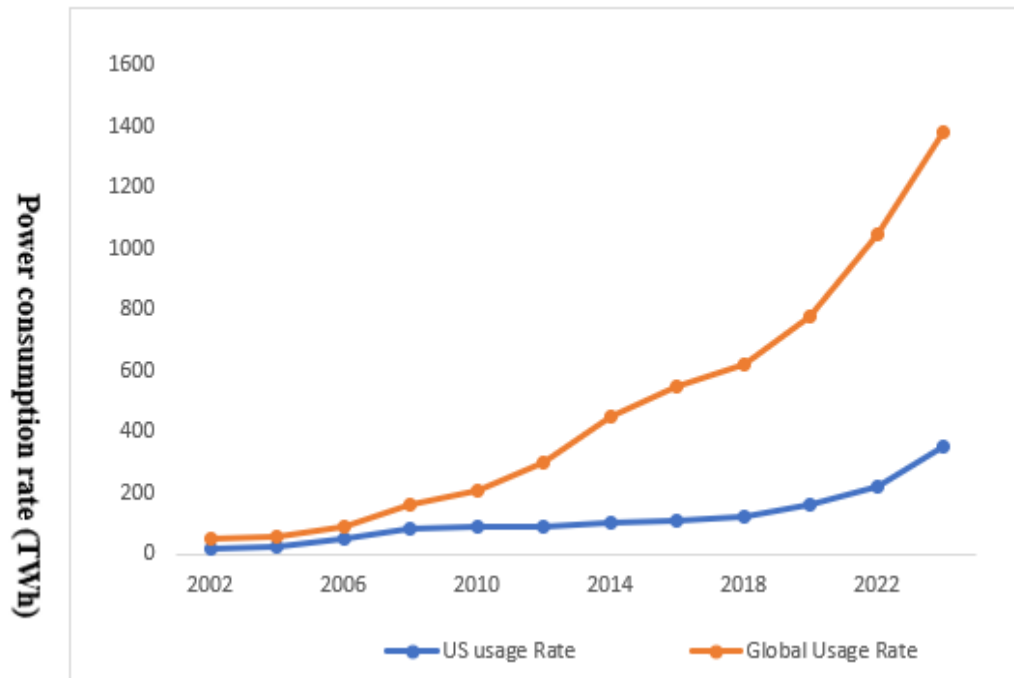


Figure 1.2: Power consumption rate (TWh) of the data centres from 2002 to 2025

Globally, data centres account for 1% (205 TWh) of the total electricity used in 2018 by the world (Masanet et al., 2020, Pearce, Fred, 2018). Anders Andrea, a specialist in sustainable ICT infrastructure, predicted in 2019 that the global consumption rate of the data centres would be more by 8% rather than the initial prediction of 21%.

Currently, watching the rate at which many firms have suddenly migrated to cloud-based workforce mode due to the world pandemic, it is inevitable that there will be an exponential surge again in the amount of energy consumed by the data centres will also exceed what has been predicted by (Andrea, Gelenbe and Girolam, 2016). However, it corresponds with the article published by (Nature 2018), which forecasted that the data centres' annual electricity demands could rapidly grow to as much as 8,000 terawatt-hours (TWh) by 2030 worst-case scenario and as low as 1,1000 TWh under the best-case scenario. Based on the published fact from the Simplian training group on 18 Dec. 2020(Simplian annual report, 2020) showed that as of mid-April 2020, there were 4833 million Internet users globally in contrast to Dec. 2020 when it has suddenly jumped to 50 billion Internet and smart device active users. Table 1.1 shows a total breakdown of the scientific report of data centre energy consumption through

different vendor experiences, considering the global data centre and the US data centre power consumption rate. Note the table below depicts the following abbreviations: Estimated Annual Energy Cost (EAEC), Estimated Energy Consumed (EEC) and Energy Used by Data Centres (EUDC).

Table 1.1: Breakdown of DC past and predicted energy consumption rate in a data centre

Year	Investigator(s)	EAEC (USD Billion)	EEC (TWT)	EUDC (%)	Region
2005	Miyuru et al.		0.061	1	US
2006	R. Brown et al.	4.5	0.089	1.5	Globally
2007	Van Heddeghem		216	1.8	Globally
2009	G.Meijer et al.	30		2	Globally
2011	Google vendor	163	2.68	10.2	US
2012	Gartner Report	106.4		12.7	US
2013	Ni Lui		91	13.2	US
2016	X. Fan et al.	240			US
2017	Marriott Marquis		102	14.1	US
2017	Atlanta		416	3	Globally
2019	Energy watch Manager	10.77	424	4.7	Globally

Furthermore, although cloud computing technology does not consume a significant amount of energy, its energy consumption rate has a high impact on the environment (Data centre energy article,2014). Therefore, for the world to enjoy a greener environment while enjoying the benefits of cloud innovation, measures must be put to minimise carbon dioxide (CO₂) emissions. Already it has been reported by (Moghaddam and Cheriet, 2015) (Whitehead, Andrews, Shah and Maidment, 2014) that 78.7 million metric tons of CO₂ are emitted by the data centres, which is 2% of global emissions from cloud data centre power supply system. A forecast by (Lima 2017) predicts that data centres would add 5.5% to the world's carbon footprint by 2025, while (Data economy, 2017) reports that the cloud data centres emit the same CO₂ as the commercial airline industry. Recently prominent cloud vendors like Google,

Amazon, IBM and Microsoft Azure formed a community known as Green Grid (Beloglazov and Buyya, 2013) with the sole aim of minimising the impact of data centre emission to the environment through promoting a more efficient way of operating the data centres. Drastically reducing the amount of heat dispensed in the environment in the form of greenhouse gas emissions during data centre operational time is a promising option; however, many cloud services struggle to achieve these measures.

Based on the discussed energy-efficiency related challenges, it is therefore so important for researchers to find a lasting solution to this cloud data centres impending issue. Researchers from all fields have proposed different processing and resource management approaches based on resource scheduling, system architectural design, virtualisation algorithms, and even locating servers to cold climates (or underwater). Some suggested demand-side management, while others concentrate on sector coupling, selling the waste heat generated to gain efficiency. However, all these proposed solutions still do not produce the desired result and remain a massive challenge for future researchers to investigate. Some researchers argue that data centres should be located in cold climate areas in order to minimise the cooling expenses and effort of managing heat utilisation during cloud data centre runtime. However, this practice will not encourage a high adoption rate of cloud technology if it can only be built in a particular region. Does the question now remain as to what proportion of the world's regions have a steady cold climate with no disruption caused by preceded and unprecedented weather change?

Therefore, to ensure a considerable level of cloud sustainability, a more holistic and pragmatic management approach needs to provide a solution to resource scheduling and architectural performance issues with the data centres in order to improve energy efficiency. This then throws a glimpse into the unnegotiable importance of this research work. Specifically, this thesis advances the state-of-art by introducing the following pragmatic management ways:

1. An approach for scheduling cloud application components with intelligent mobile agents. The intelligent agent's approach models DC system application components which are classified as critical and non-critical based on their performance strength. An agent-based algorithm was introduced in this research work; the system behaviour determined its performance. In addition, a time-based agent or daemon-based agent was used in this work to regulate, maintain, and monitor system performance based on a specific requirement aimed at reducing the system power usage rate.

-
2. A new mathematical model was introduced to improve the cloud data centre system performance while evaluating factors that affect power rate. Subsequently, most of the DC components are ignored, which may be factors that lead to an increase in the data centre high energy consumption. Therefore, this proposed model formulated a more holistic, sustainable DC management technique. Furthermore, this proposed model gave a distinct view of the data centre components that have been downplayed over the years, significant factors that increase the energy usage rate.
 3. Semantic knowledge representation of cloud data centre architectural state design was also introduced and developed in this work. Using semantic knowledge in a cloud context is novel. Moreover, it gives a simple explanation of the complexity of cloud network understanding, enabling service providers to be aware of components with more power-related issues.

1.2 Motivations

The evolution of Information and Communication Technologies (e.g. Internet of Things (IoT) based applications), and the emergence of new web/multimedia applications are all underpinned by a cloud service. Observing the trend, people and industry adopt cloud technology and its promised benefit. Recent research shows that more than 75% of young people and small businesses use cloud aided technology directly or indirectly. In order to meet up with the demand of the recent cloud activity explosion, most Cloud service provider's resort to the deployment of very high bandwidth and power-hungry equipment. Therefore, it is no doubt that this striving technology needs to have an excellent sustainability strategy to withstand its current and upcoming challenges. The energy efficiency challenge is overwhelming, and it is also the foundation of some of the other problems associated with cloud computing networks. A simple sample question is if the US data centre region alone used more than 100 billion kilowatt-hours (kWh) in 2015, then how can striving countries like Nigeria generates just 4000 (kWh) of electricity or South Africa with a total annual electricity supply of 51,309 (kWh) sustain this high energy spilt?

Therefore, evaluating the impact of high energy consumption of the DC from the Nigerian perspective based on US usage will create a vast problem because currently, Nigeria generates 80% less power below consumer satisfaction based on its high population density. Hence,

making it a lot more difficult for such a country with an existing shortage in power supply to venture into cloud adoption if nothing is done toward reducing its high energy consumption issues.

The enormous high-power usage demand on the cloud data centre is eye-catching and needs urgent attention to encourage regional system management. There is an even more exponential increase in its usage rate now. The world is experiencing a new norm in how business and people operate daily due to the unprecedented pandemic condition. More people thereby get to use cloud application and search the Internet for different reasons. Under a reasonable circumstance, requested applications are processed within a required time duration, thus expecting a very effective and robust data centre network to accomplish this task without compromising its system reliability. Therefore, to ensure cloud innovation is sustainable, there must be a "trade-off between performance and efficiency." The data centre network should be designed to have an efficient infrastructural component, with its resources duly managed and monitored. To adequately address this rising challenge, cloud vendors, service providers, data centres and network design engineers should be energy efficiency-oriented and work toward minimising power usage rate during runtime. Furthermore, the concept of green computing should be embedded in the governing policies for operating the cloud data centre as the service tries to maintain a set QoS (quality of service).

Many factors contribute to the data centre's high energy consumption rate. These factors can be technical, infrastructural, environmental and management. From a technical perspective, some cloud data centres still use the ODM version of the data centre's old technical setting but upgrade their infrastructural components. This factor leads to a system generating more heat and not having the adequate proportion of the heat-resistor, the system requirement to reduce the amount of heat it emits.

The extremely high energy consumption from computing, networking resources and the power inefficiency of hardware are the infrastructural factors while the inefficient usage of these facilities is management. According to the data collected from more than 5000 production servers over 6 months, it is clearly shown that although servers usually are not idle, the utilisation rarely approaches 100% (Beloglazov et al.). Subsequently, servers operate at 10% – 50% of their full capacity, leaving 50% as an active wasting resource. This leads to extra expenses on overprovisioning and the additional total cost of managing unused access (Pedram, 2012) (Oro, Depoorter, Garcia, and Salom, 2015).

Environmentally, some parts of the world are hotter than others, jeopardising their chances of setting up their physical cloud data centre. For instance, in the UK, very humid temperatures will automatically reduce the cooling system's energy to cool infrastructural facilities compared to setting up a data centre site in a hot region such as Nigeria, Africa, where the weather temperature is up to 48 degrees. Therefore, for each watt of power consumed by computing resources, an additional 0.5–1 W is required for the cooling system (Okonor et al. 2019). Furthermore, the infrastructure's high energy consumption leads to substantial carbon dioxide (CO₂) emissions thereby, contributing to the greenhouse effect (Zhang, Cheng and Boutaba).

Following the research findings by international data corporation(IDC) which recorded that power usage for the data centre's cooling system costs more than \$30 billion annually and inline with other research survey by different authors. Gartner Group also predicts that with the continuous increasing trend in data centre energy consumption rate, the energy cost will reach 50% of IT budgets in the next few years (Gartner report and Okonor et al., 2019)

Infrastructural factors can be complex to handle, especially when there is a technical reason for citing the server: it has been proven that even when a server is idle, it still consumes about 70% of its peak power unless it is switched off. Which, according to the research conducted by Rightscale (Clement, 2017) confirmed that cloud consumers dissipate 45% of their total cloud consumption unduly. Henceforth, making the application running on a cloud platform with built-in microservices more efficient. Therefore, service providers and data centre vendors should study and understand the data centre components, the logic behind it infrastructural operations and the techniques that aid its functionality, enabling flexibility when deploying its application or replacing a part. This would allow cloud data centre operations on any phase (be it from vendor, service or service provider) to enjoy technical resilience, scalability, heterogeneity, composability and resource optimisation.

Now connecting existing facts to finding a lasting solution to the high-power consumption rate on the cloud data centre and minimising the high level of redundancy in the system operation. After a robust literature survey, this research work was motivated by curiosity to think of a different operating technique for managing and designing the data centre system because a persisting problem always needs a different approach. Therefore, this thesis explores using mobile agent technology, a paradigm that matches network complexity due to its flexibility in operation. A mobile agent can freely move around an active network system without disruption of its performance. Agent technology has been as far back as 1970 in existence but has never been used in cloud data centre energy efficiency-related field. The

performance of an agent operational technique in distributed system motivated the quest to explore this technological attribute in cloud DC regarding energy efficiency.

Moreover, the agent approach suitably adapts to emergency staggering task request cases, which is a plus for this technology when intermingling it with cloud mode of operation. Cloud service is solely an online-based system with a less predictive pattern of when and how tasks are requested urgently need a technology that understands its operation mode, which the agent approach blends well. Now bringing agent technology into the cloud operation scenario, an agent will be embedded in the network's microservices and the application components, enabling the agent to shutdown inactive components without affecting the performance of the cloud service operation.

1.3 Research Questions and Objectives

This work's research question came from the high energy consumption rate at the cloud data centre system trending for a decade with a minimal solution. Research has shown that data centres run 24/7 all year round with the typical power density of 538-2153 W/m² (Oro et al., 2015). Cloud Service provider try to keep an acceptable high degree of system performance reliability as promised to clients; this has led to an over-provision of the resource. In most cases, it means using enormously significant quantities of redundant power and thereby consuming high energy. Furthermore, cloud data centres operating with many microservice components make it more challenging to manage. The complexity in the cloud data centre's architectural design is vast then; imagine a combination of this impending problem with finding a balance operation with online business functionality. This act is so challenging and alarming when looking at its power usage rate during runtime, which led to the research curiosity question, "How best can the high energy consumption level in a cloud data centre be reduced?". This research work found an improved and intelligent solution to this vast research gap in cloud system using a mobile agent approach.

Theoretically, the research gap is to build on the already existing cloud data centre designs. However, there is no holistic approach to solving the energy efficiency problem in the cloud data centre. Therefore, this research aims to use intelligent mobile agent technology to schedule, monitor, shutdown, and balance workload in the cloud system.

This thesis focuses on using an intelligent mobile agent adaptive model to allocate and manage resources on the cloud data centre networks. This work emphasises finding a striking trade-off

between energy usage rate and the required reliability of the system QoS performance with minimal system downtime.

In respect to the research mentioned above question, the objective of doing this research work is to investigate ways to solve the research problem, which is explicatively defined below into a sub-problem, based on the system needs as follows:

- ❖ How to regulate and manage resource allocation for client requests under overloaded and oversubscribed conditions to avoid systems having more downtime and increasing performance degradation chances.
- ❖ How to identify redundant network components and minimise their function through a shutdown and migration mechanism. This is important because some components can be consuming power while idle, so knowing when to deactivate such components will be a significant saving.
- ❖ How to monitor cloud applications status through intelligent-adaptive decision during different workloads scenario. This enables the service provider to take certain intelligent actions such as shutdown switches or migrated a VM based on a task request pattern.
- ❖ When to trigger the agent movement into the cloud network for a better outcome. The mobile agent should only be activated in the network system when underutilised, overload or oversubscribe by a sudden sequential task congestion on the data centre component.
- ❖ How to minimise the energy consumption of cloud network system during runtime without violating the system reliability level. This is the anchor of this work and should provide a stable balance between saving energy during transition and QoS through implementing intelligent scheduling policies.

Now to be able to address these research problems mentioned above, we have set the following research plans:

- ❖ Design an intelligent system model that enables agent technology to access the cloud system's network and its application components.
- ❖ To formulate a mathematical model for energy usage that can help enhance and measure energy usage in data centre layers.
- ❖ Propose agent-based technique algorithms that determine when and how an agent is used in the cloud system.

- ❖ To design and develop a Hypothetical data centre, use the network components to test which level agent technology can reduce energy while maintaining a high level of service.
- ❖ Evaluate the System QoS based on its reliability level during agent activities
- ❖ Present this idea in a more non-technical way using semantic knowledge representation.

1.4 Methodology

This research project is essential because of its potential capability of sustaining cloud technology. This work aims to reduce the power usage rate during runtime in the cloud data centre while maintaining a high QoS level. In order to attain a minimal power usage rate during runtime, the framework in figure 1.3 has been adopted for this research work.

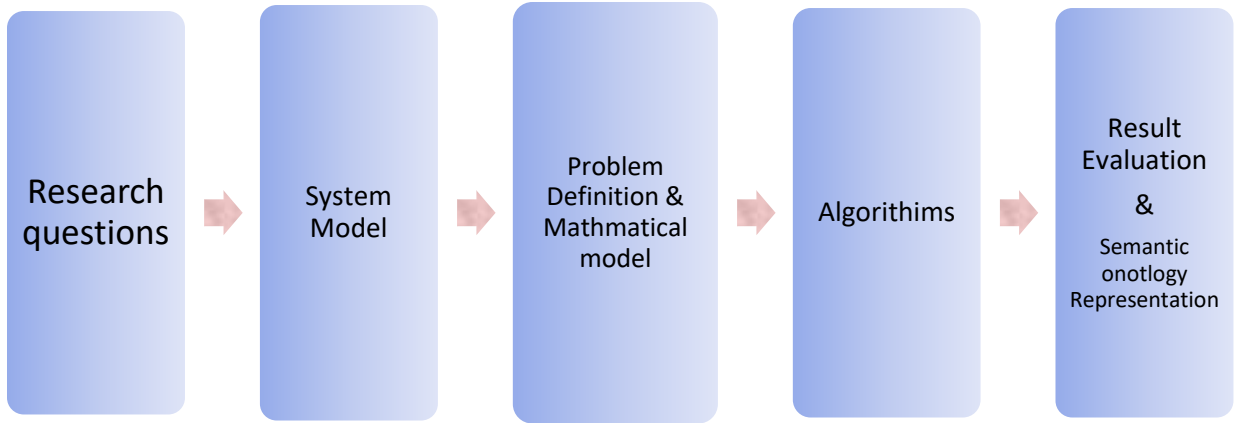


Figure 1.3: The methodology framework for this research work

The following description of the methodology has been broadly elaborated as shown below:

- ❖ Firstly, based on the research literature survey conducted during this research, which evaluated existing work on cloud networks, management tools, and vendors' opinions on cloud high energy consumption challenges. The evaluation from literature survey help narrow the problem around cloud data centre and opened up the research gap for this research, which lead to the developed research questions based on the gaps.
- ❖ Following this assumption that energy efficiency in the cloud data centre is a challenging one and can be leveraged by an intelligent agent technology, then define the system model, encompassing all the critical and non-critical components of the DC model for holistic action & clarity purposes.

-
- ❖ Define the problem Formulation based on each component power requirement and obtain a mathematical model for its energy usage based on its data centre application and network layers.
 - ❖ Propose an intelligent agent-driven algorithm that we then used to determine when to trigger the agent technology in the system and thereafter, calculate how much power it could save per system runtime.
 - ❖ Evaluation. This thesis's proposed method was evaluated using a simulator toolkit called Cloudsim and tested on a real-time system platform. Cloudsim obtained its workload dataset traces from PlanetLab. The two primary objective functions during the evaluation were energy consumption and QoS (based on system reliability level)

1.5 Research Contributions and Publications

The main novelty of this research work is based on using an intelligent mobile agent technology to solving complex energy efficiency challenges in the cloud data centre networks. The novelty is unique, and its contribution to both academia and industry experts are classified as follows:

1. A comprehensive, up-to-date review and taxonomy on energy efficiency in cloud system network and intelligent agent-based adaptive management of application resources.
2. Intelligent technique for scheduling resources on cloud application components based on:
 - ❖ A technique that considers the trade-off between deactivating a system during underutilisation and the effect on users based on SR.
 - ❖ A good understanding of the system model will enable a cloud infrastructural engineer to know which part of the system component to trigger an agent to better utilise the application component; this was achieved using an agent-driven algorithm with the target of saving energy.
 - ❖ Agent policies were made to enable some constraints considered during agent performance and activities on cloud applications based on energy saving deactivation approach.

-
- ❖ Daemon agent shutting down any inactive path in the network based on a certain threshold and triggering an alarm to other active agents.
3. Produced a resilient, intelligent-adaptive approach for managing cloud applications toward enabling green computing.
 - ❖ Assigning a time-based perspective model for multi-layer management and monitoring of workload resource scheduling of the cloud task.
 - ❖ Proposed an event-driven intelligent-adaptive technique for effective workload interaction and dispatching with optimum performance responses to users.
 4. Formulated a novel mathematical model based on the designed system model, considering the critical and non-critical components with higher power transmission level.
 5. Developed and designed a semantic knowledge representation of the proposed agent technology based on its link to cloud data centre network and application components, which gives the cloud infrastructural engineer more clarity on "where, how and when" to trigger the agent method into the system.

Publications –

Conferences

Ogechukwu M Okonor, Mo Adda, Alex Gegov David Sanders², Malik Jamal Musa Haddad², and Giles Tewkesbury² "Intelligent Approach to Minimising Power Consumption in Cloud-Based System Collecting Sensor data and Monitoring the Status of Powered Wheelchair" Intelligent System Conference, 2019.

Ogechukwu M.Okonor, Dr Mo Adda "Power Optimisation Model for Leveraging Cloud System", IEEE Conference, 2019.

Ogechukwu M Okonor, Mo Adda, Oliver Spear and Alex Gegov "Mobile Agent Based-Approach for Enhancing Energy Efficiency Cloud Data Center Network", IEEE ENERGYCON Conference, 2020

Journals

Ogechukwu M Okonor, Mo Adda and Alex Gegov, "Mobile Agent Based-Approach for Enhancing Energy Efficient Cloud Data Center Network", WAEAS Transition on Communications May 2020.

Ogechukwu M Okonor, Mo Adda and Alex Gegov, "Novel Power Management Technique for Efficient Cloud System Application Using Java Agent-Based Approach", submitted for publication.

1.6 Thesis Structure

This thesis is structured based on Figure 1.4 with the sole aim of sequential working through this categorial structure to achieving the set goal of minimising energy usage rate in cloud systems.

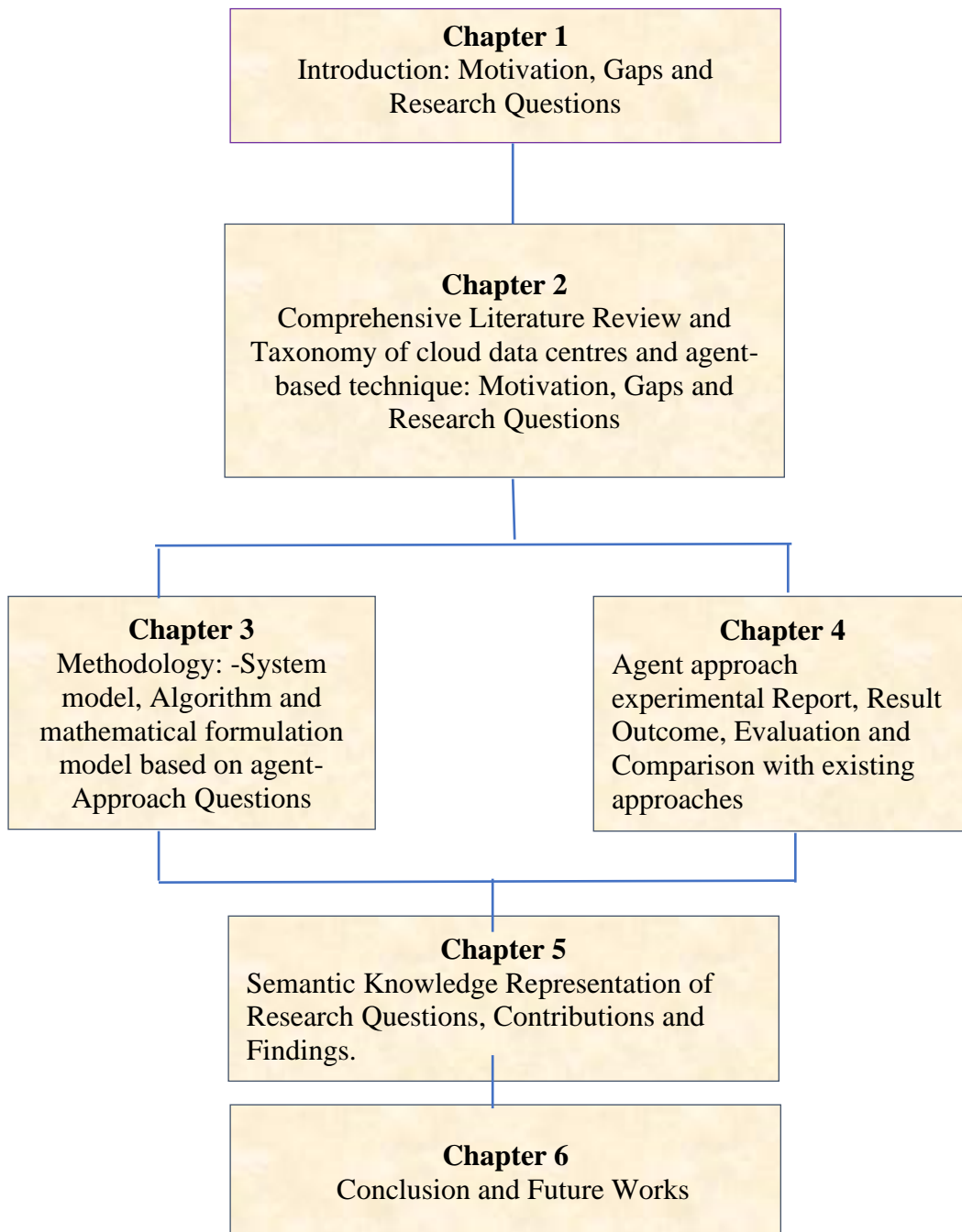


Figure 1.4: The Thesis Structure

Chapter2

Literature Review

2.1 Introduction

Cloud Computing (CC) has become the most emphasised and highly used Information and Communications Technology (ICT) paradigm. Cloud beneficiaries sometimes do not acknowledge their cloud innovation usability; they directly or indirectly underpins their daily search service through Internet activities. Cloud computing is now the trending name in communication due to its relevance to the world of computing and engineering. Cloud computing services promote rapid economic growth and empower both the underdeveloped and developing countries to acquire their desired services without limitation (Okonor, Adda, Gegov, 2019). Before the cloud innovation era, setting up a traditional data centre by an enterprise was an uphill task because it is capital intensive both for maintaining it and upfront infrastructural commitment involved.

In contrast to now, we are leveraging cloud service, where a computing resource can easily be leased based on requirement and the application deploy without stress. Cloud computing was initiated in 2006 by Google, Amazon, etc. (Gartner 2007 and Amazon online article, 2006). As many enterprises (both medium and small) strive to balance their operational cost and access advanced performance tools (such as proprietary software, platforms, and infrastructure), optimising the CC innovation services becomes inevitable because of its numerous benefits, which align with business needs. Cloud computing performances strategies were built on a utility-based business model, which enables its users to have ever-present, commodious, on-demand access to a shared pool of network configurable resources. The cloud environment is so robust and operates in such a dynamic manner that it creates an avenue for users to access their requested platforms based on their requirement through the virtual machine-assisted mechanism.

Cloud computing networks and applications are experiencing an unpredictable surge in service demand and workload due to cloud users' enormous requests. Imagine a situation where more than 100 million videos are watched via YouTube per day; Facebook has more than 400 million

active users and 3 billion photos uploaded per week: all powered by cloud technology, according to the research article of (Mittal et al. 2012).

Cloud system networks have been designed with a robust network complexity to adequately manage unexpected event occurrences such as flash crowds (which is request oversubscription when there is an outrageous increase in request size at a particular time. Occasionally, data centre hardware does unexpectedly fail, like the Sydney Amazon data centre failure in 2016, which was due to unprecedented adverse weather conditions (Online Article, 2016).

2.1.1 An explicit Knowledge of Adaptive management of Cloud system is a necessity

To be able to manage cloud services event occurrences, an adequate knowledge of the system network design and application deployment will aid effectiveness. For example, consider methods such as workload consolidation (Homs, Liu, Chaparro-Baquero, Bai, and Quan, 2017), data replication (Milani and Navimipour, 2016), auto-scaling (Lorido-Botran, Miguel-Alonso, and Lozano, 2014) and dynamic load balancing (Liu, Wierman, Low and Andrew, 2015) are only used in cloud systems as one of the remedy models to manage the unexpected event which occurs during runtime based on availability and request requirement. However, these unexpected events are often one-off and last for a short period before a solution is provided, hence economical unwise not to provide sufficient infrastructural equipment on the contrast. Sometimes, due to fear of failing users' expectations, providers tend to over-provide facilities to avoid downtime disruption. Therefore, finding a balance to what is sufficient or inadequate in cloud system by service providers is still an issue of concern. Choosing the adequate portion of system functionality can be so trivial because any mistake made during application runtime can cause a bottleneck on the system, which delays user request-response and degrades system performance. As a result, cost providers spend more money to maintain system components and betray user trust. Based on these points mentioned above, this study strongly argue that cloud resources' can be managed in an intelligently adaptive way for an utmost necessity and for sustainability purposes.

Thus, using an adaptive management approach to resource allocation and selection comes with numerous benefits. It brings about an improved Quality of Service (QoS) guarantee on cloud services, on the other hand, maintains a high level of service level agreement (SLA) to users. The QoS guarantee is very vital to estimating how cloud system performance over clients has been achieved.

2.2 Background

This section will briefly explain and then discuss background knowledge of cloud computing and cloud data centres. This section aims to give an introductory history of cloud computing, its benefits to business, its setback, and its operational model, explaining the connection between cloud services and the physical data centre. A basic understanding of data centre structural components explains data centre operational activities, and its hungry nature was discussed.

2.2.1 Cloud computing

The term "cloud" was used to metaphorise the Internet because a cloud form seems to be the best mode of description of a network on telephony schematics delivered through the Internet. Cloud computing started as far back as 1977 (Gary-Lee, 2012) (Grid5000) when it was used to demonstrate networks of computing equipment on ARPANET, thus gradually got proper attention when it was formally called "Cloud Computing" on an official internal document written by Compaq in 1996 (Online Article, 2002).

In 2006 cloud computing then became a paradigm that can never be overlooked when Amazon.com launched its Elastic Compute Cloud Product, which paved the way for other dominant vendors to adopt and build more robust cloud system networks. Cloud computing technology is so flexible to use with its pay-as-you-use business model, making the innovation so irresistible. Cloud computing models function in such a way that it positions a sizeable central server in different country regions and then distributes its resource from the servers on demand. The invention of smarter technologies has added to the relevance of cloud computing attributes. Industries and enterprises always seek a high-capacity network to run their businesses on low-cost computing rates with storage device availability. This trending business quest has led to a rapid surge in the cloud computing adoption rate. For instance, in 2019 Linux operating system was widely used and available for other operating systems platforms through cloud virtualisation and a service-oriented architecture. All these activities are powered by a cloud backbone called the data centres through their computation-intensive software programs. Despite the wave cloud computing is currently creating in the information technology world base on its numerous benefits, unfortunately, there are still impending set back facing cloud innovation such the security concerns, data compliance, governance, confusion in adoption

strategy and its energy efficiency issues. However, these setbacks are areas of problems, and research targets toward finding pragmatic solutions. Therefore, this research work focused on the energy efficiency challenge, which can indirectly provide lending steps to other cloud-related setbacks through various reviews.

2.2.2 Cloud Data centre

Data centres (DC) are the bedrock of cloud computing services. A typical data centre is comprised of a plethora of smaller networks that contain processing servers and storage devices. These networks are interconnected through a multitude of network switches to form an overarching DC architecture. A data centre structure can vary widely in its design; however, a typical data centre's contents consist of a core router, processing servers, layered network switches, and Virtual Machines (VM), each strategically positioned and connected to meet cloud user requirements. There will be no efficient cloud services without the data centres' serving power, which are located in several locations with supercomputing power and storage devices. Globally, data centres are spread over 300-4,500 square meters (Emerson 2008) and host ten thousand's server units. Ideally, a typical 500-square-meter data centre consumes about 27,048 kilowatt-hours (kWh) of power per day regardless of whether it is in an active or idle state. This amount of energy used by a single small data centre can supply electricity to 2,500 households in the UK (Emerson 2009). A data centre continuously runs computation-intensive software programs and supply to cloud clients through virtualisation. Observing from the high-level task traffic activities the data centre process per minute daily, it is no surprise that its power consumption rate keeps shooting up regularly (this underscores the amount of energy usage).

Two factors that cause the criticality of DC power usage rising challenges are; the increasing need for data computing, processing and storage (Agrawal and Sabharwal,2012). Secondly, the need to support a vast number of applications that run under DC platforms. Despite the high traffic demand level, some vendors still use their old traditional DC systems with less agility and resilience to cope with the pressure of new regular task inflow of traffic experience. Therefore, having explicit knowledge of the DC operation mechanism and its structural design

will help adapt the system's best management strategy for power usage. Related works on data centre energy efficiency have focused on different factors (from scheduling to allocation policies to operating system functionality and then to some parts of the hardware), which we will discuss subsequently. The authors [Tenna Mathew, Sekearan and Jose) based their research findings on traditional x86 enterprise server architecture compared to the modern data centre's sophisticated transaction. However, this finding from Tenna cannot be complete without the use of contemporary research because the then energy-efficient tools were not robustly designed to withstand the current pressure faced by the data centres at the moment. Many studies on power usage in data centres used the dynamic voltage frequency scaling (DVFS) management technique to predict the amount of power used at every given traffic transaction. All these factors and management techniques will be discussed in detail subsequently on related work subsystem. Figure 2.1 is the systematic representation of data centre taxonomy with its components with clarity on what makes it an energy-hungry system.

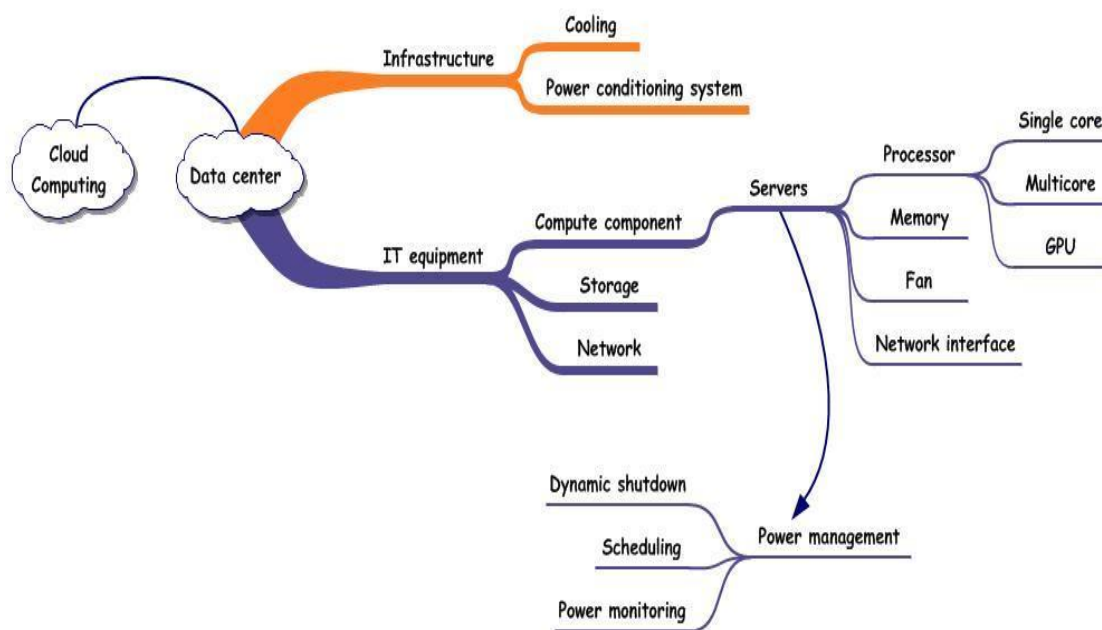


Figure 2.1: Taxonomy of a data centre

Figure 2.1 shows that the data centre system structure is classified into two main types: the computing resources (which is the IT equipment) and the Physical resources (which is the infrastructural part that handles cooling). It is essential to have an ideal knowledge that power management can be physical (data centre servers, cooling facilities, storage faculties and heat-dissipating equipment), infrastructural (based processing power or storage) and Applicational (based on computational-intensive software program DC runs).

According to (Enerdata 2014) published article, it has been acknowledged that since the invention of cloud computing service back from 1990 till this day, cloud data centre power consumption rate has doubled from 10k TWh up to 20k TWh globally. A future prediction has also forecasted that by 2040 cloud DC power consumption rate to rise to 40k TWh; it takes an increasing rise of 2.2% per year of required energy supply to sustain the cloud DC system. Previous articles also gave a total breakdown of the data centre structural component and the amount of power it consumes in percent. Figure 2.2 shows a typical data centre energy breakdown.

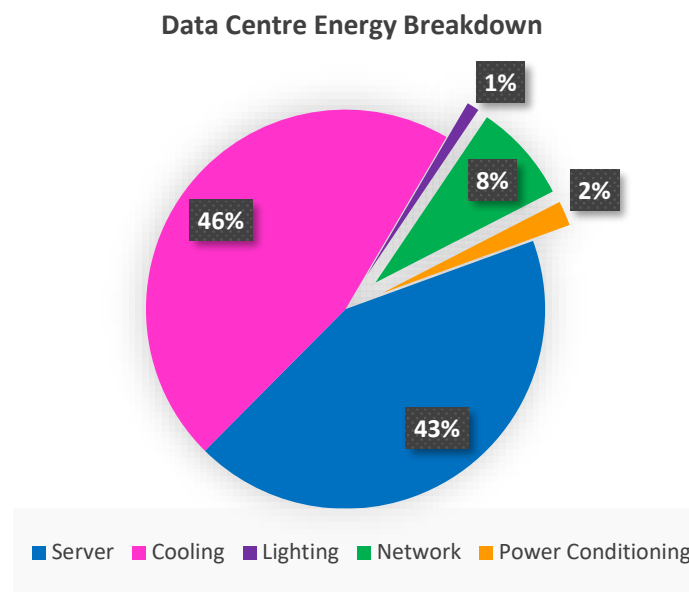


Figure 2.2: Systematic Chart representation of DC energy breakdown

Furthermore, relating these published figures, Google, a prominent cloud vendor, released a fact statement (Google annual online report, 2011)(Gartner report, 2011), on their report they used up to 2,675,898 mWh of power in 2011 with a large amount of it servicing their servers. In 2010, the US Environmental Protection Agency (EPA) estimated a 56% growth in cloud DC energy consumption level, which coincides with Fanara 2010 prediction. However, this forecast did not actualise due to a lower installed server base due to the 2008 financial crisis that paved the way for more virtualisation technology instead of investing in more physical servers (Koomey, 2011). Yet in the same 2011 (Belady et al.) estimated that the annual global data centre construction size for 2020 would cost \$78 billion, which is twice what it cost in 2010 to manage a data centre. The US data centre region alone consumed 100 billion kilowatt-hours (kWh) in 2015. The author (Li B. et al.) forecasted that the data centre's electricity

consumption in the US region would double to at least 150 billion (kWh) by 2022. Now watching the rate at which many firms suddenly migrated to cloud-based workforce mode due to the world pandemic, if the service providers do not use suitable control mechanism to reduce power usage, there may be an exponential surge in the energy consumption of the cloud data centre which corresponds with the authors (Nature, 2015) (Brown Report, 2007) prediction that in 2030 that the energy consumption rate of the DC will increase to 8000 terawatt-hours(TWh). Furthermore, a renowned Swedish researcher (Andrea, 2016) predicted that by 2025, the data centres would amount to the most significant ICT share of global electricity production at 33%, followed by smartphones (15%) (Okonor et al.,2020)

Based on what has been predicted, it is significantly vital a solvable technique for minimising DC high-energy usage level be invented to save this emerging technology, optimise the operational cost of running the DC and achieve green computing IT.

2.3 Article Selection Justification

In this section, the method for sourcing articles for this research work literature survey will be discussed as well as its obtained outcome.

2.3.1 Source of Obtained Articles

Literature was sourced through broad means such as oral and mainstream academic databases. This mainstream academic search engine gives us access to several repositories such as Institute of electrical and electronic engineers articles (IEEE Explore), Google Scholar, ACM Digital Library, Wiley Interscience, Cloud gateway, Gartner Reports, Springer, and Elsevier.

2.3.2 Search method

A two-phase search was carried out during this research work. Firstly, was the phase that has the keywords "cloud computing" and "Agent Technology". Using the keywords helped me to source existing research materials in this area more effectively. Several favourable results were found for cloud computing topics, but there were a limited number of articles on mobile agent technology based on cloud services. In the second phase, I established the importance of using intelligent agent search on networks management to have a factual basis for this research work. This work focus on static and mobile agent technology and how it could be used to manage

network performance. The observation from the behavioural patterns of a mobile agent in previous works was fascinating and inspired my curiosity to try agent phenomena on cloud networks service, focusing on energy usage matters.

2.3.3 Outcome

From my findings, it is evident that not much work has been done on mobile agent technology, and the always existing works on this area have generic concepts with no specific attention to the cloud environment or energy efficiency-related problems. For the cloud computing phase of this work, there were up to 80% of referenced research work published on both reputable conferences and journals, while the remaining 20% was the industry whitepapers based on their own hands-on organisational experience.

2.4 Review and Taxonomy of cloud power Management System.

In this section, this research study reviewed different articles and industry reports to ascertain related works on cloud data power management. Therefore, the subsections in this section will be detailing literature based on specific needs relevant to our work.

2.4.1 Taxonomy of cloud system power management system

Many research articles have been published on power management in the cloud system detailing different approaches used by other research teams to optimise its energy consumption rate and better cloud system performance. The power management approaches often used in cloud systems are categorised into two main forms (static and dynamic) with their subset. Figure 2.3 depicts a comprehensive taxonomy of power management technique.

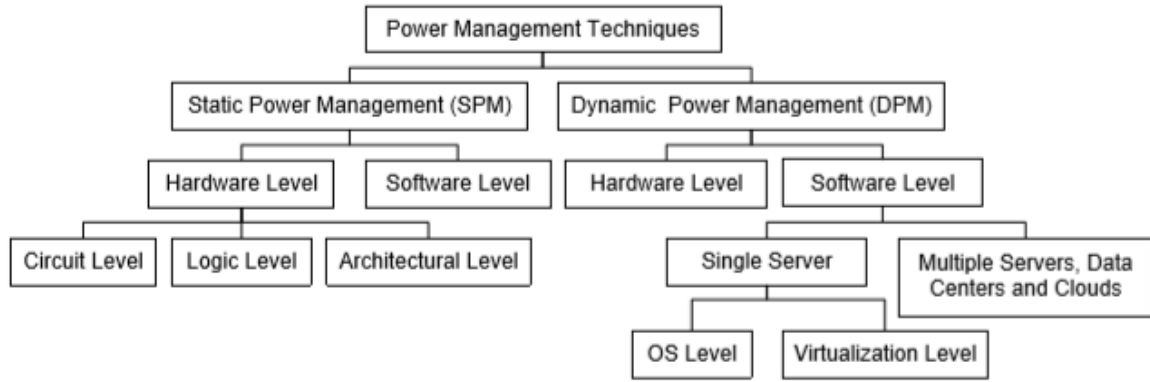


Figure 2.3: Taxonomy of power management technique

Leakage of current from the circuit, clock rate and circuit usage instances are the primary cause of static state power consumption. The static power is determined by the kind of transistor and process element its processes. Hence, the static power management (SPM) technique focused on managing the data centre's hardware components using optimisation methods that apply to the system architectural system logic, circuit, and design (Davadas and Malik, 1993). Circuit level optimisation concentrates on switch activities and how to reduce each logic and transistor-level power usage during the runtime by understanding how to apply intricate gate design and transistor-level. It is a highly complex task to handle because any mistake made leads to system breakdown and server degradation.

On the other hand, the dynamic state power consumption is mainly influenced by the hardware and software components such as the circuit, input/output (I/O), clock rate and other network connecting devices. In considering the power consumption on the dynamic state, the short-circuits current and switched network connectivity capacitance consumes up to 10% - 15% of the total power consumption. The DPM technique studied the system behaviour based on a given resource requirement during runtime, then developed strategies that easily adapt to the system design. Therefore, in DPM, the power consumption technique follows two set assumptions that enable adaptive management. These two assumptions bring about the variations of workload pattern during active runtime and, secondly, predicting the workload instance to a certain degree based on defined thresholds. The DPM approach gives the system the option to adjust to specific performance requirements dynamically. DPM is categorised by the level at which it adapts to both hardware and software components. From the hardware (DPM) comes the dynamic performance scaling (DPS), which birthed the dynamic voltage

frequency scaling (DVFS), and finally, the dynamic component deactivation (DCD) during inactive runtime.

In contrast, the user interface approach was used on the software side to manage the system power consumption level. This software DPM technique then brought about the introduction of Advance Power Management (APM) and the Advanced configuration and power interface (ACPI). Based on the DPM power management approach aspect, much research has been done in this area. However, researchers researched this area based on how they perceived the problem case leading to diverse results in this field. Still, power consumption is an outstanding worrying issue based on the literature.

DVFS was widely accepted as the underpinning technique for finding a solution to dynamic power consumption complexity due to its flexibility in adjusting to system performance. DVFS approach has highly been adopted for tracking power consumption problems to reduce the chances of the system being degraded (Pallipadi and Starikovskiy, 2006). Subsequently, DVFS has been extrapolated on a multiple server system providing coordinated performance scaling across the platform (Pinheiro, Kistler, and Rajamony, 2003).

2.4.2 Cloud Data Centre Hardware and Firmware Level

Having an explicit knowledge of the data centre hardware and firmware level brings Cloud engineers closer toward solving various data centre Infrastructural needs, which directly causes high energy usage. By definition, data centre hardware is the collective IT structural components that make up the cloud system networks such as the cables, routers, modems, firewalls, switches, hard drives, tapes drives, cooling tower servers, racks, desktops and power generators. A full knowledge requirement analysis of data centre hardware design is the first step toward achieving energy-efficient cloud data centre network. This will consider the current performance of the data centre hardware based on the present state and planning for future growth and potential risks that are inevitable through designing a more resilient and agile system. Creating a conceptual specification document for data centre design is very vital to proper management and monitoring of this resource. Already there has been an article on cloud data centre management and maintenance. Still, in this work, we will focus on hardware design and how its components affect the cloud system's energy efficiency side and the existing techniques for managing the data centre networks' hardware part. Avelar et al. 2012 defined

the supporting equipment within the data centre that aims for high power usage. The dynamic component deactivation (DCD) and dynamic performance scaling (DPS) are the two main categories of dynamic power management (DPM) techniques that have been applied to data centre hardware as accorded by (Beloglazov et al. 2012). Figure 2.4 shows a complete picture of the existing methods regarding power management, which I will explain in detail.

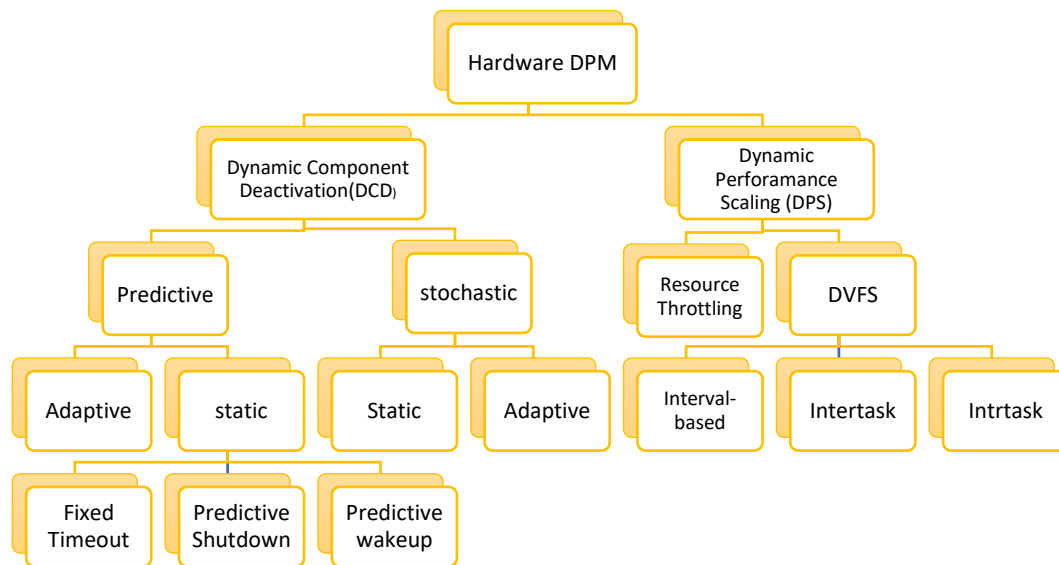


Figure 2.4: Comprehensive DPM techniques.

The DCD method was built on clock gating ideology where components are disabled and enabled based on its activeness. According to the article written by (Benin, Bogliolo, and Micheli, 2000), this approach neglects the power and performance overhead, thereby causing a low-power state, which leads to additional power consumption and delay caused by re-initialisation of the component. The problem of inadequate knowledge to reinforce shutdown and re-initialise the component into being fully active at a given time without degrading the system performance is trivial (Albers, 2010). The predictive technique is undoubtedly based on the correlation between the past history of the system performance, the present, and the future. So, this kind of technique's success depends on the past and the future, but in an online system with no adequate pattern of the sequence, incoming task follows when compared to what it was in the past. Cloud activities can be labelled as a system that is more event-driven, for instance, the march 2019 pandemic that suddenly brought about lockdown, which led to 85% of people working remotely from home

The static approach made use of a particular set threshold. The simplest of this threshold was called the fixed timeout, where a certain length of time is map out for a system to be active;

otherwise, it is considered idle and then shut down. The weakness found in the fixed timeout technique of static mode was then addressed by researchers under the predictive shutdown and predictive wakeup, where the length of the next idle time of the system was predicted based on past activities. Based on the research conducted by (Srivastava, Chandrakasan, and Brodersen, 1996) (Hwang and Wu, 2000) showed that a task request transition that occurs on the system has been assumed based on previous task activities' history and performance even before the actual users send their requests.

2.4.3 Energy Efficiency Management Level

Energy efficiency simply can be defined as a mode of reducing power the power usage rate of a given service. Energy efficiency has been acknowledged by (EU2011) as the most cost-effective way of sustaining the green climate goal and achieving a long-term low energy usage rate in cloud service. In 2010, (Garg, Yeo., Anandasivam, and Buyya) showed that ICT accounts for 2% of the greenhouse effect annually in their documented research. Energy inefficiency could be attributed to the increasing demand in traffic due to the introduction of high bandwidth requirements and Quality of Service (QoS) measures in new applications as customer adoption rate increases. Some industry giants and ICT sustainability experts came to seek ways to regulate and create policies that will make cloud service green and sustainable. These moves led to the development of standardised policies and methods such as Green computing impact organisation (GCIO), Climate saver computing initiative (CSCI) and the Green Grid. Efficiency in resource utilisation was first tried on battery feed mobile devices. The success achieved by improving battery life in an energy-efficient way bought about the venture into improving every computing device to have an optimal energy efficiency level through different means. The idea of achieving green computing has made energy-efficiency related problems a hot topic of discussion by industry experts and academic research groups. However, the more they deliberate on the way forward to achieving an optimal energy efficiency value in the cloud system, the more it scales on a limbo mode because of certain misconceptions surrounding cloud connectivity which will be discussed in detail. For instance, Fanara, 2007 describes the cloud system as a complex system with many computing, networking, and management components, making its efficient solution based on only wired distribution. Figure 2.4 shows a concise energy consumption of the cloud computing system's software and hardware level.

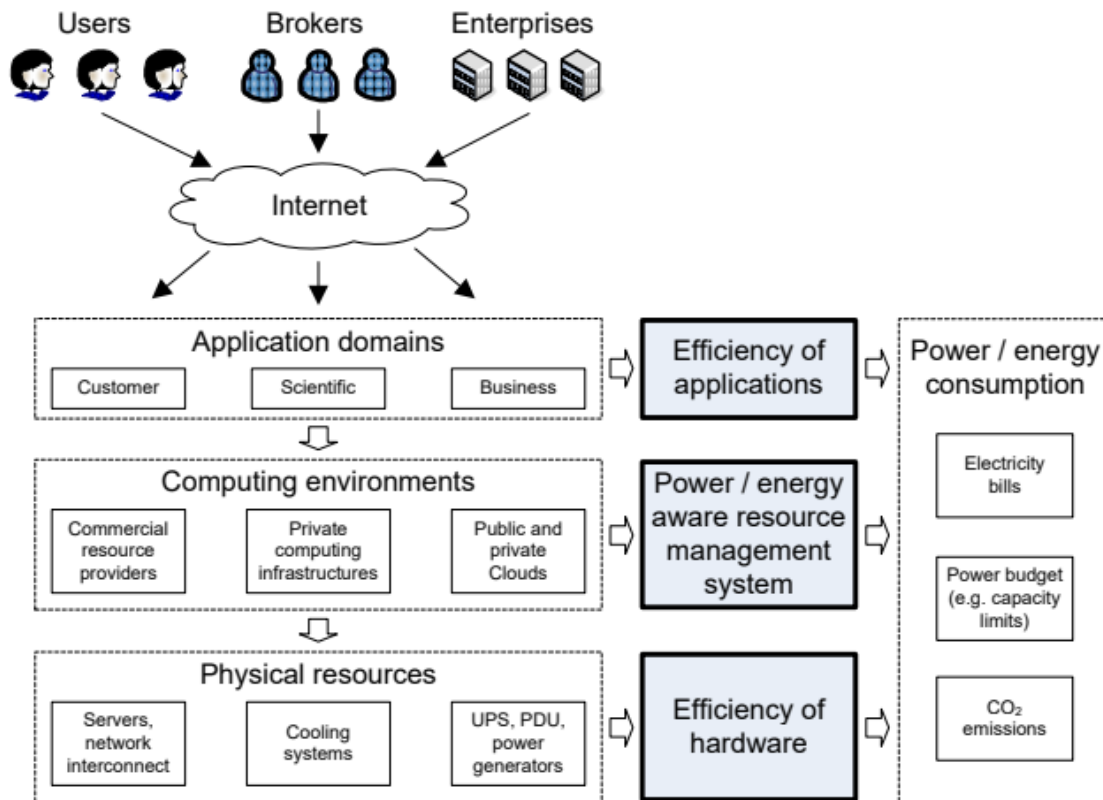


Figure 2.5: power consumption at the various levels of the cloud computing system

It is worth mentioning that cloud services use both wired and wireless design for their operational functionality because cloud services provide their services via the web application. Hence web services are also developed in such a way it needs constant continuous updating of the applications. A frequently logical manner of updating is necessary since web service has a multi-layer architectural design that is monolithic, which means that detailed attention should be given to every component involved to understand its functionality. Thus, making finding an efficient operational approach to leveraging energy usage issue in cloud system tricky. Therefore, it is imperative to identify and understand factors that cause energy inefficiency in a cloud system to present a workable solution for efficiency based on the criticality of components. Figure 2.6 shows the cloud system energy flow with its critical point of interest based on research.

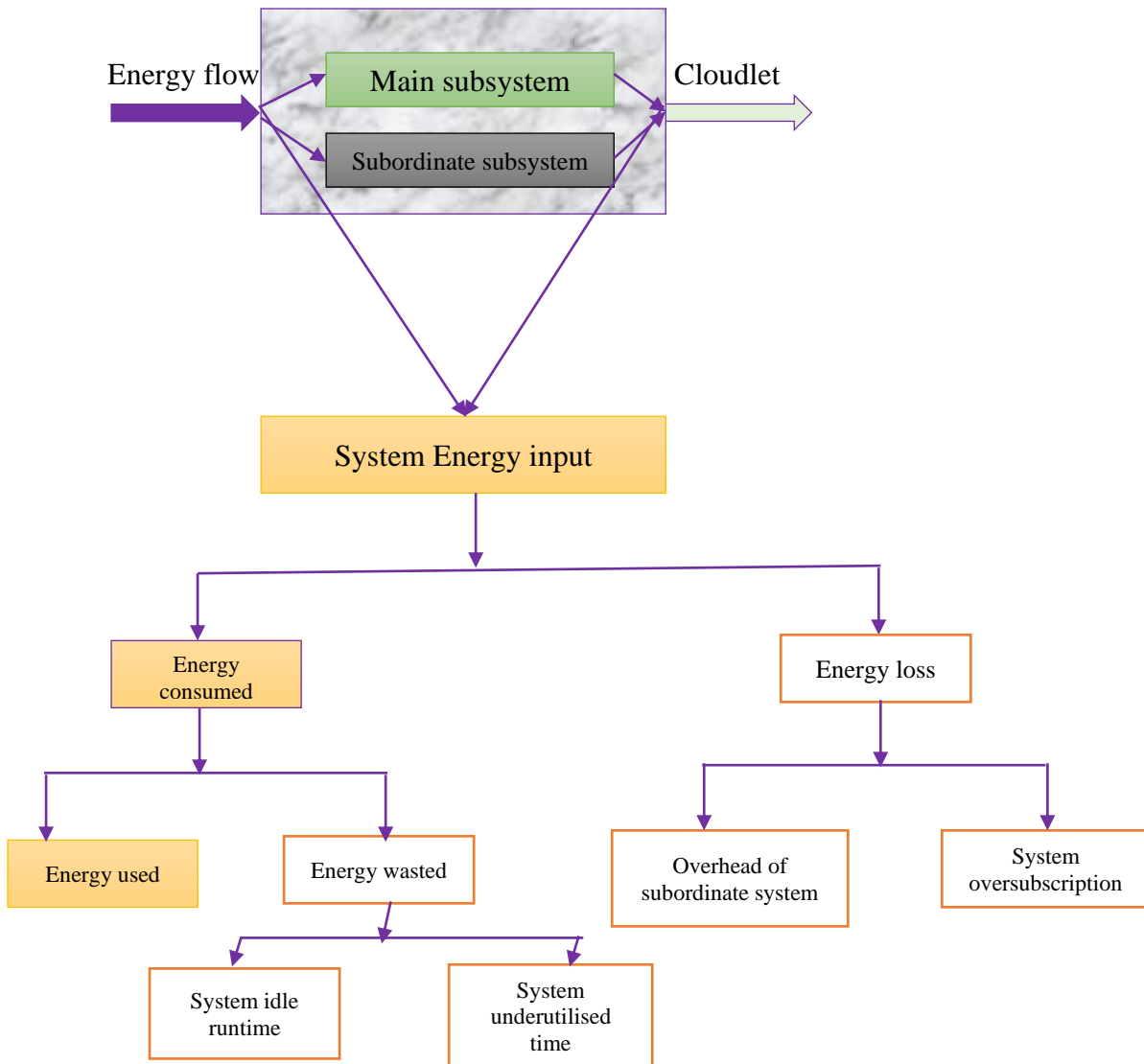


Figure 2.6: Precise critical points of areas to look out for when working on a cloud system

Observing from the system energy input in figure 2.6, which shows that the cloud system consumes energy during system runtime and, indirectly lost energy too during network interaction phase. However, according to the research conducted by (Ashrea,2012), the group validated that not all energy is used for networking since communication equipment shows the highest heat load footprint, which accounts for energy lost in the system but never used by the system. Therefore, research needs to pay adequate attention to both the used and loss energy factors to produce an optimal solution.

For proper management and monitoring in cloud service, which underpins achieving an energy-efficient goal, cloud operating engineers should have explicit knowledge of the amount of energy used by different networking component and microservice inside the cloud data

centre. Observe the wastage rate per time-based on the system been underutilised and idle. Then calculate the consequences of moving subordinate subsystem (such as cooling fan and rack partitioning) around for a more efficient outcome. For instance, when a user sends a task, it is advised that vendors find the best possible solution around its regional availability zone to high latency occurrences while transporting and conversing the task. This, in turn, increases efficiency. Energy efficiency in cloud system can also be encouraged by using and constructing more efficient hardware and software components, such as using more durable power-aware energy supply units for server operations. Mudge and Holzle 2010 suggested using slower server components like wimpy CPU cores to achieve energy-efficient; however, their idea caused performance degradation. Beloglazov et al. 2011 surveyed cloud system energy efficiency by defining a power consumption model for static and dynamic network system; however, their work focused only on idle runtime, which is just a fragment of the cloud system's inefficiency. Avelar et al. 2012 then got a different approach to their literature survey by classifying energy used by ICT and other auxiliary subsystems in a cloud system, measuring their energy loss rate. Related literature reviews on energy and power efficiency in the cloud system will be presented in a tabular manner in Table 2.2. This will consist of both the wired and wireless works of literature on this area. This research work considers factors that cause energy loss and components that consume most energy during the run time, which then gave this study the edges it has using an agent system to leverage the impending challenges.

Table 2.2: Related work on energy efficiency in a wired and wireless cloud system

Research area	Limitations	References
Scaling of energy consumption with load	The focus was only on load-balancing	(Gupa and Singh, 2007)(Nedevschi et al., 2008)(Grebennikvo and Bulja, 2012)(Claussen et al., 2010)
Energy efficiency in a general context	Did not specify any area of work concertation	(Earth, 2011)(GreenTouch, 2012)(Hoelzle and Barroso, 2013)(Yogesh s and Bahman Javadi, 2016)
Server cooling	Works were just limited to how to cool the servers facilities only	(Snyder et al., 2006)(Haywood et al., 2012)(Ayoub et al., 2012)
Energy efficiency cooling	Work was only based on heat flow and thermodynamic model	(Tang. Q and Gupta S.k.s, 2006) (Vasic N., Scherer T and Schott, 2010)
Design and development	Focused on system design and how to develop newer versions	(Saxe, 2010)(Smith and Sommerville 2010)(Argrawal and Sabharwal, 2012)
Processor architecture and design	Work was done specifically on managing processors efficient	(Dreslinski et al. 2009)(Zer et al 2010)(Mudge and Holzle 2010)[Berge et al. 2012)(Vor dem Berges et al., 2014)
Application Platforms	Their literature survey centred on how to make the application platform of the cloud system function efficiently	(Pianese et al., 2010) (Smets- Solanes et al., 2012)
Cache management	Attention was given to cache properties and purpose management strategy for best energy usage rate	(Dreslinski et al. 2008) (De Langen and Jurlink 2009) (Sundararajan et al. 2011) (Kim et al. 2013) (Tavarageri and Sadayappan, 2013)

In this section of energy efficiency management level, it is ideal to explain how the power management mechanism of cloud system works and what's contributes to its energy usage rate. Ideally, the electric current flows through the active network electric charges to produce power, which is when a system performs a given work. It can also be called energy if we sum the total amount of work done over a given time. Electric current is measured in amperes (Amps), the number of electric currents transferred by a circuit per second. Power is measured in watt, while energy is measured in watt-hour. Thus, converting work done at each watt's rate when one ampere is transferred becomes one volt with measurement in kilowatt. Therefore, energy can be defined as the capacity to done work, while power is the rate at which the work was done. The simple equation for power and energy is shown below for clarity purpose:

$$E = W \quad (1)$$

$$P = \frac{E}{T} \quad (2)$$

where

T is the time (which is the duration of the work to be done)

W is the work done, P is the power and E in the energy

It is worth noting that power and energy are different concepts but with interweaving meaning which is why it has been used interchangeably in this research. Thus, while power is the amount of energy used over time, it will also be relevant to acknowledge that power consumption of a system's power consumption can sometimes not reduce its energy usage. For instance, a system's power consumption level can be reduced by twisting the CPU activities by lowering its performance. Simultaneously based on the system's changes to achieve lower CPU energy consumption, it will sometimes result in programs taking a long time to complete its execution. Research has also shown that any effort to reduce peak power consumption level on cloud data centre always directly reduces the cost allocated to electricity expense and infrastructure provisioning costs, such as costs associated with UPS capacities, PDU, power generators, cooling system, and power distribution equipment.

2.4.4 Cloud Data Centre Energy Management based on Network Level

In view of optimising cloud computing resources' energy consumption while maintaining reliable system performance, it is crucial to understand the cloud data centre network level and its contribution to the cloud system's inefficiency. The data centre's network level is the enabling components for communication on a cloud system network because it allows connection between IT computing fragments and the storage resources. These networks IT computing components consist of servers, switches, routers, power distribution infrastructures (such as links, aggregation elements and nodes). Kliazovich, and Bouvry, 2010 acknowledged that the data centre's networking level components are intensively power-hungry equipment types. (Brown, R et al., 2008) accorded that the cost of operating and managing the data centre network level has tripled due to the increase in demand for computing capacity. Based on the related work done by Shang l et al. 2009 figures showed that the network-level components consume about two-thirds of the total IT energy, which is alarming. An American researcher's traffic prediction (Kilper et al. 2011) forecasted that North American until 2020 will continually have an exponential increase in their network-level traffic. The network-level power consumption consists of three primary systems, which are the fibre connection within the data centre structure, the fixed network connectivity between data centres from different availability zones in the same regions and the end-user networks level that offers the wireless last hop connection to end-users that access cloud service through (smartphones and tablets). Fanara et al., 2007 reported in one of their articles that the network level contributed 5% to the total energy consumption of the data centre structure annually. In 2010 another researcher (Abts et al.) also observed that networking level components power used accounts for approximately 20% of the total power when the servers are utilised at 100% and also skyrocket to 50% if the server utilisation reduces by just 15%.

The data centre network-level experience multi-level complexity, and this is due to the level of system performance it does. For instance, when a user wants to connect to another data centre in the same region but on a different availability zone, another application is launched to enable the network to migrate data between the various data centres. According to Wang et al. 2014, this transaction cost more energy usage rate during the communication process. The high energy consumption on the cloud data centre's network level has led to a green hardware proposal through rate adaptation (Nedevschi, Popa, and Ratnasamy, 2008). Rate adaptation is explicitly defined as the hardware mode for system energy saving achieved by operating a device at a lower voltage (i.e. Dynamic Voltage Scaling - DVS) or frequency (i.e. Dynamic

Frequency Scaling – DFS). Nedevschi et al. also confirmed in their article that an ethernet link dissipates 2-4W when operating at 100Mbps- 1Gbps. However, it can dissipate 10-20W operating at 10Gbps, which has a drastic devastating effect on the data centre's energy-saving algorithm.

The research work conducted by (Lorch and Smith, 2001) proposed dynamic voltage frequency scaling (DVFS) to minimise servers' power usage level. However, since idle servers consume about two-third of the peak and load, the efficiency of DVFS is limited. In 2009, (Liu J et al.) observed that the average workload performance by typical data centres is about 30% of the resources and this cause a high level of energy inefficiency. Furthermore, the authors (Hlavacs, Costa, Pierson, 2010) displayed their research finding that showed that the IP router linecard consume up to 43% of the total energy used to serve and actively power the router. The rest of the network level cloud data centre past literature will be presented in Table 2.3.

Table 2.3: Tabulate Literature review on the cloud network domain

Research area	Limitations	References
DVFS and alternatives	For the DVFS technique, attention was on how to minimise power usage on only the computing processor, which is not the only contributing factor to the cloud data centre's high energy consumption rate. I, therefore, recommend testing this method on other factors too.	(Megalingnam et al. 2009) (Anghel et al. 2011) (Chetsa et al. 2012) (Kahng et al. 2013) (Hankendi et al. 2013)
Network architectures	Their approach discussed the entire network communication and how the design of a network affects its flow; however, their work did not consider the network's energy-efficient aspect and on the cloud.	(Claussen et al. 2009) (Ravavi and Claussen 2012) (Yang and Marzetta, 2013)
Traffic Engineering and Routing	Their work was not on a distributed system and therefore cannot be implemented in cloud networks	(Zhang et al. 2010) (Vasic and Kostic 2010) (Vasic et al. 2011) (Cianfrani et al. 2012)
Network-level design	The work focus on network-level energy saving; however, it considered a few of the network level components.	Gyarmati L et al. 2010
Network-level Rate adaption	Proposed the optimal R.A. and practical R.A. for reducing energy usage rate in the cloud data centre; however, performance expectation was not satisfactory based on SLA.	(Nedevschi, s. et al. 2008)

2.4.5 Cloud Data Centre Energy Management based on Application Level

The data centre level is one of the essential system levels that determine resource performance and resource allocation, which is vital for understanding how to minimise power consumption in cloud systems networks. Cloud data centre application level is purely software-based. Figure 2.7 below shows the classes entail in application level and their subsystem.

Each compartment has a unique feature that contributes immensely to cloud service efficiency. Therefore, studies need to pay adequate attention to these areas for an optimal insight into how best to manage and leverage application-level services; thus, they still have an energy-efficient cloud system.

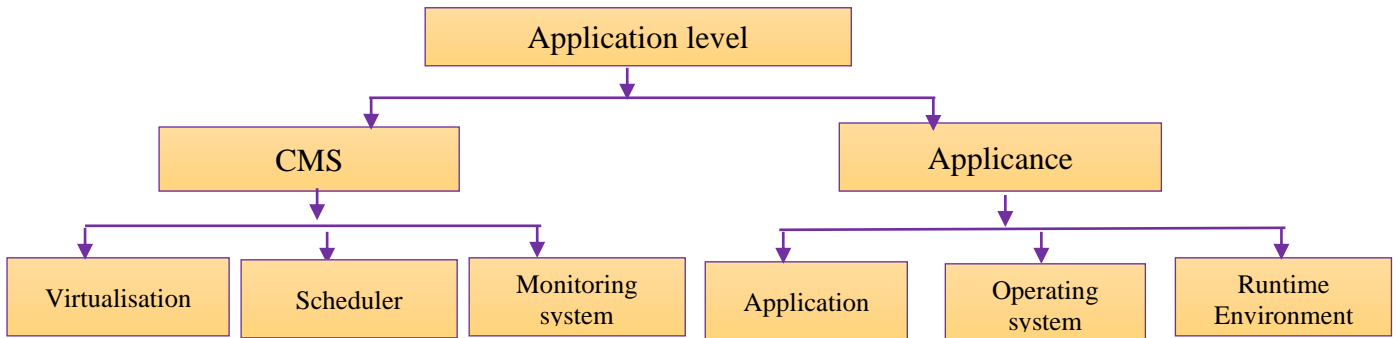


Figure 2.7: The application-level was divided into the cloud management system (CMS) and the appliances sections

Under the CMS, we then had the subclasses such as the virtualisation, scheduler, and monitoring system software. On the other hand, we have the subclasses like system runtime, operating system, and other appliance side applications. Therefore, related works done in this area will be discussed sectioned based on the cloud system's classified CMS and appliance application level. Emerson 2010 suggested that cloud system managing and monitoring should be very critical to both the industry and academic experts. He claimed that adequate knowledge of these areas would increase energy efficiency, and vendors easily identify the area of concern in the data centre facility. The CMS manages the physical and virtual machines within the cloud system (e.g. OpenStacks (OpenStack 2012 and Xen hypervisor (Citrix 2012))). Virtualisation is a better technique for managing physical resources in the cloud. It works in such a way that it brings flexible moving services between servers. Virtualisation also enables multiple VMs to share different application multiplexing from a single server. Uhlig et al. 2005 recorded that virtualisation technology offers an extra infrastructure layer on top servers that can deploy multiple more VMs functionality (Mastelic and Brandic 2013) (Jin et al. 2013) details a more effective way of using virtualisation technology. However, it acknowledged that virtualisation could consume resources and consume energy through the hypervisor if not adequately allocated and managed. From Jin et al. report, it could be seen that a hypervisor on full virtualisation mode(e.g. kW (Rathat, 2012)) had much higher overhead consumption of

11.6% than the one on paravirtualisation mode with just 0.47%(e.g. Xen (Citrix 2012)). Now understanding that virtualisation technology can be suitable for cloud system management and monitoring when used adequately, and in contrast, be very toxic if not implemented correctly. Zhang et al. propose VMs that can self-adapt to their available resource and allocate the available system's demanded resources. Borretto et al. 2012 suggested VM be configured in such a way that the middleware adapts to the VM resources demands to its needs. Borretto also considered the time it took for the VM configuration to model to adapt to change from one state of the physical machine to another (on and off). Cardoso et al. 2009 detailed several problems of handling CMS parameters for VM resources sharing. They specifically proposed an approach for solving power-efficient issues with VMs virtualisation in a heterogeneous environment. Their method leveraged the min-max and distributed the parameters of VMM that show-cased the minimum, maximum and proportion of the CPU allocated VMs sharing the same resources. Their approach feels good to implement but can only benefit enterprise owners, not the commercial cloud environment that works with strict SLA users. Kusic et al. 2009 suggested that the power management issues associated with virtualisation in the heterogeneous environment are a sequential optimisation problem that can be solved by using limited Lookahead Control (LLC). Their main focus was to maximise the resource providers benefits by minimising both the level of power consumption and SLA violation of the system. They used the Kalman filter to estimate future requests and predict the system performance's future state and the essential relocation process. However, their proposed method needs a lot of simulation-based learning to apply a specific adjustment to the application level. The approach also took the optimisation controller 30 minute to execute only 15 nodes due to the application's complexity, which is not sustainable in a cloud system with complex networking arrangement. Nathuji and Schwam 2007 proposed the hypervisor's virtualPower context, a power extension associated with the VMs with the software CPU power state. This VirtualPower states allows hardware and software to be coordinated by using the best power mode and DVFS.

Based on the VM placement context, Belgoglazon and Buyya 2010 proposed a VM to server mapping architecture. Their work applied an adapted model of the best-fit decreasing heuristic, a sub-member of the bin packing family. However, the heuristic family's solution cannot be optimal because it doesn't represent the heterogeneous family, which is the main property of cloud system networks. Hence sorting criteria are also required for the servers to decide which bins are to be filled out first. Verma et al. 2008 formulated a problem power-aware dynamic

placement system that was applied on a virtualised heterogeneous system as continuous optimisation (that is at a particular timeframe, the placement of VMs is optimised to minimise performance); however, the proposed algorithm fails to handle the required reliability test to validate the performance level. Gandhi et al. 2009 proposed allocating an available power budget among available servers in a virtualised heterogeneous server farm while minimising the request's mean response time. A theoretical queuing model was used to investigate the different factors' effect on mean response time variation. In contrary to Gandhi find, in 2012, Borgetto et al. proposed sorting servers and VMs out for mapping process using the best-fit and first-fit algorithm. However, their algorithm was centric on some components and did not represent a holistic method for solving energy-related issues in a cloud system. Other literature related works will be tabulated below in Table 2.4.

Table 2.4: A tabulated review of virtual machines related works

VM related works	Limitations	References
VM placement	The work done by the author focused on just placement with some constraints that did not apply to real-time	(Beloglazov and Buyya 2010) (Barbagallo et al 2010) (Kamitsos et al 2010) (Hoyer et al 2010) (Borgetto et al. 2012)
VM migration and consolidation	Their work was more solid and offered workable solutions; unfortunately, more research needs to be done to bring their concept into full functionality	(Choi et al. 2008) (Hermenier et al 2009) (Kumar et al. 2009) (Feller et al 2010) (Liu et al 2011a) (Cioara et al 2011)
VM reconfiguration and hardware management	The work showed improved progress to an existing research finding; however, it still had some performance overhead which is a concern.	(Zhang et al. 2005) (Nathuji and Schwan 2007) (Stoess et al. 2007) (Cardossa et al. 2009) (Kim et al 2011)
VM scheduling	This work proposed a scheduling algorithm; however, the technique needed to be tested on other platforms to validate its performance before presenting an ambiguous assumption.	(Burge et al. 2007) (Berral et al.2010) (Beloglzov et al. 2012) (Potverini et al. 2014)

2.4.6 Cloud service Scheduling Mechanisms

A cloud system scheduling process starts when a user submits its requested job to the cloud information registry. The job then gets to the data centre networks, where the cloud broker classifies the jobs based on the SLA and requested services. Afterwards, each job is assigned to one of the available servers based on defined constraints. In turn, the servers perform the requested job and then respond to the broker, which then transmits the job to the user as a result. Scheduling is one factor that seriously affects the power consumption level in cloud service. It does determine the cloud system network performance and reliability. In addition to scheduling being the critical tool to system performance, it also has other functionalities such as load balancing, complying with QoS, deadline constraints and maintaining a specified SLA

(Beloglzov et al. 2012). Many approaches to scheduling in the cloud system have been discussed in previous works by different researchers based on their chosen case study. This work will focus on scheduling problems with energy efficiency in the cloud system service in view. Scheduling procedures are carried out on both the hardware and software parameters. According to the related work done by authors (A. Sharm, Y. Yao, and L. Huang)(Okonor et al.2020) scheduling, procedures have been classified into three types, name ly: resource scheduling, task scheduling and workflow scheduling and each has subsets attached to it as shown in Figure 2.8. The authors (Ramezani F., Lu J., Hussain F) described issues associated with the cloud system's scheduling process as a combinatorial optimisation problem with complicated details. In this thesis, we considered literature in relation to task and resource scheduling due to its impact on the outcome of our field of study

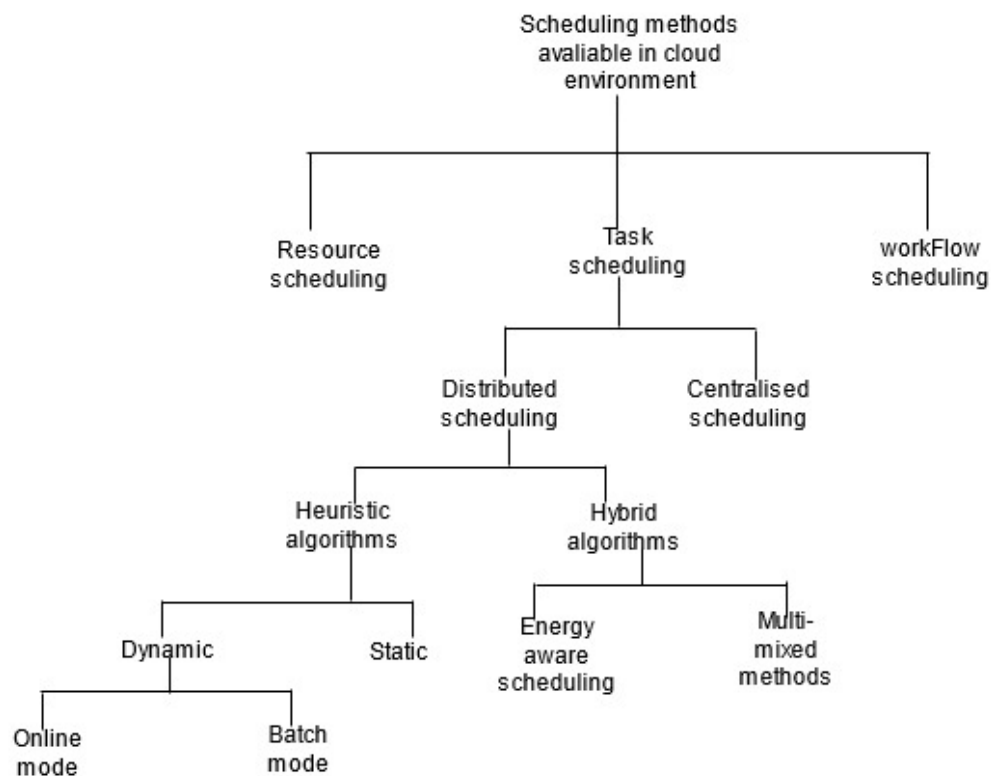


Figure 2.8: Related works on Various Task scheduling

Task and workflow scheduling are often called NP-hard problems, which can be biased when evaluating from complex nature of cloud computing connectivity. Numerous scheduling algorithms such as FCFS, Round-Robin, min–min algorithm and max-min algorithm have been proposed in the literature to solve them (Cheng, Zhou, Lama, Ji and Jigang, 2016). In

comparison, resource scheduling was considered a bin packing problem. Resource scheduling uses the scheduling mapping process on the virtual machine embedded on servers to allocate machine tasks in a specified order. Task scheduling is often seen as a more accessible form of scheduler than resource scheduling because the scheduler only deals with a pool of jobs with no form of interdependency and executes them in an arbitrary order; while in contrast, the resource scheduler is more complicated because it consists of a set of dependent jobs communicating with each other to ensure availability (Beloglzov et al. 2016) and the scheduler maps the jobs that go to each VM by considering their dependencies and communication with their physical server. (Tracy et al., Weicheng et al., Ning et al. and Xiaonian et al.) did a good literature collection on task scheduling. Most of the research concentrated on the heuristic scheduling method based on this effectiveness (Okonor et al., 2019). The author (Beloglzov et al. 2012) labelled the heuristic approach as the best approach for scheduling because it takes less time to complete a task, not considering some vital aspect of scheduling in an online system. The heuristic method can either make a static or dynamic turn as it applies to the cloud system. The heuristic method uses more knowledge base information to work; like Minimum Execution Time and Minimum completion Time is one of the subsets of heuristic strategies that assigns jobs on a certain machine with the assumption the set machine will take less execution time.

In contrast, Min-Min and Max-Min, another subset of heuristic methods, select the smallest job first from all the available jobs on queue to execute, presuming it will be fastest. However, heuristic the main limitation with this approach is that the method does not consider resource availability before scheduling, thereby causing load unbalancing and performance degradation. The rest of the literature survey on task scheduler algorithms has been shown in table 2.5 below for clarity purposes.

Table 2.5: Literature survey on task scheduler algorithms

Scheduling Approach	On view Parameters	Benefits	Limitations
First Come First Served	Arrival time	Easy to implement	Doesn't take all most vital parameters
Genetic Algorithm	Makespan, Efficiency,	Better system performance	Long execution time
Min-Min, Max-Min	Makespan, expected completion time	Better outcome based on makespan compare to other algorithms	QoS not taking into account and poor load balancing,
Switching Algorithm	Load balancing, performance	Better makespan	Time-consuming and higher cost of maintenance
Simulated Annealing	Optimisation, Makespan	Better makespan	System QoS not met based on the heterogeneous setting.
DENs	Traffic load balancing, congestion, energy consumption	Job consolidation was done to save power	Only jobs data-intensive application was concerned with no insight on computing demands
A priority-based job scheduling algorithm	Priority was given on specific jobs, completion time.	Design on a defined decision-making process which got priority prefers	The proposed method on be considered for improvement for better performance.
K- per cent Best	Performance, Makespan	Chose the most available machine	Resources were selected based on only the completion time of a machine.
Cost-Based Multi QoS Based DLT scheduling	Load balancing, QoS. Makespan, Cost, Performance	It gives an overall system performance	Machine failure, Communication overheads and performance degradation

Virendra et al. and Jose et al. have done an intensive literature review on resource scheduling procedure. Resource scheduling comprises three main functionalities: mapping resources, execution of resources, and monitoring of resource. This can be very difficult to achieve because of the online cloud system's complexity with no operation pattern. The main disadvantages of resource scheduling are that it comes with high uncertainty, dispersion and complication that follows heterogeneity of online resource. Figure 2.9 below demonstrates the content of the resource scheduler and how each phase of the content operates. Figure 2.9 showed that phase 2 of the resource scheduling comes into place after resource provision has been completed.

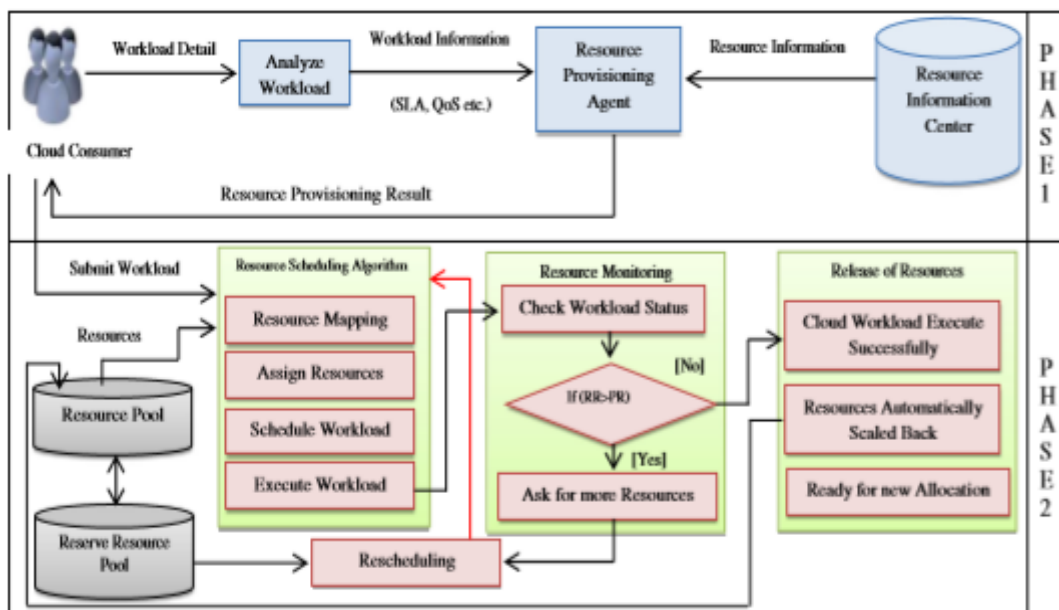


Figure 2.9: Resource scheduling in cloud system

2.5 Intelligent Mobile Agent Approach

This section will provide an understandable knowledge of an intelligent agent phenomenon through a detailed discussion of mobile agent history, attributes, intelligent nature of agent technology and mobile agent evolving in computing networks and then cloud.

2.5.1 Overview of Mobile Agent Approach

Agent technology is a smart computational entity that can act on its own or on behalf of other entities to achieve a set goal. Remote Procedure Call (RPC) is the engine that powers agent technology. The RCP is the process whereby a computing program causes a specific method

to execute in a different address space, very different from its initial state but still performing as if it is always at the original address space through coded instructions. Agent technology has two perspectives, the end-user agent perspective and the software perspective. Agent technology is in two types, and their migration path classifies these types. Some mobile agents have a predefined path they follow, and these types are often static and called a stationary agent. Simultaneously, the other agent roams freely in a network through a dynamic migration path called a mobile agent. Therefore, a stationary agent is an agent that stays in the same system; it initially started its execution process and communicated with the networks through RPC and messaging. In contrast, the mobile agent is not bound to one system; it can begin its execution process in one network and then end up in another network with its data intact due to its unique ability to transport itself around. Both agent technologies types (stationary or mobile agent) have the following special features.

- ❖ **Autonomy:** This is the mobile agent's ability to act independently on its own while adapting to system traffic demand changes. Mobile agent minimises the potential of technical errors occurrences. It can flexibly connect itself to the task and likewise disconnect once through with the execution of the given task; this, in turn, reduces the human struggle to keep up with the system operations.
- ❖ **Re-activity:** The ability of the mobile agent to react to external events, stimuli and consequently sense its execution environment, which enhances its ability to adapt to any sudden change in the system performance dynamically. The mobile agent also has the ability to replicate itself within a short period to accommodate varying network flow patterns.
- ❖ **Communicative:** The mobile agent has the capacity to communicate with the network so frequently based on the system traffic flow and components operation. It can respond to an incoming task so swiftly because the communication flow of agent within the networks and also to other operation application protocols is exceptional.
- ❖ **Learning:** The mobile agent can quickly learn from the network performance pattern and operate based on any operational state, then improve the system performance by making adjustable decisions based on certain set thresholds by the system administrator.
- ❖ **Robust and Overcomes network latency:** the mobile agent technology can easily dispatch itself from a central controller and regulate system activities through the same controller while overcoming any delay in system response time to any incoming task.

The mobile agent has been evolving over the past decade and has been utilised in various networking resource fields such as network management, networking trafficking, and detecting a fault in network systems design. Not much work has been done using agent technology of cloud data centre network; however, I will be reviewing literature based on the areas mobile agent technology has previously been used. Hence, link our literature findings to the existing problems facing the cloud data centre and thereafter draw an underpin guideline for adopting its approaching cloud networks environment. It has been referred by Bohoris et al.,2002 that it will be impossible to manage the complexity of current network size and connectivity without the help of an embedded agent system. Bohoris acknowledge that the mobile agent approach will change the ideal of network management nightmare into remotely programmable platforms. In 1997, (Biezcza and Pagurek) used mobile agent techniques to design their project management workout to better network system performances and management. Same Biezcza and Paguriek, in 1998, came up with more updated and refined ideas on a theoretical way of implementing mobile code to enhance network maintenance relating their theory to the operation system interconnection (OSI) model with the focus on the 7th layer, which is the application layer. They initiated the plug and play network approach using a mobile agent, which they claim was valuable and critical to leveraging network system management. However, their work was purely theory-based and was not thoroughly tested on the network environment to ascertain their claim.

In 2001, Gavalas used the mobile agent approach to monitor and maintain network system performance. He focused on transferring the management data between managers and the actual application management nodes on the system. His work answered the questions around the simple network management protocol (SNMP) when it comes to managing how data was transferred during runtime, unfortunately, his work did not concern the impact of mobile agent on other network components as the data are transmitted and its implication on the entire network performance.

In 2003, Adhicandra et al. continued investigating data transfer network-related challenges based on Gavalas et al. findings and then want further to ascertain a better agent-based performance using broadcasting and itinerary model for collecting the management data from the network as their case study. He works gradually transported the network data transmission detail to the network manager to update him with the system performance far better than the work done with the SNMP protocol.

In 2017 the institution of Technology, in cooperation with east china Jiaotong University and Nanchang Jiangxi, used mobile agent technology in an entirely new field. The research group used the agent technology as a part of an electric grid to improve network system management and use the agent approach to extract information from the grid. This approach provides fantastic service to their client; however, there was no report to show it has a good turnaround on network performance.

2.6 Summary

In summary, in this chapter, an intensive discussion on the cloud computing concept based on documented literature was reviewed and evaluated. The evaluation was based on the energy efficiency and power usage rate on the cloud data centre, which is currently problematic. Thus, this research work researched what has already been done to mitigate the high power usage rate in the cloud data centre. From observation from previous works, then considered the componential level of the data centre to provide workable solutions. Also examined the agent technology and their characteristics. This research work then proposed the agent technology as a potential solution based on the features it processes and the evidence gathered from the literature of its unique abilities to operate autonomously on a complex system.

From the research survey conducted, it was easy to form the used research questions and then mapped out the phases of achieving solvable solutions, leading to the next chapter of this work.

Chapter3

Problem Formulation and Algorithm Design

3.1 Introduction

This research project aims to investigate existing methods for improving the energy efficiency level of data centres, particularly cloud-based service infrastructure and propose a more efficient method for reducing energy consumption across cloud data centres. This chapter describes the research methodology employed, methods proposed to answer the research questions, and the rationale supporting the researcher's design. There will also be a description of measures to validate the reliability of this study's outcome.

3.2 Research Methodology

This section will describe all the different methods and systematic phases deployed during this research investigations to ensure thorough research findings.

3.2.1 Problem Statement

Finding answers to the questions ask in chapter 2 of this research is essential because, as pointed out in the literature review, cloud data centres increase, exceeding planned energy consumption rates. On the other hand, more businesses continue to move their IT infrastructure to the cloud (Gartner Report,2020). Also, as witnessed with the recent COVID-19 pandemic, there appears to be a surge in the adoption of cloud services such as teleconferencing. It is estimated that cloud data centres energy consumption in 2020 may exceed the projected 140 billion kilowatts per hour (kWh) as observed from literature (chapter 2). These cloud services developments reinforce the need for measures that can help ensure efficient energy utilisation across data centres.

One of the most frequently stated reasons for high energy consumption in cloud data centres is the sub-optimal energy use by components in a data centre, resulting in energy wastage. Examples of components found in a data centre include servers, virtual machines, switches, lighting, and cooling. The effect of this is that energy may be dissipated even when no active work is done.

Typically, in traditional cloud data centres, almost all components are kept in a powered-on state, including non-critical components. This is often a result of fear of failing users' expectations. Consequently, providers tend to provide surplus resources to avoid downtime disruption. Finding the right balance between system availability and resources management is non-trivial because a wrong decision could have grave consequences.

The literature shows that many techniques have been proposed in previous works to address the data centres' energy efficiency problem. Broadly speaking, these techniques are either used to adjust hardware power consumption to match its current workload or power down devices to conserve energy. As shown in Chapter 2, there are two notable categories: statically and dynamic power management techniques. Static techniques are widely believed to be inefficient and difficult to manage, especially in heterogeneous systems typical of cloud data centres. Dynamic power management (DPM) techniques, on the other hand, gives the system the option to adjust to specific performance requirements dynamically. DPM techniques often employ the following strategies: resource allocation, migration, consolidation, scaling and load balancing. Resource allocation commonly refers to the assignment of resources, in this instance, system memory, CPU and storage to a particular task. On the other hand, migration describes a computer system's movement, usually a virtual machine (VM), from one physical hardware environment to another. Lastly, consolidation encompasses the migration of running systems among physical hardware environments based on set policies to improve resource utilisation and energy efficiency.

So far, most of the studies reviewed investigated VMs. Apparently, very little attention been paid to the energy efficiency of other data components such as switches and servers. Therefore, this study argues that whilst VMs continue to play a critical role in data centres in terms of flexibility, performance, and overall energy utilisation, it is pretty likely that some factors applicable to VMs may differ from those that affect other components. As such, the generalisability of methods that investigated only VMs raises some concern. The table below shows the areas at which some of the significant energy-efficient cloud network research investigators emphasized their experiments on.

Table 3.1: Significant energy-efficient cloud network research investigators

Investigator	Techniques			Energy efficiency	Workload Type		Resource allocation	
	VM Consolidation	Host scaling	Switch scaling		Single	mix	Single-layer	Multi-layer
Okonor et al	√	√	√	√	√		√	√
Beloglazov et al.	√	√		√	√			√
Teng et al.	√	√		√	√			√
Liu et al.		√		√		√	√	
Goiri et al.		√		√		√	√	
Hasan et al.		√		√	√			√
Kim et al.				√	√		√	
Nguyen et al.	√	√			√			√

In the next section, we describe the research design and rationale. We then went further to list ways to solve the problem we've already identified during the literature review. A mathematical representation was used to model some of this research findings.

3.2.2 Research Design

This study proposes a model drawing from lessons learned from the literature review that can help better guarantee efficient energy consumption, focusing on non-critical components. To achieve this, a design and creation research methodology was adopted for this project. This is a prevalent research methodology in computing and focusing on developing new artefacts such as constructs, models, methods and instantiations (C.R Kothari, 2004). One of the main benefits of this approach is that it is intrinsically a problem-solving approach and comprise five main steps: awareness, suggestion, development, evaluation and conclusion.

Using this approach, this study reviewed previous works, interacted with practitioners in the industry, and attended conferences on the topic area to understand energy utilisation in cloud data centres and the problems. Building on this, a conceptual model of how to reduce energy consumption, especially across non-critical components, was developed. This conceptual model was thereafter implemented and compared with other existing methods to ascertain its

performance. Finally, we then demonstrated the adaptability and usability of the proposed framework and their practical contributions. A validation process was conducted to validate the research solution by linking back our work to the stated research question. Hence evaluated the research outcome with other existing methods to ascertain its contribution to knowledge and practical impact.

3.2.3 System Model

Figure 3.1 below illustrates the system model proposed in this work. As illustrated in the diagram, the model comprised of the following key entities: users, application, cloud manager. Users: These are the business clients that submit their service request to the cloud network through an application protocol medium. The data centre then receives these requests as a network packet.

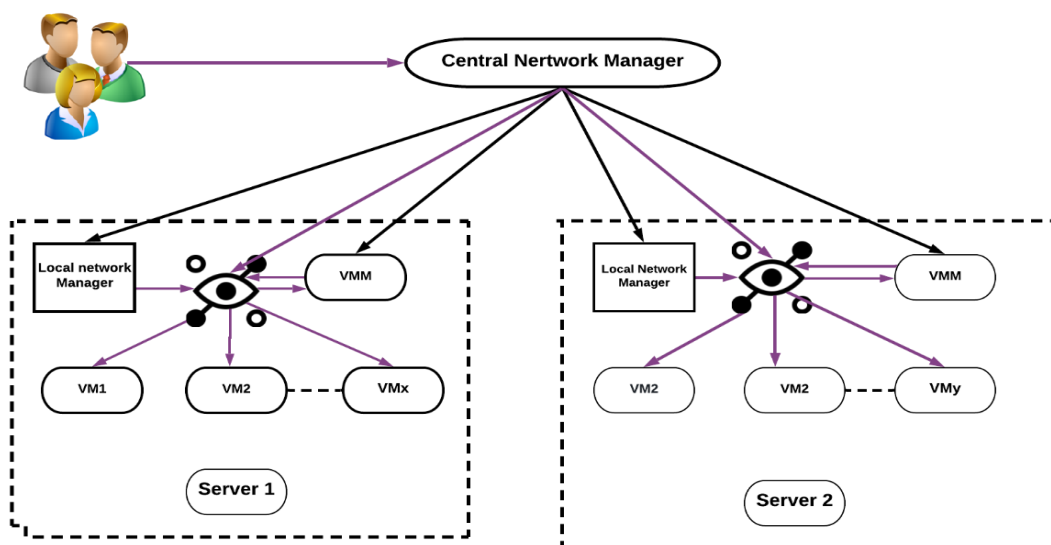


Figure 3.1: System model design

Application: The application entities in this system model operate as a travelling vehicle for the user and a response vehicle for the system network. Inside the system, there are diverse working applications. However, they are session-based.

Cloud manager: They are responsible for fulfilling the cloud services to the client based on request. They manage not only the service request side but also address the resource allocation within the cloud networks.

Critical component: These are the critical parts of the cloud system that must remain active during application run time.

Non-critical components: This part of the system can be set as active or inactive at any given time, depending on the requirement of the system being processed. The component's utilisation level ascertains a non-critical parameter. A mobile agent system controls the activation and deactivation of non-critical components, i.e., adjudicates the appropriate power management strategy to implement based on system performance status and designed component allocation policies.

3.2.4 Power model

To calculate the total power consumed by a data centre (DC), we first considered the formulation done by Zheng et al.,2012 that evaluated a similar concept as the building block for our argument. However, they based the formation on just the server output and then classified it as the data centre's energy consumption, which is arguable based on the DC components. The formula is as follows.

$$P_i^{server} = \begin{cases} 0 & \text{if server } j \text{ is enable, } w_i = 0 \\ p_i^{idle} + \sum_{j=1}^{w_i} \mu^{VM_{i,j}} * p_i^{dynamic}, & w_i > 0 \end{cases} \quad (1)$$

P_i^{server} can consist of idle power and active power. The idle power state is considered to be fixed while the active power is dynamic and linear to the total CPU utilisation of all the VMs on the server. Therefore, if no VMs are hosted on the servers, the server is off entirely to save energy. $VM_{i,j}$ is the j th VM on the i th server, while w_i is the number of VMs assigned to server i .

$$\mu(VM_{i,j}) = \sum_{c=1}^{A_j} \mu(App_c) \quad (2)$$

$\mu(VM_{i,j})$ is the utilisation rate of the $VM_{i,j}$ which is the sum of j th VM applications. c is the component id and A_j remains the application components numbers.

Elaborating in detail on how cloud DC works components will give a clear understanding of what to expect when calculating the energy models involved. The full description of the variables used in formulating this energy model is shown in the Appendix1.

Constraints and Objectives

Equation 1 and 2 are subject to the following constraints

$$\sum_{i=1}^M w_i = N \quad (3)$$

$$\sum_{i=1}^{w_i} \mu(VM_{i,j}) \leq 1, \forall i \in [1, M] \quad (4)$$

The total number of VM is defined as N, while M is the total number of servers. Equation 3 shows the total number of VM assigned to the server w_i equals to the sum of VMs. Equation 4 then shows that total VM utilisation strength can't be greater than the allocated server utilisation.

Therefore, the objective function of the above problem is formulated as

$$\min \sum_{i=1}^M P_i^{server} \quad (5)$$

Proposed Mathematical Formulation

Here observing from equation 1 above, VM utilisation quantifies the cloud data centre's total energy cost. This research argued this point from Zeng et al. because a product performance can't be summed up using just one factor of its functionality. Based on the flaws, bottleneck and research curiosity, the observation from previous works has inspired us to form a new power mathematical model to cover the observed gaps. The proposed mathematical model considered both the critical and non-critical components of the data centre that bring about the data centre's active networking communication state, not limiting formulation to one factor. We calculated the energy used during the network transmission process as the amount of voltage remitted all its resistor. Therefore, for a cloud data centre power cost to be valid, we did sum the total computing power (both the computing and networking device element), which other researchers often overlook. For clarity purpose, each computing and networking (namely routing, connecting and processing) components are represented in vector form as –

Routing elements ($sw_1 \dots \dots sw_m$)

Connecting elements ($co_i \dots \dots co_c$)

Processing elements ($srv_i \dots \dots \dots srv_n$)

Virtual machine element for each server can be written in matrix form as -

$$\begin{bmatrix} srv1 \\ srv2 \\ \dots \\ srvn \end{bmatrix} \text{ contains } \begin{bmatrix} vm_{11} \dots vm_{1n} \\ vm_{21} \dots vm_{2m} \\ \dots \dots \dots \\ vm_{n1} \dots vm_{nv} \end{bmatrix}$$

where

srv_i : the server

sw_1 : the switch

vm_{ij} : the virtual machine

co_i : the connector

This approach gave a better view of how minimising data centre activities can contribute to an energy-efficient data centre. From the presented logic above, I formulate the total amount of power consumed by the data centre at its standard operating rate(normal DC with no agent functionality) and when an intelligent agent is involved. This brings us to calculating the complete power DC(P_{cdc}) consumed at a given time as follows:

$$P_{cdc} = P_{swi} + P_{srv} + P_{vm_{ij}} + P_{fi} \quad (6)$$

P_{swi} is the power consumed by the switches. P_{fi} is the power consumed by the fan. It is worth noting that the switches form the basis of the interconnection fabric that delivers task requests to the hosts for processing. Energy consumption of a switch depends on the following; number of ports, port transmission rate, types of switch and the deployed cabling solution based on the type of data centre operational. Now, this research linearly formulated the switch consumption rate based on the provided switch components information. The energy consumption model for switches is as follows:

$$P_{swi} = P_{chassis} + n_{linecard} \cdot P_{linecard} + \sum_{i=0}^c n_{ports} \cdot P_r * UtilizationFactors \quad (7)$$

Simplifying further for a detailed insight into equation 7 we have the following equation, which is still on a traditional normal state:

$$P_{swi} = P_{chi} + \sum_{j=1}^{n_{lc}} P_{lcj} + \sum_{j=1}^{n_{pi}} P_{rj} * U_{ij} \quad (8)$$

where

U_{ij} : Utilisation factor of an active port j per switch i

P_{chi} : Power consumed by a switch i base on hardware chassis

P_{lc} : Power consumed by the active line-cards

P_{rj} : Power consumed by the active port running at a transmission rate ‘ r ’

P_{aLcswi} : power consumed by the switch when the agent controls the line-card

n_{lc} : the number of linecard

n_{pi} : number of ports

To concern a situation where equation 8 above is under an agent controller, we then formulate the power consumption power as :

$$P_{aLcswi} = P_{chi} + \sum_{j=1}^{n_{lc}} (1 - \beta_j) P_{lcj} + \sum_{j=1}^{n_{pi}} P_{rj} * U_{ij} \quad (9)$$

$$\beta_j = \begin{cases} 0 & \text{if a linecard is enabled} \\ 1 & \text{if a linecard is disabled} \\ & \text{otherwise keep on standby} \end{cases}$$

Where β_j is between 0 & 1 and is used by the agent to control the linecard operation.

Now simplifying equation 9 further, we then get:

$$P_{aLcsw_i} = P_{swi} - \sum_{j=1}^{n_{lc}} \beta_j P_{lc_j} \quad (10)$$

Considering the agent activities on the switches, α_i is tabulated below as

$$\alpha_i = \begin{cases} 0 & \text{if a switch is off} \\ 1 & \text{if a switch is on.} \\ \text{otherwise keep on standby} \end{cases}$$

Where α_i is between 0 & 1 and is used by the agent to control the switch operation.

$\alpha_i = 0$ means the whole switches are powered off,

$\alpha_i = 1$ means the switches are fully active.

Therefore, the total power consumed by an agent enabled switch follows as:

$$P_{agentsw_i} = \alpha_i * P_{aLcsw_i} \quad (11)$$

To a great extent, α_i is strongly related to β_j if all the network-card of a given switch is disabled, then the switch itself should be turned off.

Computing server being one of the computing components we examined to develop a comprehensive model also contributes heavily to its high energy consumption rate. It will be worth noting that this work server and host have been inter-used severally periodically during the cause of this work server. It will be ideal to say that a host server's energy consumption is proportional to the CPU utilisation, which then brings us to the formula below:

$$P_{srvi} = P_{cpu} + P_{memory} + P_{fan} + P_{io} \quad (12)$$

The energy model for the cost of power used by the server when it is under the influence of an agent will then be associated with:

$$P = P_{srvi} - \sum_{j=1}^n \delta^j P_j \quad (13)$$

$$\delta_j = \begin{cases} 0 & \text{if a server is off} \\ 1 & \text{if a server is on.} \\ \text{otherwise keep on standby} \end{cases}$$

Where δ_j is the agent controller and it can also be represented as $0 \leq \delta_j \leq 1$

Then

$$P_{S_{rvi}} = \sum_{j=1}^n P_j - \sum_{j=1}^n \delta_j P_j = \sum_{j=1}^n (1 - \delta_j) P_j \quad (14)$$

Following the same reasoning as in the switch formulation in equation 9 & 10, the power consumed by the server under the intelligence of an agent follows the same logic and parameter as the α_i and β_i in switches, however, in this case, measures for *cpu, memory, fan and P_{orts}* . The power consumed by the fan is related to the power consumed by the CPU & memory, and this result can only be obtained by simulation.

Understanding from the literature that the VM is the basic unit of allocation and consolidation resource in the data centre, we considered its power model using the following terms that are vital to its operation.,

We define VM as

$$Vm = \{op, r, \Delta t, host\} \quad (15)$$

Where

op = operation of the Vm

r = the request scale of the Vm

Δt = the execution time of Vm

host = the hosted node of the Vm

The total power for VM will then be -

$$P_{Vm} = \sum_{j=1}^{wi} P_{Vmi,j} \quad (16)$$

The load balancing of the VM allocation is determined by the power threshold set by the provider on the broker controller(both upper and lower limit) to reduce unnecessary load on the physical machine(server) hence avoid triggering sudden extra workload on the fan. The fan's power is, to a certain extent, maintain to avoid constant high activities on it due to oversubscription of VM and server allocation. The agent approach, therefore, migrates overloaded and oversubscription VM to other underutilised servers to be able to control the power consumption of the servers using the defined threshold will now be represented as:

$$P_{VM_{agent}} = \sum_{j=1}^{wi} au^{VM_{i,j}} \quad (17)$$

Where au is agent utilisation.

Applying the defined conditions, we then calculate the total power cost of the cloud computing device, which is the summation of the power formulation for switches, VMs, servers and fans under agent scenario:

$$P_{C_{agent\ DC}} = \alpha_i * P_{aLcsw_i} + \sum_{j=1}^n (1 - \delta_j) P_j + \sum_{j=1}^{wi} au^{VM_{i,j}} \quad (18)$$

System Reliability(SR) Performance Test Metric

For any model to mathematically show a significant value of improvement in the cloud data centre environment, it should be proven that its operations did not obstruct the system performance and hence did not violate the system reliability. It is worth stating that understanding the system adaptability to both internal and external factor that aids better system performance is vital. During workload migration, interaction and system shutdown, there are bound to be some interruption and request failure with some predefined system reliability test metric which mathematically is represented as follows

$$System\ Reliability = \frac{num_f}{num_a} \quad (19)$$

Where num_f = number of failed requests

num_a = number of all incoming requests

With constraints on all incoming requests as follows $SR \leq \alpha$ to validate the quality of the QoS where all allocation of workloads considers the average response time($avg(t)$) it got for distribution of requests to happen based on a given threshold $(avg(t)) \leq \beta_j$

Therefore, while trying to model an agent approach in the data centre network, we were aware of the stated performance metric and consciously maintained a minimal violation during agent execution time. Putting this idea in a mathematical context, we then represented the SR violation(level) cause due to over-utilisation of server mostly when the CPU performance was at maximum as:

$$V.SR = V.SRo\mu * V.SRm \quad (20)$$

Where $V.SRo\mu$ is the percentage of time at which the physical machine (PM) (such as servers, CPU, etc.) experience up to 100% utilisation(operational utilisation). This can be represented as follows:

$$V.SRo\mu = \frac{1}{M} \sum_{i=1}^M \frac{T_{si}}{T_{ai}} \quad (21)$$

Where M is the number of PMs; T_{si} is the total time of 100% utilisation of the PM, that led to SR violation, while T_{ai} is the total time the PM_i was in an active state.

$V.SRm$ represents the overall system performance degradation caused by migrating a VM from one state to another. This is represented as

$$V.SRm = \frac{1}{N} \sum_{j=1}^N \frac{C_{dj}}{C_{rj}} \quad (22)$$

Where N is the number of VMs; C_{dj} is the estimate of VM_j performance degradation caused by migration, C_{rj} is the sum capacity of the CPU requested by the VM_j during its active migration lifespan.

Conceptual model

The concept of distributed computing, especially cloud computing, is becoming most popular for networks communications. However, it comes with rising demand for its system processing

power and maintaining an acceptable SR based on the system performance level. According to (Pleisch and schiper, 2004a), the concept of finding a trade-off toward liberating the restrictive functionality of individual computing components like servers and switches, which are defined by its computing interface, can be challenging. These computing interface challenges have led to the service providers' inability to regulate the cloud computing components adequately, thus costing the vendors a fortune to operate and subsequently limiting cloud technology adoption. Consequently, the mobile code-computing paradigm has also been noticed and recommended because of its robust performance mode on data and network transmission domain.

Interestingly, the features of agent technology are beneficial to distribute computing network to leverage the complexity of the system; however, the motivation for chosen mobile agent technology is far beyond the features and is encapsulated in these mentioned seven reasons such as:-

1. **Agent reduces the network load:** A computing system often depends on different communication protocols that involve multiple interactions to be enabled to work on a given task. Mobile agent aims to dispatch the package to a destination server where the interaction can happen locally in the network. The agent also reduces the inflow of raw data into the system by moving computation to the data rather than the data to the computation.
2. **Agent overcomes the system latency:** Mobile agent brings solutions to critical real-time system problems with a high need to respond to real-time changes in due time. Agent device dispatches itself from a central controller to act locally and directly execute the controller's instructions.
3. **Agent executes asynchronously and autonomously:** In distributed networks, mobile devices rely heavily on a fragile network connection, making tasks that require continuous connectivity between the mobile device and fixed network infeasibly tricky to manage (both technically and economically). The agent system then solves this complicated problem by providing a task embedment approach with a mobile agent.
4. **Agent adapts Dynamically:** Agent technology can understand its executing environment and dynamically react to any new change autonomously. In cases where there are multi-agent operations, they often distribute themselves to all the hosts in the networks to maintain the best position for solving a specific problem based on system configuration.

-
5. **Agents are heterogeneous:** A computing system is primarily heterogeneous both from a hardware and software viewpoint. Due to this diversity in network system content, the mobile agent depends only on its execution environment while offering other optimal conditions for seamless network integration.
 6. **Agent encapsulates protocols:** In a distributed system, it is challenging to implement protocol coding of outgoing and incoming data exchange, resulting in a legacy problem in network protocol. Mobile agent solves this problem by moving to a remote host which establishes a channel for mitigation through proprietary protocols.
 7. **Agent technology is robust:** The agent has a high tendency to react dynamically and promptly when a network computing situation changes or an event suddenly triggers in a distributed system. If a server is being shut down, all agents operating on the said server will receive an alarm and be given time to dispatch to another server or kill themselves for smooth network execution to continue despite the incident.

Therefore, introducing a mobile code concept to cloud network vendors will bring relief and workable solutions to existing cloud platform challenges. They will use the code to modify and regulate the functionality of the network components flexibly. Our research work used the agent technology to solely bridge the gap between the cloud data centre high energy consumption and its performance level based on SR. The mobile agent technology here added an intelligent new and exciting communication paradigm to the existing network communication procedure. A mobile agent can move between machine without interruption. The system design makes this approach unique to striking a neutral ground to managing the complexity in cloud architectural design and its performance efficiency. Observing from the mathematical model above, which has shown that the traditionally based scenario had limited flexibility in accessing the cloud data centre components without partially or entirely degrading the system performance. However, the flexibility, autoimmunity, re-activity and learnable nature of the mobile agent makes it the best approach for solving the complexity associated with the data centre complex design. According to (Claudio et al. 2017), the cloud system depends on a data centre network's geographical distribution. It displayed a clearer picture of how complex it is to manage the cloud network. A display of agent tool is thereby highly recommended based on these three agent attributes, namely:

- Agents are at their best, where the problem domain is geographically distributed.
- The subsystem exists in a dynamic environment

-
- The subsystem regularly interacts with each other flexibly.

With respect to these attributes, mobile agent codes are a potential solution to cloud complex challenges. Figure 3.2 below shows the characteristics of mobile agent technology that can be leveraged in the cloud DC network. It will drastically and efficiently have a positive impact on the system's energy usage rate. The mobile agent can communicate with clients, servers and switches through message-passing (MP) or Remote-procedure calls (RPC). (Nikaein,1999) outlined that the agent can suspend itself after sending a request to the servers and waiting for the response to minimise energy use on its part and kill itself after executing a given task. Therefore, making the injection of an agent into a cloud system beneficial as it causes no negative impact on system performance

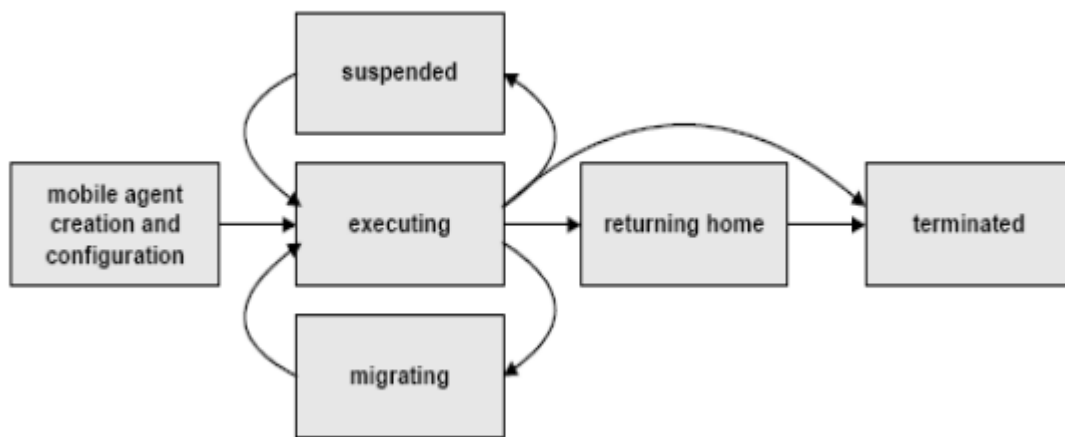


Figure 3.2: The mobile agent performance lifecycle on the cloud data centre system

Furthermore, bringing back the context at which (Lange and Oshima 2003) presented the mobile agent idea as a java code transported in block form with program instruction makes the approach more appealing and exploring. That's why this research explored its features in our research work. The mobile agent technique shows significant improvement when compared to other existing approaches and less performance degradation from the experimental viewpoint because it maintains the same state and code while travelling through the network. Furthermore, the ability of the agent to maintain its original state during transmission gives agent technology unique identification and stability. This feature helps the agent approach execute its functions at any given time with an effective outcome. This leveraging features of the intelligent agent approach on the cloud system is shown in Figure 3.3.

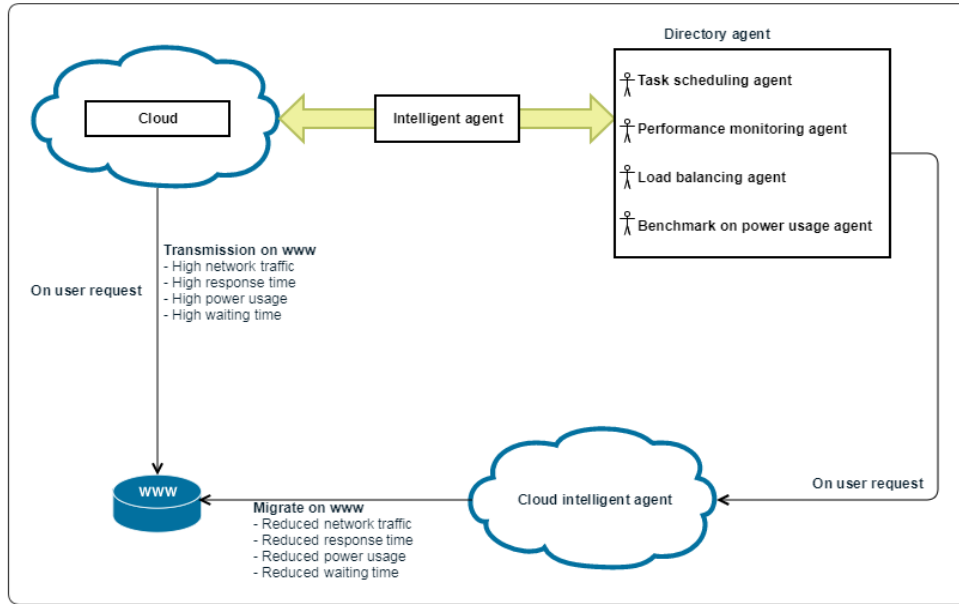


Figure 3.3: Mobile agent deliverables to Cloud Data centre

Figure 3.3, the agent code travels through the data centre networks and intelligently monitor, maintain, automate and manage the variation in the power consumption. The agent then gets update messages from the broker controller that facilitates its activities and guides the actions the agent code takes on the system. After a dispatched agent has finished its assigned task, the controller then terminated its life unless it is reassigned to a new task.

Figure 3.3 gave the clarity that led to developing a flowchart based on the mathematical formulation to build our simulation and guiding framework for an achievable agent in cloud system performance. To the best of our knowledge, our proposed system is the first of its kind to be used in the cloud data centre; however, agent technology has been used in the past on the ad-hoc network. The flowchart in figure 3.4 shows the resultant effect of the design adhered to during different stages of this research work. Therefore, this Flowchart is the original prototype for the power optimisation processes of the Cloud DC we investigated during the cause of this research work. Subsequently, we elaborated on the flowchart when discussing specific techniques toward achieving energy efficiency on the subsystems, such as the switch and server power usage rate, VM migration toward minimising physical machines' inefficiencies.

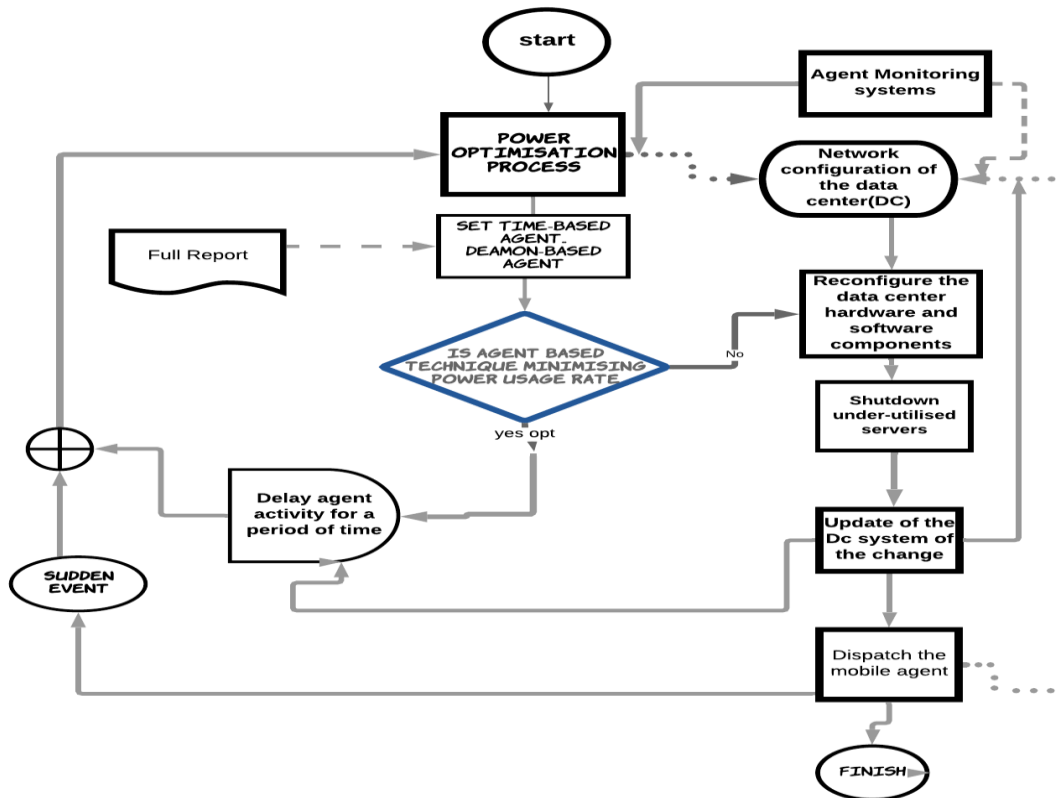


Figure 3.4: Flowchart for system power optimization process of Cloud DC

3.3 Description of Algorithm

The algorithm in figure 3.5 below is the pseudo-code for our proposed intelligent mobile agent operation on the VM. The idea is to intelligently migrate any underutilised VM from its original sitting position to another physical server (host). The mobile agent has been fed with the hostlist, and the Vmlist, too, from the simulation created classes. The agent's role is to fix the problem that was messaged to it by the broker controller or if it can't handle it. It promptly reports back the information about the network state stored on the information registry that feeds the controller. For clarity purpose, all servers in the system are all potential sources for VM residing location. The server and the VM are characterised by their resource utilisation rate, namely CPU and memory utilization. A VM can be dynamical or statically migrated based on the defined requirement. However, being that VMs are portioned on top of the servers, also known as the physical machine (PM), the process of migration of any VM should be logically, pragmatically and intelligently done to avoid system performance degradation. Thus, this proposes algorithm intelligently sort VM utilisation rate while autonomously moving around the

system. Suppose the process encounters any VM with less utilisation rate less than the set rate for normality. In that case, it sets the alarm for other agents to be alert and sort also the host that is also under the same category. The agents will now migrate the VM to another server and shutdown the underutilised server on the spot while continuing with the VM's placement to the adequate pm destination. Therefore, the intelligent agent carries a set of rule that is the yield stick for its actions on the system. The agent has been flexible and can flexibly interact with the system to ascertain the resource utilization rate of every server and VM in the system through their list. For every resource move or shutdown, the agent is speedy enough to update the system constantly for the resource's immediate status to avoid bottleneck during task distribution.

Algorithm: Intelligent technique for Migrating VM for Power saving

Input : *hostList, VmList*
Output: intelligent live migration of VMs

```

1 hostList:SimAgent to choose from hostList where to migrate VM based on certain threshold
2 VmList:SimAgent sort under-utilized VMs from VmList
3 foreach  $A_{ij} > \sum_{i=1}^{|h_j|} A_i$  do
4    $MinPower \leftarrow MAX$ 
5    $A_{ij} > allocateHost \leftarrow NULL$ 
6   foreach host in hostList do
7     if agentHost indicates host has enough resources for vm then
8        $Power \leftarrow estimatePower(host, vm)$ 
9       if  $Power < MinPower$  then
10         $allocateHost \leftarrow host$ 
11         $MinPower \leftarrow Power$ 
12      end
13    end
14  end
15  if  $allocateHost \neq NULL$  then
16     $Migration.add(vm, allocateHost)$ 
17  end
18 end
19 return migration

```

Figure 3.5: An intelligent technique for migrating VM for power saving

3.4 Development Environment

The approaches proposed in this thesis have been implemented and tested in an environment with the following different server specification, as recorded in Table 3.2 below. The server had multicore CPUs, and each of the multicore CPU has K cores, while the core has M MIPS as a single core which makes the CPU sum up its total capacity as K*M MIPS.

Table 3.2: Server's specification

Server	CPU Model	Cores	Frequency(MHz)	RAM(GB)
HP Proliant G3	Intel Xeon	2	633	4
HP Proliant G4	Intel Xeon 3040	2	1,860	4
HP Proliant G5	Intel Xeon 3075	2	2660	4
IBM server x3250	Intel Xeon 3470	4	2935	8
IBM server x3250	2* Intel 5675	12	3067	16

Although cloud computing research attracts a lot of attention, very few simulator environments and research platforms can model cloud challenges. Cloud data centre network often operates on a heterogeneous and sometimes proprietary-owned mode, making it difficult for researchers to implement prototypes policies on the real hardware. Considering the cloud system's architectural structure, it can be difficult to change a prototype policy or adjustment when testing an idea. This is the justification for using a simulation environment as a benchmark application tool to open up the possibility of research ideas to real-world scenarios. There have already been some simulation tools used for grid computing, such as SimGrid (Buyya and Murshed) and OptorSim (Bell, D. G. Cameron and Zini). However, grid-enabled cannot model a virtualised environment and had less API to model cloud resources effectively. Table 3.3 is a tabulated review of some selected simulation tools that have been used in a virtualised environment and their limitations based on our research work requirement.

Table 3.3: Review of selected simulation tools

Simulator	Function	Limitation	Reference
NS 3	Simulate general networking hardware structure and performance	Cannot be used for model cloud-specific features such as workload scheduling, VM placement policies and prediction techniques	http://www.nsnam.org/ .
iCanCloud	It simulated a large scale of cloud resource cost performance	It is designed for just cost modelling and cannot be used for energy efficiency-related works because it enable APIs for it.	A. Nunez, Vazquez-potletti, Caminero, 2012
RC2Sim	Simulate interface that is used to test to test the cloud management software on a single host	It is not extendible and cannot be used for energy efficiency-related work on the hardware layer.	D.Citron and Zlotnick , 2013
NetworkSim	Simulated network elements to estimate just the network transmission time	The tool is not flexible enough to adopt other ideas and hard to extend. There the agent approach should it work in it	Garg and Buyya, 2016

CloudSim 3.3.0 simulator is the framework used in the research work. It is a discrete event-based cloud simulator implemented in java, allowing flexibility for data centre structure to be simulated in its platform. It gave us the flexibility to use the API's to model cloud environment replicating different scenario. When we encountered challenges on adding an agent API into the simulation tool, it was easy to extend the framework to design a tool that considers a mobile

agent's behavioural impact in a cloud platform. Cloudsim was able to simulate cloud data centre, physical machine, switches, network links, and virtual topologies to measure both performance metric to guarantee QoS and the system's energy efficiency level. CloudSim is a compact java-based multi-layer simulator with the following attributes that makes it suitable for this research work in the cloud:

- It allows for the creation of cloud entities based on the research requirement.
- It offers a changeable infrastructure and network modelling tool
- Allows users to define and test new cloud policies in the platform
- Provides VM, host, network and application provisioning.
- It can model and simulate large scale cloud data centres
- It has libraries that can model the energy management of the data centre configuration, which has always been the limitation of other cloud simulators.
- Users can easily compare and test their newly implemented policies with other existing policies embedded in the cloudSim3.3.0 framework for easy validation.
- Cloudsim 3.3.0 provides an enabling platform for a developer to be able to organise their code effectively

Figure 3.6 shows the class diagram of a Cloudsim 3.3.0 and makes it the right choice for this thesis experimental evaluation simulation tool.

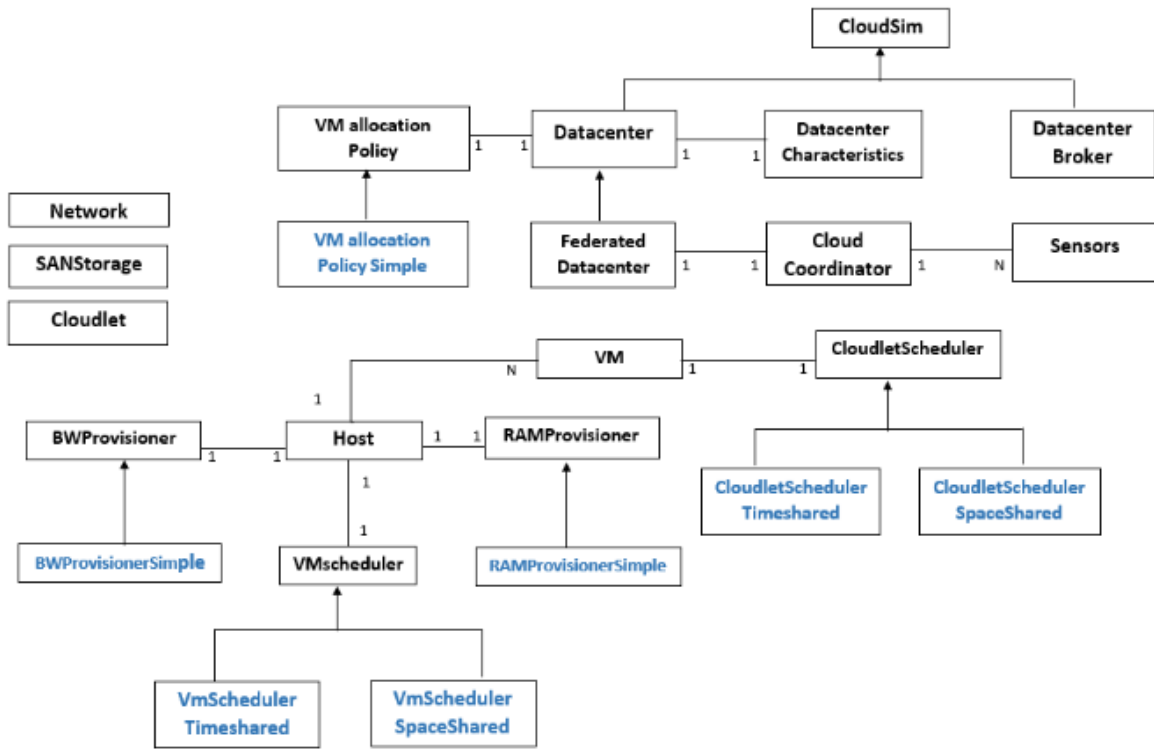


Figure 3.6: Usable classes in Cloudsim 3.3.0

Observing from figure 3.6, it can be seen that Cloudsim, has many extendable platforms which allows events to add another event to the timeline during runtime. For instance, the event that optimises power consumption can add virtual machine migration to the timeline or host shutdowns. In a CloudSim simulator, once tasks are triggered to run, the simulation environment continues without stopping to be active until the last job finishes its execution. This feature is unique, hence enabling us to test our energy management algorithm and distinguish their functionalities. For better understanding, we further explain some of the classes in Figure 3.6 as follows -

Data centre: this class contains many hosts, switches and virtual machine.

Cloudlet: A cloudlet in CloudSim context is the task. it is defined in the CloudSim environment by its length, which is the number of instructions executed by the processor to finish a task

VMAllocationPolicy: it is the abstract class used by the CloudSim tool to allocate hosts to VMs.

Host: the class represent the physical computing machines that contain the memory, storage, I/O resources which enable a researcher to be abstract the underlying hardware dependencies.

VM: VM classes represent virtual machines. In a CloudSim environment, VM is created for each user task submitted in the platform; then, the created VM is allocated to host machines. CloudSim simulation contains 12 packages. The 12 packages are sectioned into two main categories: the core entity classes (host, data centre, VM, cloudlet) and the associated classes (VmScheduler, VmAllocationPolicy, CloudletSheduler, UtilisationModel and DatacenterBroker). The justification for choosing this version of CloudSim is that the power package can extend all these named classes to calculate their components' power consumption rate during run time.

Based on the defined scope of this research work, CloudSim 3.3.0 simulator still has its limitation that nearly hindered the progress of this work because it expects that users should use the power and the network package simultaneously, which for some testing scenario like ours will override its purpose which taking into detail the power consumption rate of critical and non-critical components of the cloud data centre. Therefore, to understand the networking component and their power usage rate, separated means derived from observing each computing element separately. This led to the extension of CloudSim 3.3.0 to AgentCloudSim 3.3.0, where those challenges were addressed by creating additional classes that monitored the power and network package performance separately. Secondly, CloudSim 3.3.0 packages could not understand the ordinary extension of classes to incorporate agent technology within the existing class. However, the AgentCloudSim framework creation made room for additional network packages that included the agent entities classes such as AgentDatcenter, AgentHost, Agentswitch, Port and RawPaquet. Below in figure 3.7 is the diagrammatical representation of the AgentCloudSim framework.

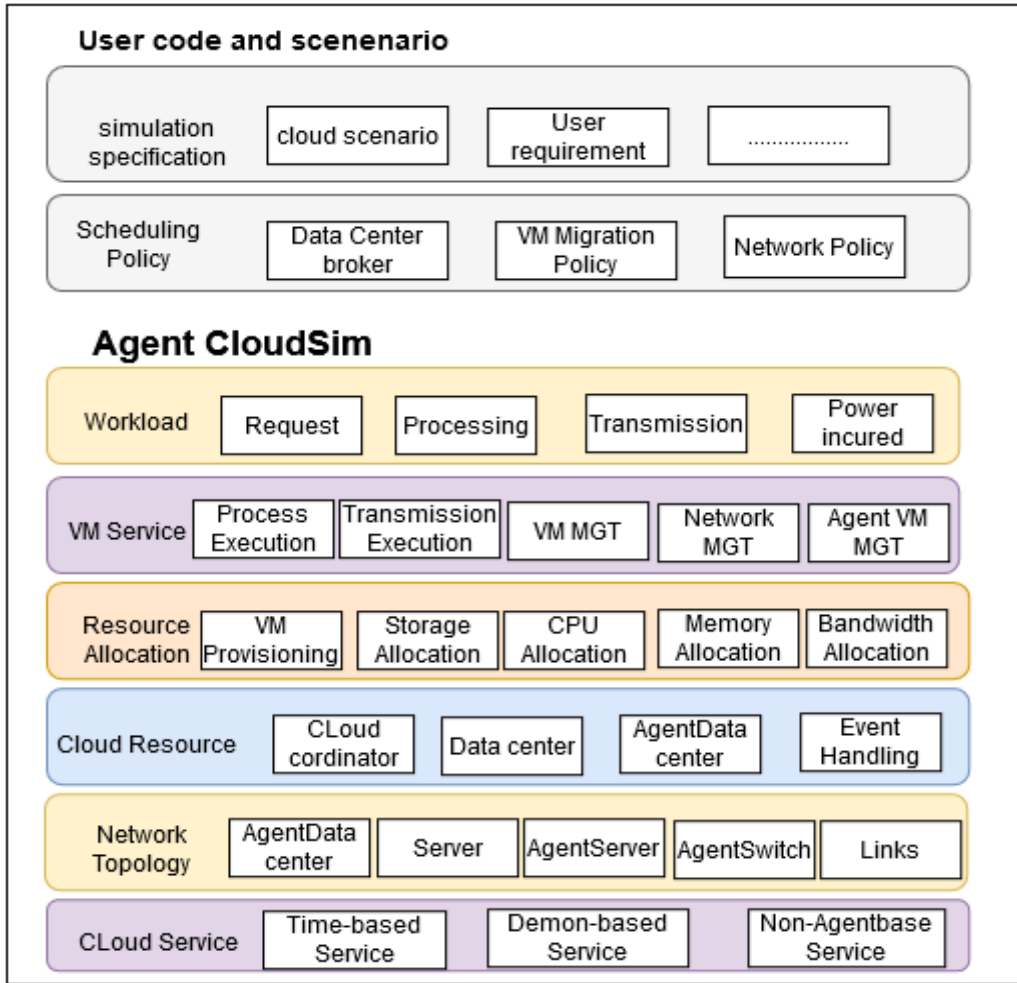


Figure 3.7: AgentCloudSim Framework

This research work used the three-tier data centre topology during the design and the development of the research case examining scenarios. The three-tier data centre architectural component consists of the following—the core layer containing $(n/2)^2$ n -port switches. The centre layer (aggregation level) has $n/2$ switches, and the Access Layer connects to the servers, also called hosts; the VMs resides in the servers housed by the access layer. We compared our design to cisco vendor design where their system consumes the power of 8-port (2960-8TC-L), 24-port (2960-24TC-L), and 48-port (2960-48TC-L) and their switches consume 12W, 27W and 39W, respectively (Cisco, 2017). Observing the Cisco proprietary's power consumption gave us insight into what to expect when developing a simulation environment with our synthesised topology. Figure 3.8 is the designed three-tier data centre structure we used to explain the DC structure and its connection to the simulation development.

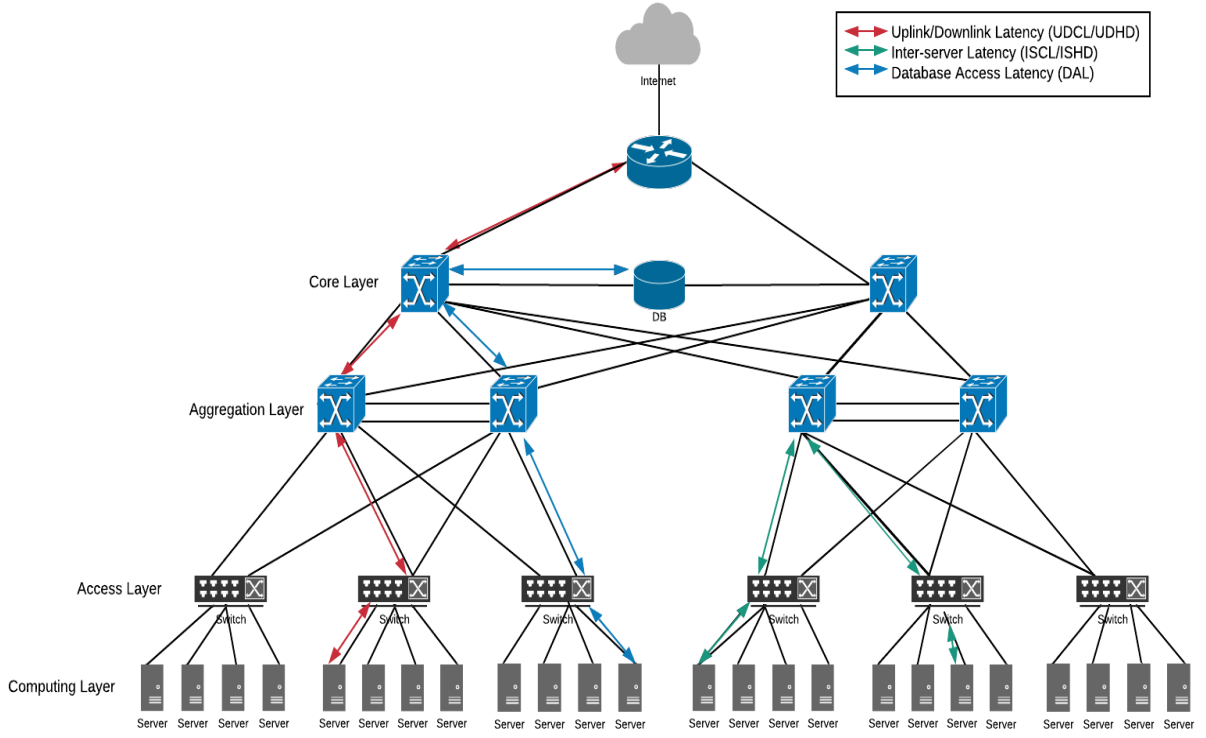


Figure 3.8: Data centre structure

Observing Figure 3.8, it is evident that each server is linked to a switch. In the simulation design development stage, we project that Agenthost contains a UpperSwitch parameter to correspond with our design topology. Each switch is then linked to one or more switches above and one or more switches below. Therefore, in this environment, the SimEntity class has the extensible feature where the AgentHost, Server, Switch and AgentSwitch classes can be added. Now on performance scenario, when a task is sent to the system, the task is first converted to the packets based on the networking protocol operation mechanism, then call the method sendrawPacket on the switch or server.

3.5 Simulation data Source

The source of the data used in this research work is from CoMon project, which has a PlanetLab monitoring infrastructure (Buyya, Ranjan, and Calheriros, 2012). PlanetLab's data models a real system workload traces, and this attribute of PlanetLab provides insight into the actual performance traces scenario. In PlanetLab, the CPU utilisation of the VMs was simulated. Its utilisation values were gathered every 5 minutes and from more than a thousand VMs instantiates from servers in more than 500 locations around the world. Furthermore, 10 days of

PlanetLab workload performance traces were randomly collected over time from March to April 2011 timeframe, and this exercise brought about 11,746 24-hour total resultant workload traces. PlanetLab's workload is IaaS based can be likened to AWS EC2 cloud environment because its VMs are created and managed by multiple independent users, while the infrastructure provider is not aware of the application workloads in the VMs. The PlanetLab vast simulation workload data-trace gave us the flexibility, repeatability, and reproducibility to experiment with repeated time weighing and evaluating different scenarios' result outcome. It is worth noting that the PlanetLab dataset is not an actual AWS or other cloud vendor dataset. However, it is an application that is highly close to HPC system performance, which is familiar to the public clouds, thereby making it widely accepted in a research environment for cloud experiments. Furthermore, HPC workloads are often easier to manage when considering a VM consolidation system workload because of the resource utilisation's infrequent variation.

3.5.1 Resource Type

This resource work used a heterogeneous resource type. It is what noting that the homogenous resource cannot effectively satisfy the cloud operational condition because its service is limited to single types, which will fail to thrive in the cloud online distributed platform. Therefore, justifying why, we used the heterogenous resource type due to its flexibility and adaptability during task scheduling and workload allocation. For instance, Amazon EC2 has provided over 50 types of VMs in its platform. These VMs are categorised based on amazon set conditions such as (general-purpose, computation-intensive purpose and memory-intensive purpose (Beloglazov et al.,2014) for optimal utilisation of VMs components. The PlanetLab provides an IaaS infrastructural environment that enables independent applications to run on its platform, one of the critical requirements we need to perform our research experiment successfully. Therefore heterogeneously, we measured and evaluated the power usage rate and CPU intensity with lower dynamics of multiple independent cloud data centre resources and applicants.

3.5.2 An Agent Technique for intelligent VM consolidation and Migration

In this section, an agent technique was used to consolidate VM usage and migration VMs dynamically. Before deciding which VM to consolidate and migrate, we subsection the factors

we considered: server underload detection, server overload detection, and selecting VMs that would be migrate based on their current state.

Firstly, we used an agent policy algorithm to determine which server is on overload level above 75% of its performance capacity. The agent then searches for the server under 40% utilisation termed underloaded server and then migrates Vm from overload server to them through the intelligent agent VM allocation scheme. Then, for any server with less than 15% load, the agent system flexibly checks for available servers in the set threshold range, moving the load to. In the initial starting stage of the migration, the intelligent agent system sorts all the VMs in decreasing order of their CPU utilization rate based on the information from the cloud registry and then start the migration task. Hence shut down the underutilised server that VMs has been moved from to avoid redundancy performance. Figure 3.9 below is the flowchart for implementing VM allocation using the agent technique.

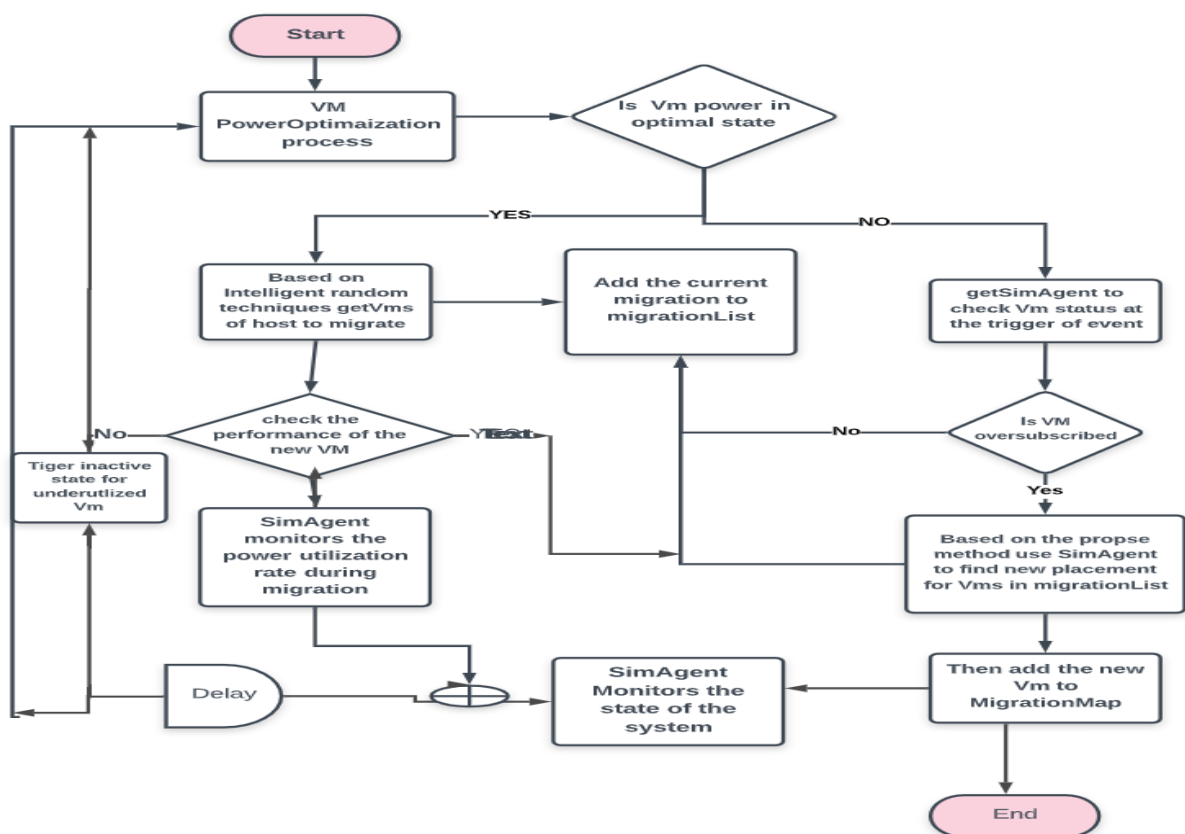


Figure 3.9: Flowchart for implementing VM allocation using agent technique

The importance of this can be seen from a two-dimensional viewpoint. Before the intelligent agent technique, the static CPU utilisation threshold approach has been operational; unfortunately, it came with various bottleneck because of its fixed nature in a complex distributed cloud environment with unpredictable workload scheduling scheme, especially that it always set CPU utilisation to 100% making performance degradation inevitable occurrence.

3.6 Evaluation

This study proposed a new method and designed a prototype tool demonstrating the functionality and applicability in addressing the research questions asked. This study conducted two tests to affirm that its method/tool are consistent with its developed mathematical model and that its results are reproducible. The first test ascertained the correlation between daemon-based agent method, time-based agent method and non-agent method on cloud system and how consistent their communication was across varying workloads on the network. The second test determined the level of reduced power utilisation achieved by the different agent techniques at various stages, which data center component was most effective and the prerequisite conditions for optimal performance.

Similarly, this research considered the impact of its proposed method on system performance, i.e., high-performance degradation cases during system consolidation. Based on the observation from past works and their associated limitations, this study thereby tested and observed the cloud system operation during the triggering and injection of agent technology into its network. There was a probability that the system may experience varying system performance levels beginning from when the different agents are triggered because agent first starts its activity as an external factor before adapting to the system trend. Therefore, it tries to learn the current system pattern so quickly until the system has been successfully consolidated and attained system stability. Thus, assessed this research method's overall impact on how reliable the system was during the cause of using an agent and then observe if there were violation and performance degradation on the network or the system at different stages, this is called system reliability (SR) violation. The SR was also mathematical model, which computes the average SR level for all active servers and the average performance degradation level score of migrating a live VM from one server to the other without obstruction.

The PlanetLab dataset with a load varying from 100 VMs up to 2400VMs was used to test this research method's reliability and validity in a data centre environment. Different simulations

using an incremental workload were run to capture the different behaviours, observing the throughput, System performance reliability level, and power dissipation at each stage. The various methods assessed handled the received cloudlets using available hosts.

3.7 Summary

This chapter has introduced the methodological approach used in this research work. The demon-based, time-based and non-agent intelligent scheme was used to schedule and monitor system performance in the cloud data centre. This research work also proposed an intelligent mobile agent policy to regulate and migrate VM based on their performance to avoid overloading and underutilising VM and physical machine. These three approaches invent and implanted in this study aimed to reduce cloud systems' energy consumption rate while actively servicing customer requests. During the simulation, the Java agent was used on two scenarios time-based scenario and demon-based. The time-based agent was able to capture the switches, server and links utilisation activities and shut down any component generating power without having an equal measure of active service. The next chapter will now be used to discuss the implementation and evaluation of this approach after the simulation of this approach using the agentCloudSim simulator.

Chapter4

Results and Evaluation

4.1 Introduction

This chapter discusses the results of the experiment during this research work. In order to compare the effectiveness and the performance of the proposed model used in this research work to other existing approaches, the energy consumption level of using agent technology on data centre switches, virtual machine, servers and the other non-critical components were captured, measured and evaluated.

In this chapter, the experiment was conducted on various scenarios using the agent code to ascertain the effectiveness of integrating agent attributes to different components of the data centres. Therefore, this research work considers the shutting down of switches and servers intelligently, underutilised or over utilised based on a set threshold defined in the java agent code. Secondly, the agent code monitors the virtual machine activities, hence migrating VMs based on the same agent technique but focusing on the VM capacity and limitations attached to each VM and its associated host. Then the system performance was examined based on the QoS's reliability during the cause of using this approach on a cloud data centre by evaluating the behavioural impact of agent code on the system as it communicates with the entire network. Finally, the analysis of the result obtained in this experiment with other existing valid findings is discussed.

4.2 Virtual Machine Migration Result Discussion

From the observations based on the conducted research, the mobile agent encapsulates the virtual machine's behaviour, then acts on it through the migration of VMs. A competitive analysis of an agent-based migration of VM in an intelligent way to ensure dynamic VM consolidation problem got a solvable solution was conducted and then compared to other existing approaches. From the obtained results, it is evident that migrating the VM instances at the approximate time to avoid underutilization and overutilization of virtual machines on top of the physical servers through a hypervisor's aid improved the system performance metric. This also aids in shutting down network components that were supplying electrical current to

the instances, which saves a significant amount of power during the processing time compared with other research works. Figure 4.1 shows that at each traffic flow, which is from the time the cloudlets were sent to the processing queue to the time it was returned to the client, as a complete executed task. The virtual instances receive and allocate cloudlets based on the set capacity. The agent system monitors the transitional flow and decides which VM to migrate based on the defined thresholds and constraints. This intelligent method has access to the entire system and with very high-speed performance level. This resulted in the system's increased agility and the amount of energy saved at each virtual machine's migration period using our algorithm was significant. At each phase of the traffic metric, a substantial amount is saved. As shown in Figure 4.1, the system has maximum energy savings of 90% and a minimum saving of 49%. This shows a very high chance of using mobile agent technology to break the deadlock on the cloud system complexity challenges. Despite the savings observed at each stage, there is minimal disruption to the system's performance due to the agent's attribute to flexibly and autonomous movement around the network without interrupting the system distribution process or traffic flow.

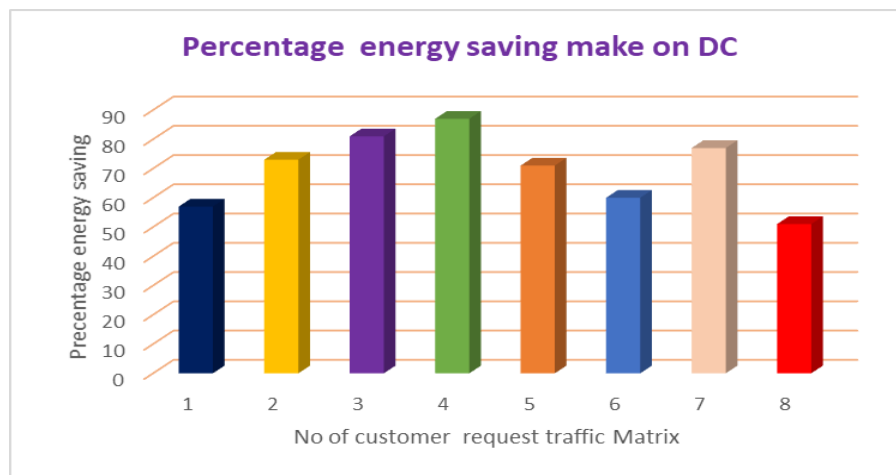


Figure 4.1: Amount of energy saved at each virtual machine migration period

Therefore, measuring the agent's performance based on a single traffic flow of cloudlets which can also be called tasks, has an outstanding progressive result outcome which encourages the adoption of using agent scheme on the other components and network's monitoring phases. Furthermore, another phase of the experiment on VM instances was conducted with varying VM capacities and different time intervals in varying days, as shown in Table 4.1. The result shows that at each stage, the agent technique can save energy at all levels. However, at some

point, the demon agent has higher performance with fewer energy savings; therefore, to evaluate this work, the mean, range, and standard deviation (SD) mathematical method is used to obtain the average performance, enhancing performance as shown in Table 4.1.

Table 4.1: Workload based characteristics of the CPU utilisation

Data	Number of VMs	Mean (%)	S.D (%)	Median (%)	Quartile (%)
03/03/2011	1052	12.31	17.09	6	2
06/03/2011	898	11.44	16.83	5	2
09/03/2011	1061	10.07	15.57	4	2
22/03/2011	1516	9.26	12.78	5	2
25/03/2011	1078	12.08	14.14	6	2
03/04/2011	1463	10.56	16.55	6	2
09/04/2011	1358	11.12	15.09	6	2
11/04/2011	1233	11.56	15.07	6	2
12/04/2011	1054	11.54	15.15	6	2
20/04/2011	1033	10.43	15.21	4	2

It is worth noting that for each day in Table 4.1, not all the VMs stated were used. This selection criterion was strictly based on request and performance demand, which means that some of the instances will automatically be shut down or put to sleep mode by the agent technique. The importance of having more VMs instances can not be overlooked in a cloud environment, for there are always needs to have redundant backup machines to ensure the system is highly reliable as promised by service providers, thereby making rooms for overprovisioning of resources in case of sudden service request trigger or any case of disaster recovery. Figure 4.2 shows the impact of the mobile agent technique on the cloud data centre.

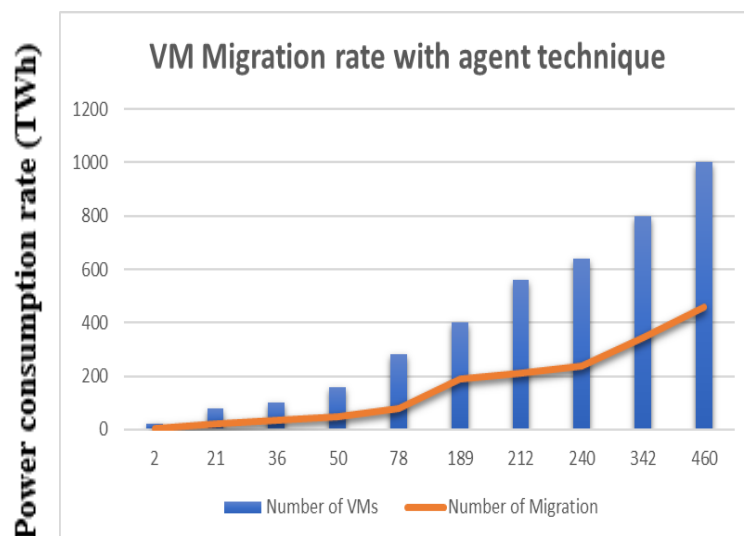


Figure 4.2: Impact of mobile agent technique on cloud data centre system

When the agent code was induced into the system on a 5-minute interval, it was able to systematically migrate several VMs at each stage based on their performance level with incoming workloads. The mobile agent also determines this under a set threshold, emphasising maintaining a high level of system's performance and low power usage at each stage. Figure 4.3 shows the power used during the migration and the system's violation during migration from one virtual instance to another.

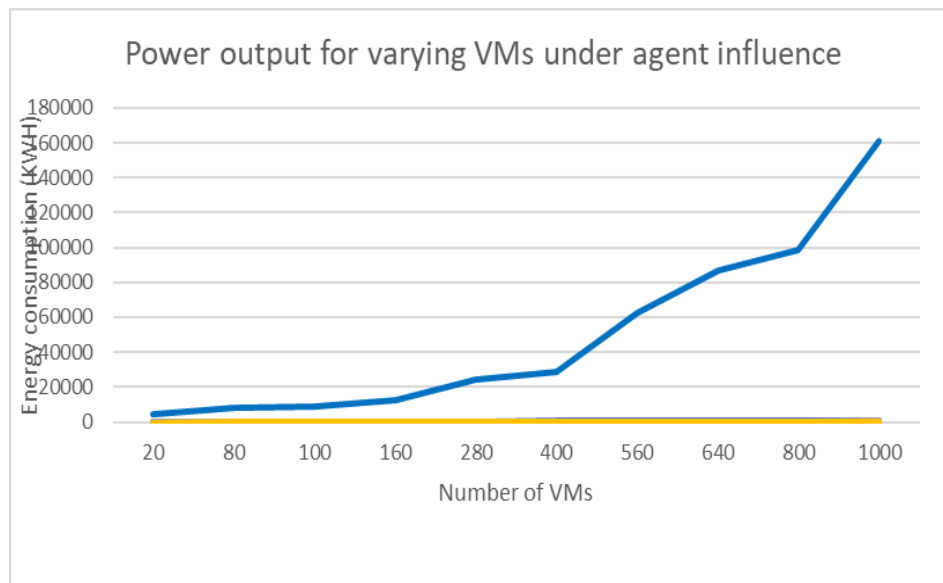


Figure 4.3: Power usage rate during migration

Figure 4.3 depicts the total power usage during the migration time of up to 1000 VMs with no consideration of a lower scale of VMs under different behavioural level. This leads to the second phase of this experiment, where the proposed framework's impact on another VM scenario was conducted and tested. This time with 100 VMs and with a fixed number of hosts (servers) and a variable amount of workload on each VM per time. Figure 4.4 is the outcome of these settings. The blue line is the time-based agent reaction, the red line is the demon-based agent, and the green line is the non-agent impact on the system. From the graphical representation, it is observed that the time-based agent took more time to execute. Still, the host power consumption during time-based regulation is far better than the demon and non-agent scenarios. The time-based agent also did better than the other schemes when there is a constant increase in each VM's load per time. This gain is also evident in cases where workloads were

suddenly migrated to other VMs to maintain adequate balance on the power usage rate and system performance.

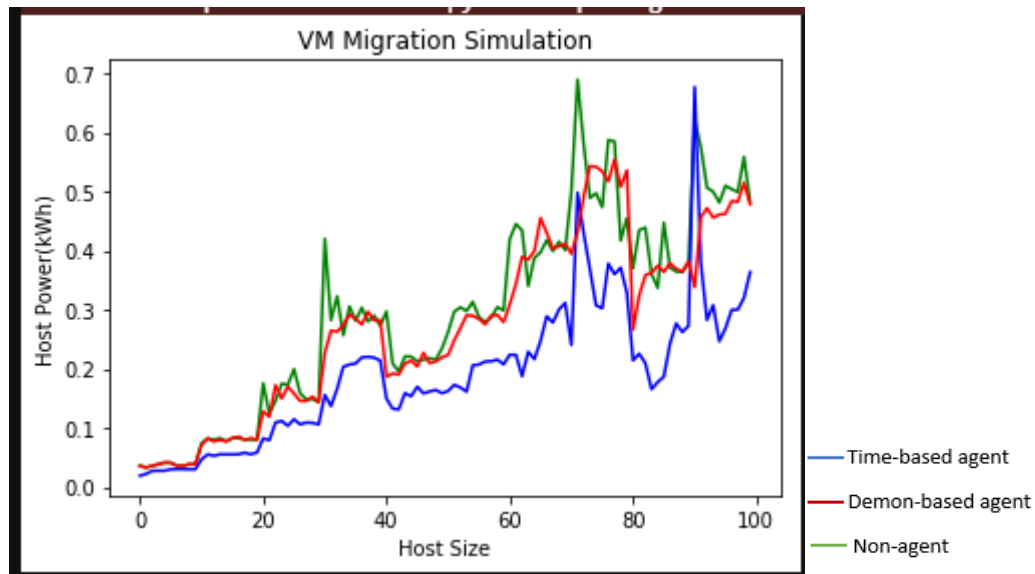


Figure 4.4: Host power simulation behaviour under VM migration

The importance of this research is put to the test based on the system's throughput and performance during VM migration under varying scenarios which is the intention of this research work. Figure 4.4 demonstrates the simulation's overall viewpoint while measuring the amount of energy used by 100 hosts with more than 200 VMs. More than 50 of the VMs were migrated during runtime based on the upper and lower defined threshold embedded on the central broker controller, which loads the agent with the information on their performance rate. The mobile then migrates VMs that are underutilized and shut down the host station while monitoring the active VMs. Hence overutilized VM are also attended to by the mobile agent to ensure they didn't exceed their set threshold to avoid sudden system degradation and delays in workload scheduling. Under staggering load fluctuations due to the dynamic nature of contemporary traffic pattern, the results of both demon and non-agent activities were non-optimal when compared to the time-base agent type. This confirms that the time-based agent gives optimal result because of its dynamic nature to traffic patterns in a time-driven way. It also adapts so rapidly to sudden change no matter its size or level. Most importantly, it intelligently stabilises the network during this process without inhibiting the network's performance.

During this process, the system reliability metric was used to monitor the effectiveness of this mobile process. The result showed that the system performance at all levels during the agent technique was efficiently maximized and running as expected due to the agent's ability to distribute systems adequately. Consequently, the movement of the mobile agent on the cloud data centre did not affect the system's performance at any stage of the migration because of the agent system's intelligent communication and robust migration attribute.

After observing the agent's technology's behavioural pattern in this research work, the process was extended to other existing methods to ascertain the mobile agent process's reliability on other platforms. The graphical result is shown in Figure 4.5, which proves that the mobile agent's code threshold algorithm performs much better than other existing methods under the same parameters and conditions.

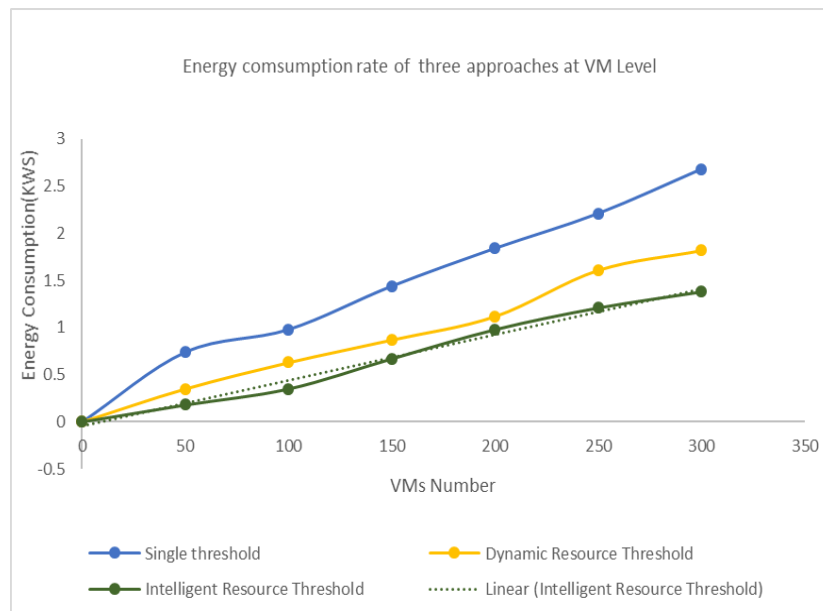


Figure 4.5: Comparison analysis of three different VM threshold

Evaluating the proposed algorithm's impact with other existing algorithms used in Cloudsim 3.3.0 environment, four policies that used different threshold conditions were selected for this evaluation, keeping the same parameters and power threshold to ascertain the variations based on minimizing power usage. These policies were single threshold, dynamic resource threshold, intelligent resource threshold and the linear curve used to evaluate the impact of the intelligent resource threshold under agent operation. Observing from the plot in figure 4.5 intelligent resource threshold performed better than other policies. This is attributed to the ability of the

agent to communicate autonomously with the network at all point to monitor and manage system activities promptly.

Furthermore, this research work evaluated its technique by comparing agent policy to other existing approaches based on study patterns and similarities. These policies are Linear Regression (LR) policy, Threshold Ratio (THR), Mean Absolute Deviation (MAD) and Interquartile Range (IQR). Observing Figure 4.6 graphical representation shows that the mobile Intelligent Agent (I.Agent) approach uses less power than other policies. The result shows that their actual policies have some hidden distribution variables that were not open to the public. However, the available data from these four selected policies were significantly coherent across services which is the reason behind this choice to enhance consistency.

Figure 4.6 also depicts the level of energy reduction by each policy. Thus, the mobile agent's policy saves up to 26.9% with desirable system performance during the experiment for the overall saving. Moreover, on singular observation, it achieves up to 68.9% power saving when compared to 17.4%, 14.6%, 21.3%, and 24.1% power consumption savings from LR, THR, MAD, and IQR policies, respectively.

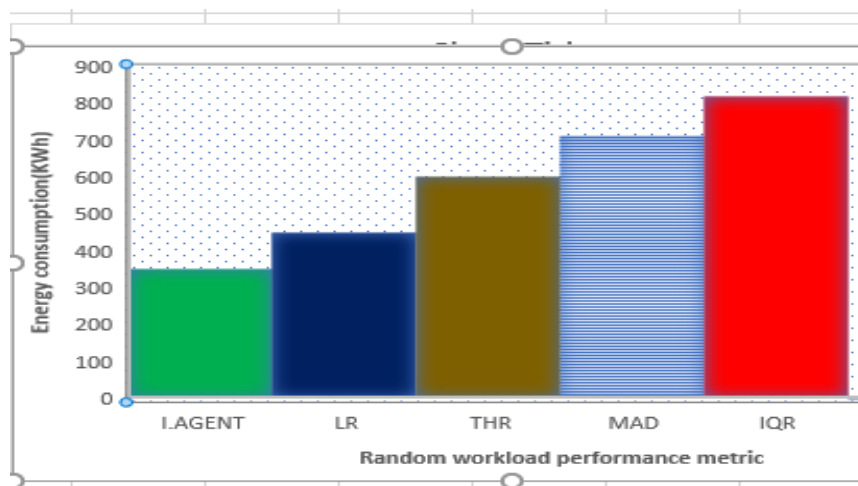


Figure 4.6: Comparing agent policy to other policies in terms of energy efficiency metric

4.3 Server and switch Performance Evaluation

In this section, the mobile agent's results of various simulation outputs were evaluated for both time- and daemon- base. The observed simulation results are based on five set goals, which are to observe the agent's behaviour on the components of the switches and servers, test the

effectiveness of implementing agent technique on the cloud system with the proposed mobile agent called Java agent, check the number of energy-savings from the agent technique at each triggered scenario, compare the performance of agent approach to other existing methods to validate its performance strength and finally, to observe the impact of the agent on the entire system when it is deployed based on the system reliability test metric.

In this phase of the simulation results, the time-based, demon-based and no agent strategy were deployed simultaneously with various hosts, VMs and cloudlets. Table 4.2 shows the different data centres' configuration used for the number of hosts, VMs and Cloudlets, and the total power consumption used by each agent type.

Table 4.2: Data configuration component and its power output on different agent scenario

Number of hosts	Number of VMs	Number of cloudlets	No agents (kWh)	Time-based (kWh)	Daemon based (kWh)
40	80	200	6806.72	6267.80	5919.80
80	160	400	12639.82	11879.62	11167.57
160	320	800	22464.54	18918.70	17248.26
200	400	1000	25416.96	23693.75	24675.75
500	1000	2000	75950.70	74635.53	98537.98
1000	2000	4000	151277.62	154017.18	160453.17

From these results, it is clear that the agent technique models a better system with reduced power consumption when compared to a system with no agent monitoring and regulation of resources. The power consumption used by both the time-base and daemon-base agents is significantly lower than that of the system with a no-agent approach by more than 25% efficiency. These simulations were conducted under different time intervals (ranging from 5 seconds, 20 seconds, 40 seconds, 60 seconds and 100 seconds) to obtain a proof-of-concept dataset that both the time-based and the daemon-based agent methods significantly reduce the power consumption within a cloud data centre network. Table 4.3 shows the host's power consumption rate and switch as used by the three different approaches.

Table 4.3: Total and split host and switch power consumption

Number of Hosts	40		
Number of VMs	80		
Number of Cloudlets	200		
	No Agent	Time Based	Daemon Based
Host Consumption	5388.764	3870.679	4503.655614
Switch Consumption	1817.956426	1532.414558	1416.144814
Total Consumption	7206.720543	5403.093618	5919.800427

Figure 4.7 shows the resultant simulation display of the two different agents' power consumption levels: time-based and daemon-based agents. The figure shows a consistent argument that the data centre power consumption level constantly increases when both resource and workload size increases and adds pressure to the electric system supply.

Power Consumption of Different Agent Sizes for TimeBased and DaemonBased Agent Types

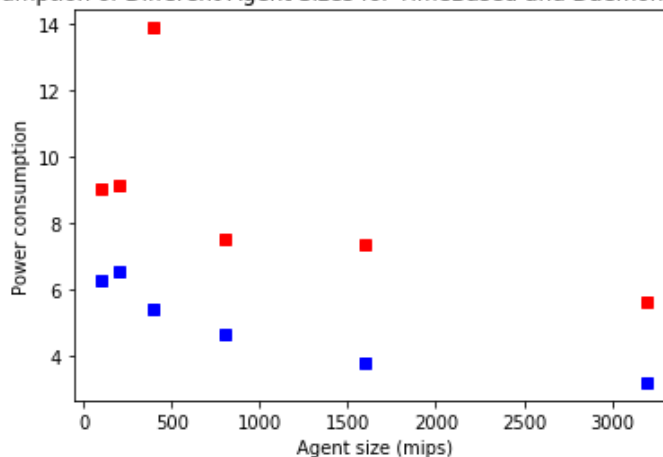


Figure 4.7: Resultant display of the power consumption level of the two agent types

From the graphical representation of Figure 4.8, it is evident that the mobile agent has a crucial part to play in reducing the energy consumption rate in the cloud data centre system. Each scenario, both on small- and large-scale workload, is replicated on the same power-saving strength proving that it is the solution of the data centre's complexity. Figure 4.8 models the power consumption level of the server, switch and the entire system. Figure 4.8 explicitly displays the link in the performance of the three scenarios with optimal power usage. Figure 4.9 is the graphical representation of the resulting output for the power consumption rate of the non-agent and agent approach in the cloud network at the interval of 10 seconds.

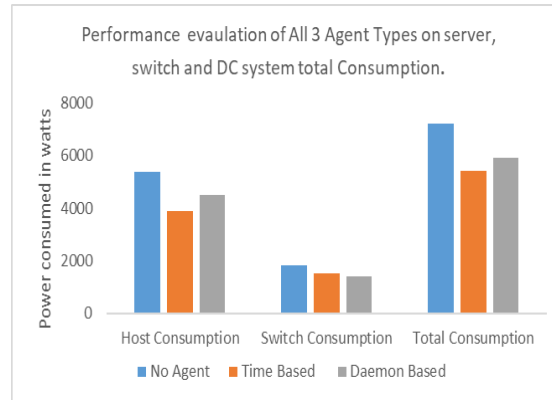


Figure 4.8: Performance evaluation of all three agent types on the server

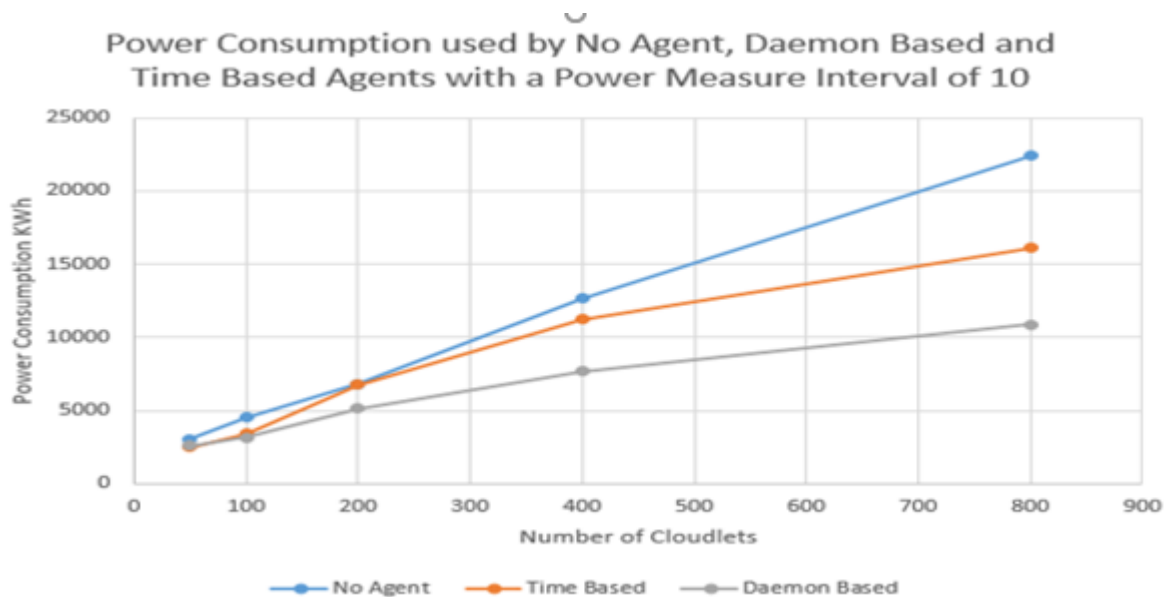


Figure 4.9: Shows the power consumption of all the different agent and non-agent scenario

The behavioural effect of this process is to ascertain which stage the agent technique should be used, and from Figure 4.9, it is clear that it can be used on every part of the data centre. This is because its impact on the system helps the system leverage the interactive nature of the agent's approach - it rapidly picks up the new threats existing in the system with its mobility, learnability and autonomous attributes. Figure 4.10 compares the traditional data centre power usage rate with the mobile agent-based data centre.

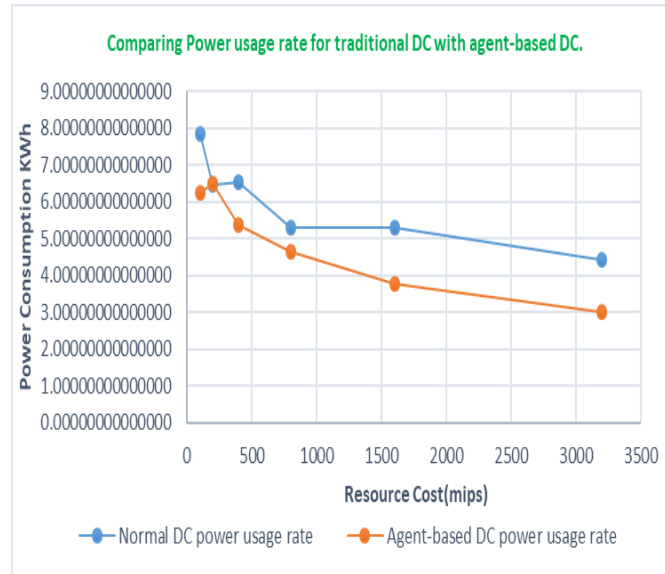


Figure 4.10: Compares the performances of traditional DC power usage rate with Mobile agent-based DC

Continuing with evaluating the impact of the agent-based approach on different components, Figure 4.8 shows the research findings using a traditional data centre with original manufacturer fitted components that are statically operated. From the observation, it is evident that the power usage is higher at every stage, even with minimal static manipulation, which has a drastic system degradation impact on the system, which encourages system violations to a great extent. From the discussion of the simulation results shown in all the figures, there is a significant improvement in each agent type and no-agent present output performance. This indicates that the presence of an intelligent agent within a cloud data centre could theoretically improve the system's performance with less obstruction while using minimal energy in a given unit time. Figure 4.11 compares the mobile agent policy with other existing approaches.

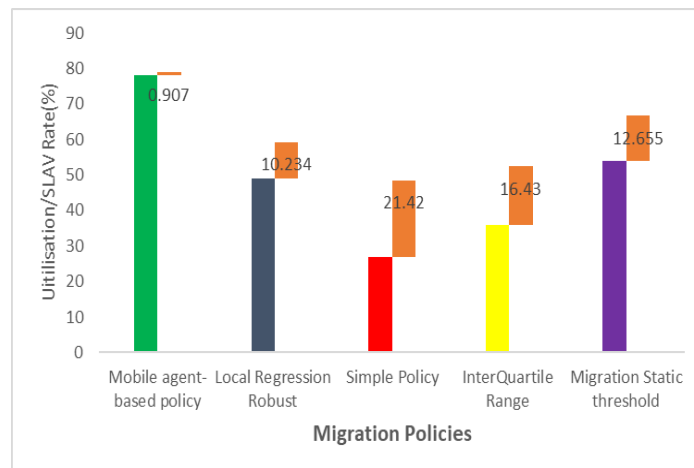


Figure 4.11: Comparing mobile agent policy to other existing policies

This research further validates the mobile agent approach's reliability and resiliency by comparing the findings with other existing approaches. Some of the works on cloud energy efficiency did not consider the system's switching power consumption impact. Therefore, it is tricky to compare. However, some policies that have a generic effect on the overall system's performance is used to model the research findings. DVFS method has always been the stepping block for testing a new technique, which led to the choice of the policies for evaluation. From the results in Figures 4.8, the data obtained has proven that an agent's use increases the system's agility and reduces the operational cost of maintaining a data centre network. Figure 4.8 clearly shows that the mobile agent's performance in the network is outstanding, with insignificant service violation of the system. It is now worth stating that based on different dimensions, it is observed that mobile agent suitability on a cloud environment can no longer be overlooked because of its dynamic adaptive nature on both the deterministic and non - deterministic environment. The graphical representation of Figure 4.8 shows that at all levels, the mobile agent can minimise energy efficiently without disrupting the performance of the system, which is vital when considering the quality of service the providers provides to their client even in their quest to minimise power usage. Comparing this research work with other existing research works shows that this research work gives a more promising result and a more efficient way of operating the cloud data centre network.

4.4 Cost of managing cloud data centre's system using mobile agent technique

The overall cost of running a data centre with an agent approach shows a significant system's improvement and minimal power usage rate compared to other methods. Observing from the simulation's obtained parameters, it is pertinent that the mobile agent is deployed in the system at every point. As a result, it saves up to 35% of electricity usage, which directly reduces the operational cost of managing data centres and thereby aiding the sustainability of business values. Furthermore, the system's throughput at all scenarios with and without an agent is also considered. As each mobile agent aims to reduce the total power consumption used at a given cloud data centre setting during runtime, the data centre's throughput must be unaffected to avoid system performance degradation.

Mobile agent's impact on the cloud data centre has resulted in a drastical increase in system agility since the running cost of deploying an agent-based system operation is significantly lower while still minimising power usage rate. Therefore, the throughput based on the number of executed cloudlets was measured and calculated to show the efficiency of using an agent approach in satisfying users' request using this formulation:

$$C = \sum_{ci \in C} X_c \quad (1)$$

where X_c is $\begin{cases} 1, & \text{cloudlet } c \text{ just finished execution} \\ 0, & \text{otherwise} \end{cases}$

ci = cloudlet from the list of cloudlets.

C = List of cloudlets

Figure 4.12 depicts the system's throughput when the network performs its scheduling duties with no agent and has a demon and time-based agent. At all points, the two mobile agent approaches outperformed the non-agent method. From the figure, it can be observed that at some points, the time-based and demon based agent interweaved and deteriorated. This can be associated with an agent operation mechanism that is driven by event or time. Is it worth noting that the agent communication mode can vary based on events due to its communication means?

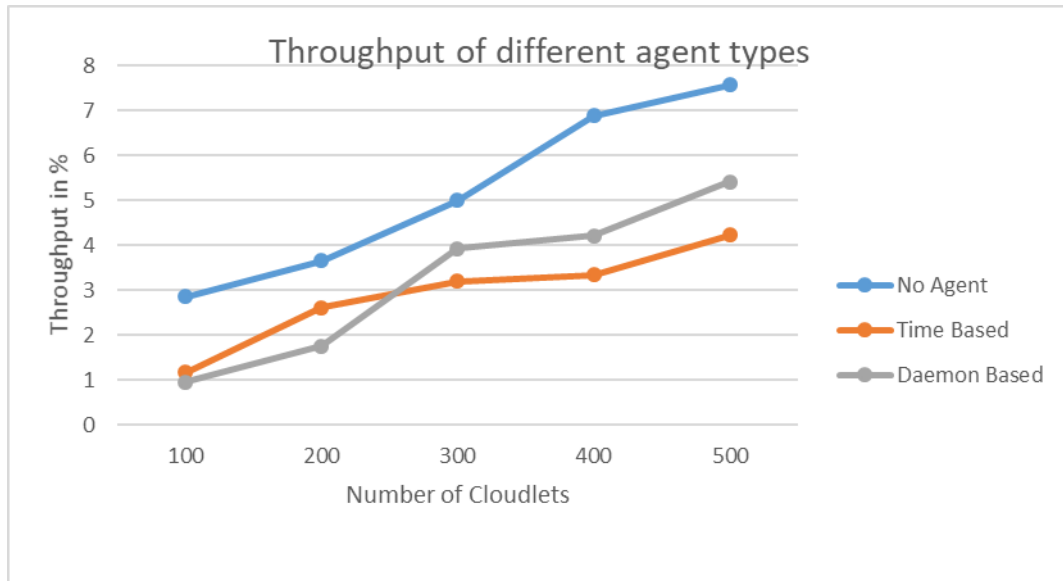


Figure 4.12: The throughput percentage Performance of the system at all stages

The agent transfer protocol is often the mobile agents' communication channel to send messages through the network and then transport the end-users requested tasks. This process can then trigger spontaneous action on the platform, which is observed from the throughput performance graph in Figure 4.12. However, mobile agent techniques come with flexibility, allowing the agent code to gradually collapse or kill itself when activities on the network and its components respond are not spontaneous with the network event.

4.5 System Performance Reliability test.

In this research, great attention was paid to the proposed approach's impact on the cloud's network transition platform. This is because any technique that considers progress or optimal performance status without directly comparing its approach with its system reactions during the process can never be described as best-fit. Therefore, system performance is based on the quality of service rendered to clients at each stage of the system network transition from a task's arrival to when it is responded to and then sent back to the end-user. Hence, end-users quality of service is critical to validate their satisfaction level with the provider's service. The SR in this work is calculated as the number of rejected VMs and servers due to the mobile agent's inability to manage its allocation, migration intelligently, prompt shutting down with no effect on the existing process during the triggered action system monitoring. Table 4.4 shows all the system violation occurrence and when compared with other research works. At some point in running the simulations, the relationship between some variables associated with the network could not trigger an obvious system's violation because of how insignificant some were. The table also shows the SR violation level under varying workloads for different VM policies during the migration and placement process. Different ranges of workloads were used to enhance result output reliability.

Table 4.4: Average (SR) violation encounter by a different technique in the real workload

Data	I.Agent(%)	THR(%)	MAD(%)	IQR(%)	LR(%)
03/03/2011	5.32	10.14	10.28	9.78	6.90
06/03/2011	4.56	10.13	11.06	10.05	9.23
09/03/2011	4.23	10.25	10.35	10.56	10.14
22/03/2011	4.12	10.86	11.87	10.34	9.87
25/03/2011	3.18	12.08	10.34	11.02	10.23

03/04/2011	4.76	10.11	10.29	10.05	10.02
09/04/2011	3.98	10.25	10.12	10.38	10.52
11/04/2011	5.14	12.09	11.04	10.90	11.12
12/04/2011	6.39	11.23	10.45	11.27	11.05
20/04/2011	3.52	10.74	10.89	11.02	11.28

Based on the findings from different scenarios, it was clear that at different phases of the simulation work, the system performance reliability was measured to avoid ambiguity and vague assumptions. As a result, there is an insignificant system degradation at each phase, which is very encouraging based on the previous research assumption that triggering agent technology in the cloud environment may interrupt the system communication flow and increase the power consumption rate. Unfortunately, the reverse is the case, as shown in all the presented results. The mobile agent has minimal or no violation of the system's performance at every stage because it is a lightweight code with high-level intelligence. The result from Table 4.4 indicates that the mobile agent approach has an optimal value with a reduced system downtime violation in percentage both on random and real-time workload. It is worth noting that violation cannot be avoided entirely, but its ability not to affect end-user service is critical and sensitive.

4.6 Summary

This chapter discusses the simulation results from the proposed framework and then compares it with other existing methods. The main focus, which is to save energy while actively transacting requests, is achieved. Observing the figures and tables presented in this chapter, it is inevitable that this approach could save a substantial amount of energy while still maintaining a high level of system reliability. The agent model introduced into the system can deactivate, activate, migrate, consolidate, and monitor the system's performance at every given time. Agent lightweight feature automatically makes its presence in the data centre system with little or no disruption while performing its routine transactions. The mobile agent intelligently deals with sudden system overload, which is prevalent in an online system due to customer demand's unpredictability. It is also able to deal with the underutilisation of resources in the switches, VMs and servers. The simulations based on real trace show that the mobile agent approach can reduce at least 32% energy consumption rate at every sampled scenario more

than non-agent baselines. Hence, it validates the feasibility, scalability, and reliability of using agent-based approaches on the cloud data centre environment to minimise energy usage rate while maintaining a standard quality-of-service.

Chapter5

Semantic Knowledge Representation

5.1 Introduction

In this chapter, the importance of using semantic knowledge representation to graphically demonstrate research contributions and findings on agent-based approaches for an energy-efficient data centre is presented. Semantic ontology describes and relates the links between the complex state of cloud data centre and mobile agent phenomena. Ontology semantic web is then used to represent the research defined questions, research outcome, correlate the importance of having mobile agent operation in cloud data centre network and the research working experience to extend the existing practices of managing cloud data centre energy-related challenges.

Semantic knowledge helped this research work capture all the necessary stages during this study's formulation and practical stages.

It brought insight into the components and factors affected more by the energy consumption level of the data centre and enhances the presentation of the mathematical model used in this work.

5.2 Overview of Semantic Ontology

By definition, a semantic ontology is an act of presenting descriptive knowledge as a set of concepts within a domain and the connection that links them together. It models a complex idea into a more simplified context. Many industries and research groups choose different semantic knowledge representation methods, ranging from Unified Modelling Language (UML), Web Ontology Language (OWL), Graphic, etc. In this research work, a semantic protege ontology is used because of its ability to display a frame's core values in an object-oriented manner. Furthermore, it has an open architecture that allows other modelling platforms to be built and supports other plugins development, allowing back-end and interface extensions where need be. Therefore, the semantic protege ontology knowledge representation tool displays a direct relationship and link between the cloud data centre network. Its impending

challenges and agent technology provisions can leverage the high-power usage in the cloud data centre network.

5.3 Related Work

This research work discusses the related work base on the general usage of semantic knowledge representation using protege 5.0. Though this a computing software tool, not many research works have been modelled using it. This research work is the first to use this novel approach to modelling hybrid problems. This work used to model the system processes, research contributions, and the trend of future sustainable cloud system and their connectivity to the benefits of intelligent agent technology on leveraging network complexity and energy efficiency. This approach shows a pretty well design frame of our research aims and displays the facts based on applications' usability and agent characteristics acceptability in a cloud environment.

In 2009, the authors (Ling Zeng, Zhu, Xin and Xiaodong) conducted a research study using semantic ontology to display the importance of continuing education. Their work modelled the university system's need to have a pretty looking modelling tool that appeals to old learners and easy to search course details based on their personal needs and constraints. This work is useful to date, which has encouraged its application to this research work. In 2010, the authors (Simona Elena and Varlan) performed theoretical research work on the impact of using the semantic knowledge ontology in every aspect of computing design and implementation to enhance project continuity. This work is useful but fails to ascertain practical validity. In 2012, the authors presented (Tramp, Frischmuth, Ermilov and Shekarpour) research using the semantic web tool to design an architectural platform for the social network. They used the combination of vocabularies and protocols to showcase a coherent, intelligently distributed semantic social network with more functional capacity and thus connected to a centralised source for easy accessibility. Therefore, the existing literature shows that the semantic web knowledge representation using protege has not seen the spotlight on many computing fields. This, therefore, underscores that this work is the first to use protege to model complex ideas on the cloud network and its energy efficiency-related issues.

5.4 Semantic knowledge representation of power management techniques

Semantic knowledge representation OWL is used to present the existing power management technique for clarity purposes and validate that mobile agent technology has never been used in the cloud's energy efficiency-related research. Figure 5.1 explicitly displays the power management approaches that have been previously used in the area of power minimisation, which was discussed in Chapter 2. This research work evaluates each management technique's impact, limitations, and usability rate based on an accepted report by academia or industry experts. Therefore, presenting these techniques again using the semantic ontology graphical display, not a technical context, gives people from other fields a sense of value and understanding to the aim, research questions, limitations and contributions of this research work. Figures 5.1, 5.2 and 5.3 give a characterised understanding and predicted behaviour of previous methods and current mobile agent's method. It thereafter gives in-depth link connectivity of mobile agent attributes to different cloud data centre network components both for the critical and non-critical components. More so, it shows a clear understanding and display of mobile agent's usability in the cloud system platform with no significant violation due to the agent system's learnability, flexibility and autonomous mode of communication in the network's communication flow during traffic allocation, migration and scheduling.

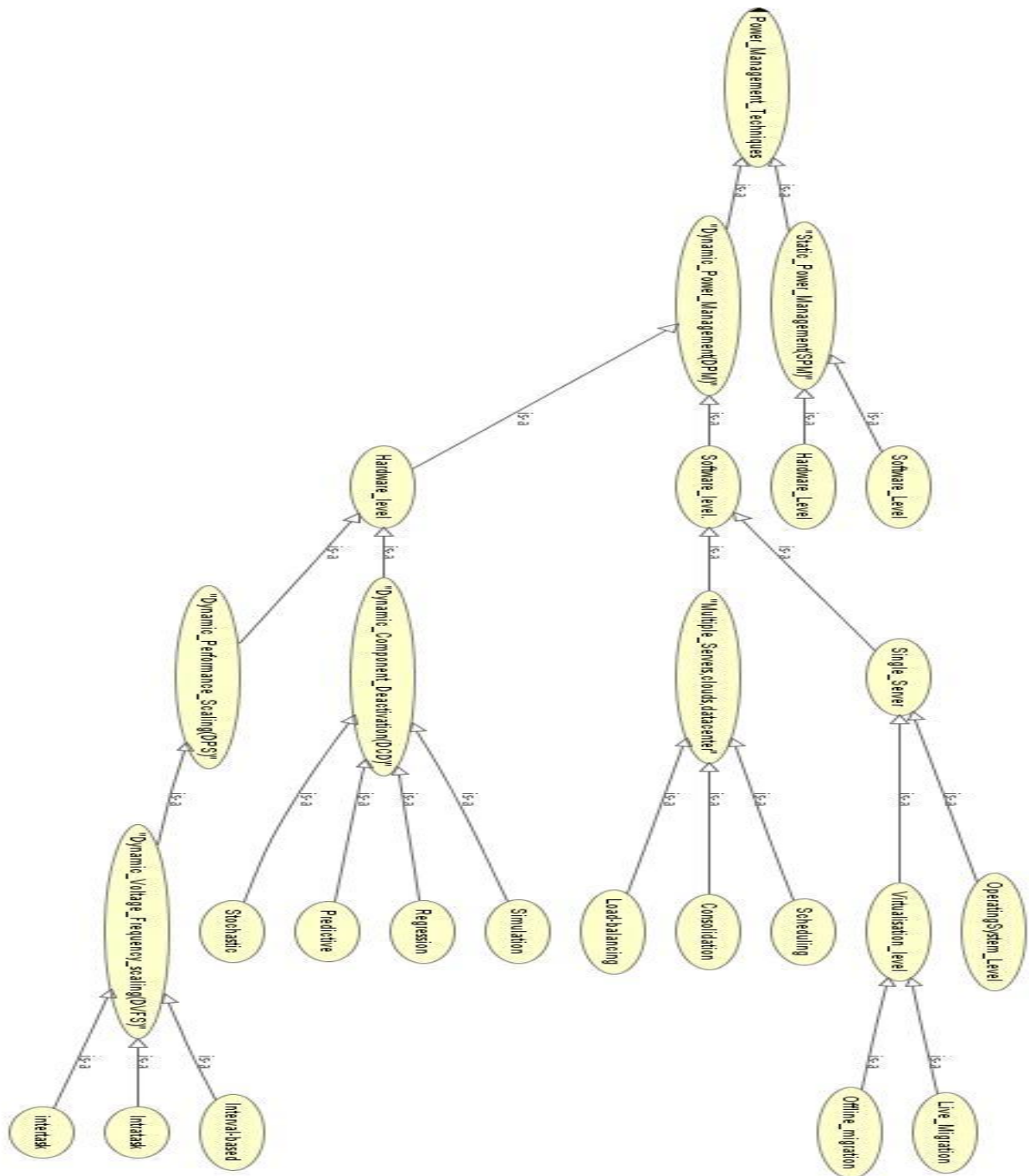


Figure 5.1: Semantic web representation of power management techniques

5.5 Semantic ontology of agent-based system

The semantic web's concept brings into the spotlight the doubtable features of agent technology and then relates it to a complex system; in this case, it connects the agent to the cloud environment. Finally, the protege software is used to design and develop a robust, intelligent, educative, and scalable but simplify looking agent technology with its links, connectives and various characteristics which can be leveraged by cloud network and any other distributed networks. Figure 5.2 shows a consolidated research approach functionality in this research work, and the discussion explicitly elucidates the usefulness of the agent-based system classification.

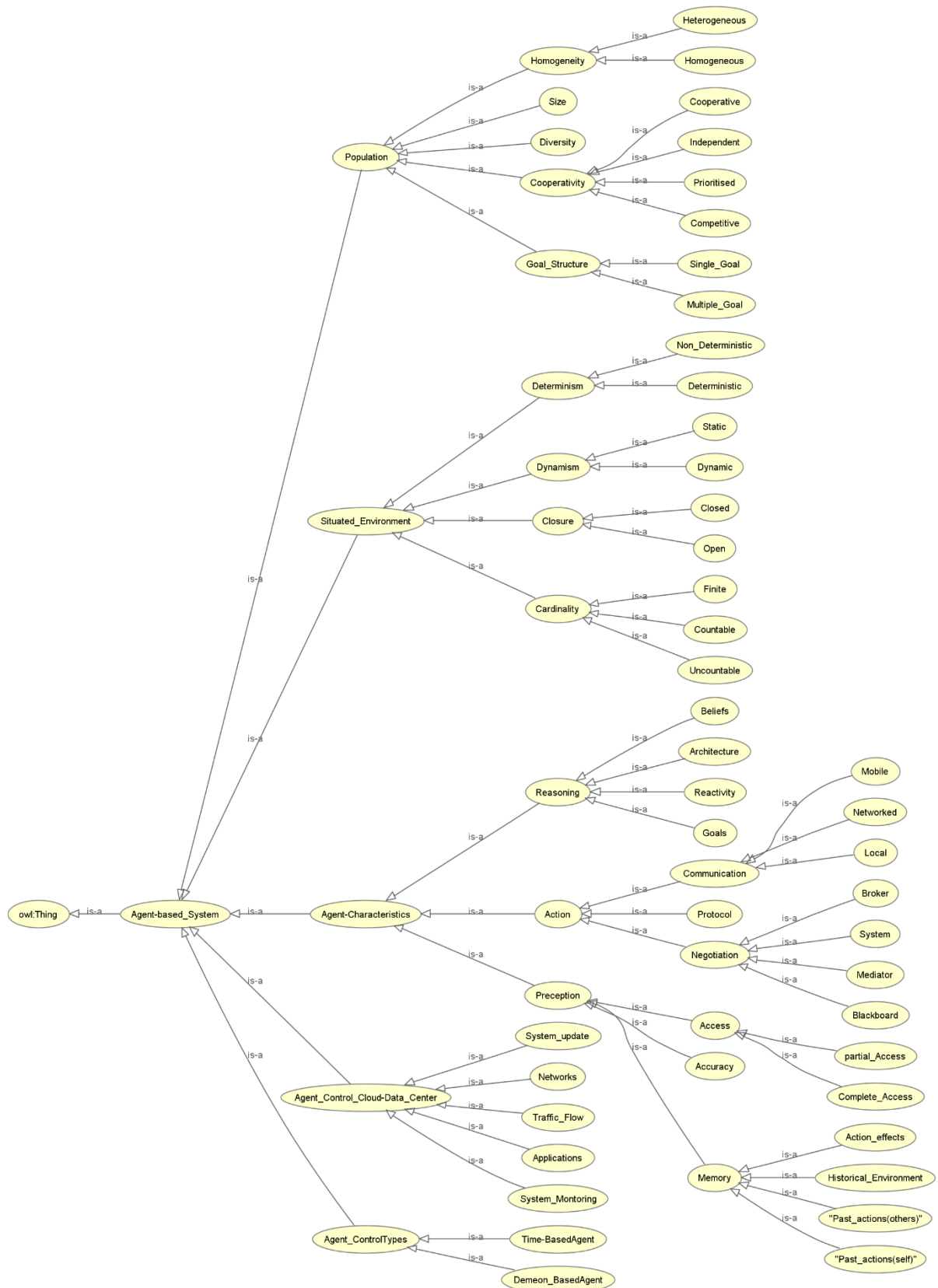


Figure 5.2: Taxonomy of agent-based system

5.6 Semantic knowledge representation of an agent-based system with cloud data centre solutions

The literature review in Chapter 2 of this thesis work explains the characteristics of the agent-based system. However, the connection of these characteristics can sometimes be vague, challenging and complex to connect the workability of the agent technology and cloud complex network. Therefore, with the aid of ontology, protégé produced the semantic connection between the agent characteristic and the cloud data centre network, and then the corresponding communications channels where the agent works. Finally, the agent characteristics are classified into three major sets (action, reasoning and perception) and link to their associate subsets. This process helps users to understand and trigger the right actions with the right tool.

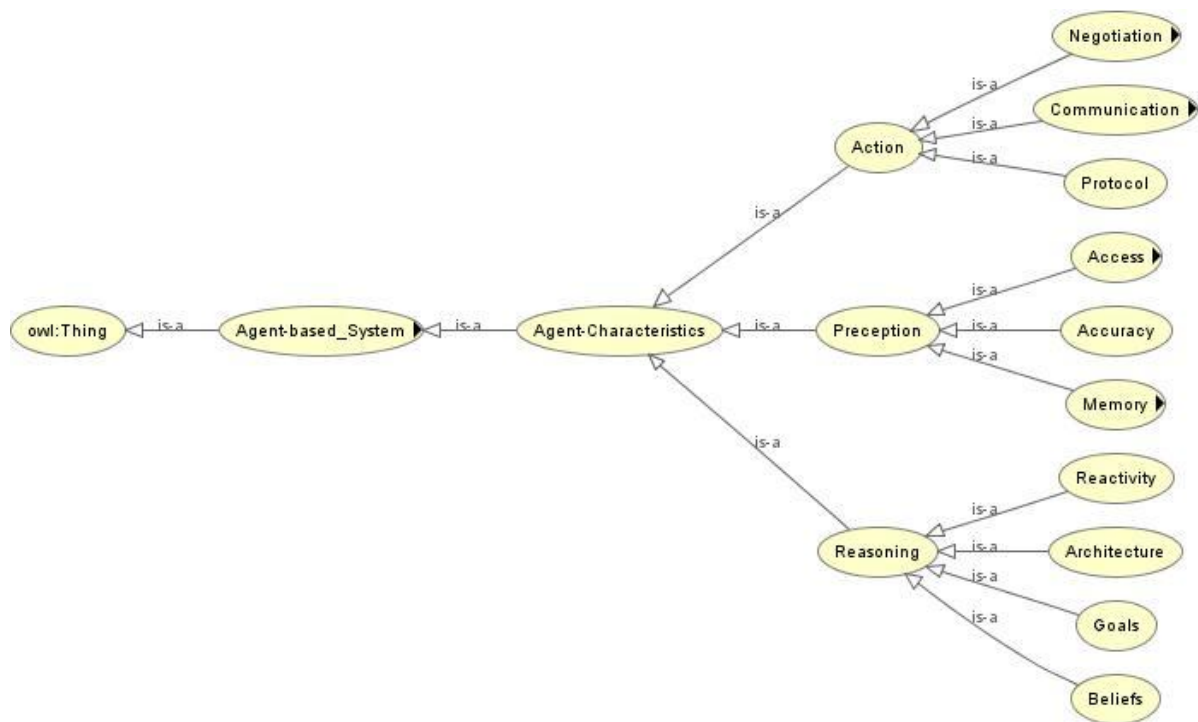


Figure 5.3: Semantic display of agent characteristic

At this point, an explicit model of agent characteristic has been achieved. This then builds up to using this same approach to consolidating the research problem with finding a workable solution to the clouds' data centre high energy consumption. The semantic ontology is also used to model the cloud data centres' components. This step will help providers and data centre engineers understand the data centres' various operational levels and their level of

contributions to energy-related challenges in the cloud data centre. This brings us to present Figure 5.4.

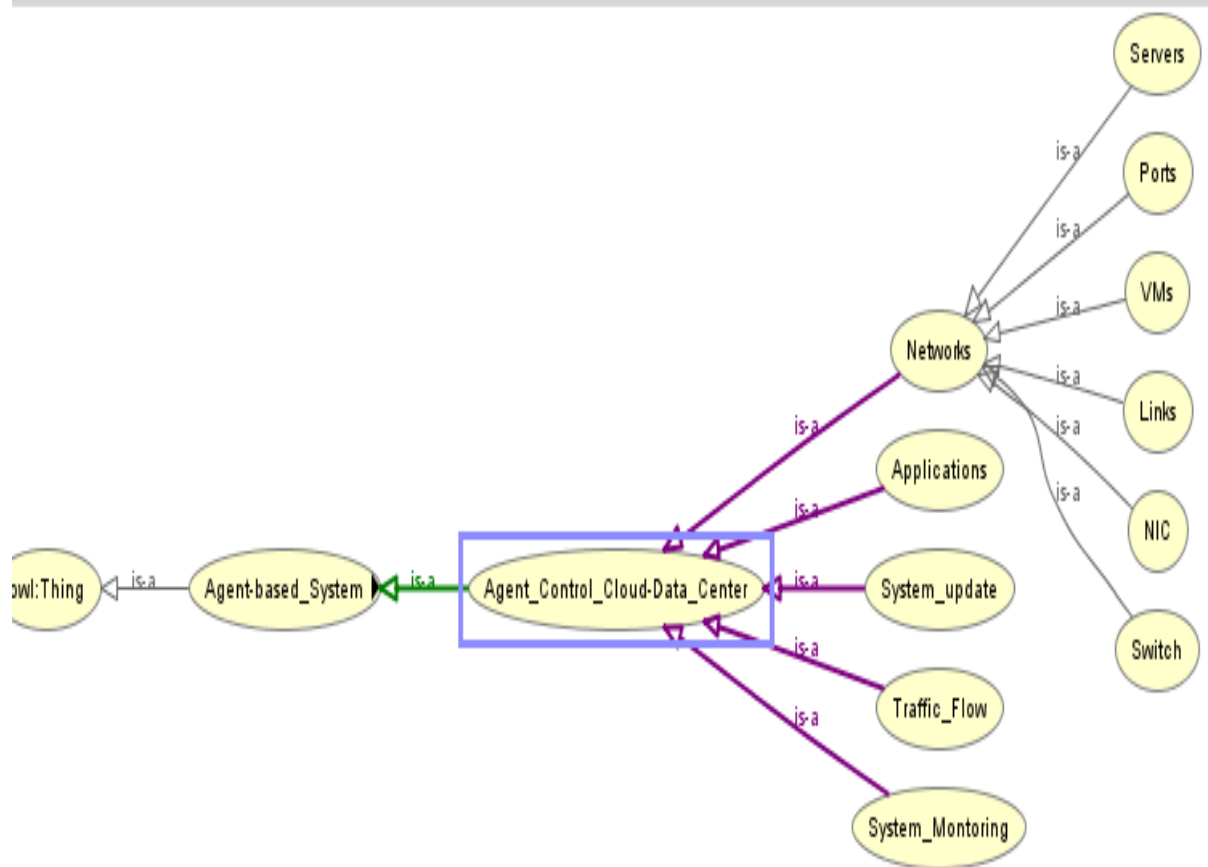


Figure 5.4: Cloud data centre components under agent influence

Figure 5.4 represents all the cloud data centre components that can be maintained and managed by the agent's technology on a homogenous or heterogeneous system with their associate sub-systems. Figure 5.5 presents the different agent types used in this research work.

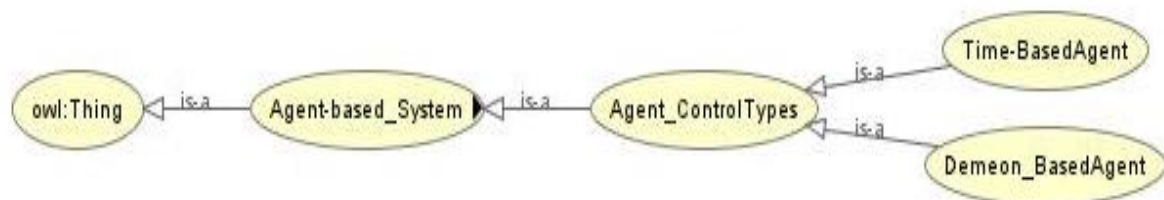


Figure 5.5: Agent-types used in this research work

In this research, two types of mobile agents are used in different scenarios with different network components and varying workload. These two agents were used to validate the effect of a mobile agent's system in a cloud environment under complex operational mode.

5.7 Summary

Based on the semantic display, it can easily be noticed that the agent technique is the core to solving the complex-related issue of a cloud data centre. The data centre's high energy consumption level cannot just be managed using the existing static and dynamic approaches as shown in power management available semantic in Figure 5.1. The intelligent behaviour of the agent is based on its inherited characteristic. Figure 5.3 shows that the mobile agent technology is a better technique for minimising power usage rate based on the experiment conducted during this research work and in a real-time test system when a mobile agent code performance is observed. In this work, it can be seen in Figure 5.4 the two types of agents used. However, mobile agents are not limited to these two types, but each enterprise should work out the type they want to use in their system based on their business requirement model. Therefore, mobile agent technology has proven reliable, scalable, robust, and resilient from all observed angles while still maintaining a minimal energy usage level during system processing and run-time.

Chapter6

Conclusion and Future Work

6.1 Introduction

This chapter summarises research contributions on agent-based approaches for an energy-efficient data centre. In the section, we will summarise the research outcome and the research working experience trying to extend the existing practices of managing cloud data centre energy-related challenges. We encountered some obstacles during this work that will now be the future direction for upcoming research works in this field.

6.2 Summary and Contributions

The irresistible benefits of using cloud technology can be in high jeopardy if the challenge of the energy consumption rate it faces continues without a workable permanent solution. The cloud vendors such as Amazon, Google, Salesforce, IBM, Microsoft has always used data centre as the backbone for their cloud activities and applications globally. Vendors always aim to satisfy customers' needs by ensuring they receive a flexible, accurate, suitable and timely service on request. Unfortunately, the challenge of high energy consumption rate threatens the vendors promise to the customers. The data centre can consume high energy level as much as the power consumed by 25,000 households in the US, according to (Darathna et al.,2014). Compared to developing countries with significantly lower energy generation level, it is better not to imagine the demand the high energy consumption can cause in such areas. Furthermore, the rapid growth of the use of the Internet, smart device, the need to use different complex data-driven applications, and sudden switch to the remote working environment caused by the pandemic has increased the data centre's activities heightens its energy consumption level.

Therefore, it is very expedient that cloud vendors find a soluble pattern to minimise the cloud data centres' energy consumption rate to improve its efficiency, system reliability and finally reduce the operational cost of running the data centre.

Original research has suggested VM consolidation and Dynamic Voltage Frequency Scaling (DVFS) to solve this energy consumption challenge. However, the solution provided by this approach was minimal and lacks efficiency, especially when there is a sudden surge on the system with a high level of system overload.

From Literature, we have stated in previous chapters that inefficient resource utilisation causes a major cause of cloud data centre high energy consumption rate by application on server and switches. To utilise the data centre infrastructure resources efficiently, data centre components should be observed to have an explicit understanding of its operational capacity, encapsulate its logic, and monitor its interface functionality to ensure flexible manipulation and migration of components based on its performance rate.

This research work is the first research documented work to have used the mobile agent approach in the cloud computing-related field and also to use its features to leverage energy efficiency-related challenges in the cloud data centre. As we have previously pointed out, the agent technology was used in 2012 for the wired and wireless network during traffic regulation and sensor monitoring. The agent-based technique was used to shut down inactive switched and server and migrate VMs during their underutilised phase. In this thesis, we have conducted a rigorous literature search through a survey, proposed the agent-based approach to fill the literature gap, implemented the prototype of our finding through simulation, and monitored the activities of the data centre under agent controller traditional design settings.

Chapter 1 gave a background understanding of cloud computing, its enabling technologies, its challenges and why an intelligent agent system should be welcomed. We then stated the objective and the contribution of our work to the literature and research world. Chapter 2 was mainly a literature review on existing literature based on energy efficiency techniques, approach and the ways it was implemented. Different taxonomy based on this research work was produced to prove the significance of the research which was classified as followed (a)cloud data centre design, (b) workload scheduling (c) VM migration and system maintenance and (d) monitoring - discussing their advantages and disadvantages.

Chapter 3 discussed the methodology adopted during the process of investigating the impact of using an agent-based approach on cloud data centre network.

Chapter4 was the experimental result outcome. This work also compared and contrasted the agent approach result against other existing approaches in this chapter to validate the proposed findings.

6.3 Future Work

Although there was significant progress made upon applying agent-based technology to cloud data centre networks with substantially achieved deliverables. This agent proposed method delivered an intelligent adaptive management technique for resources and applications handling to reduce high energy usage with significant improvement; however, there are still research gaps and challenges in the energy-related area to be further explored for cloud technology sustainability. This section provides some insights into some research gaps and promising future directions to broaden the scenarios of agent-based techniques to mitigate the challenges and also improve the management of resources in the cloud data centres networks.

6.3.1 Managing different resource types

Finding an adequate management technique for multiple data centre resources coordination will enhance the cloud environment's effectiveness and resource efficiency because it promotes holistic optimal resource usage. For instance, the mobile agent technology has just been used to improve the server, switches and VMs power usage level during runtime with great success. Therefore, this approach can also be extended to memory, storage, and cloud registry handling in the upcoming works to obtain a more precise decision on energy-hungry factors and better ways to maintain them if it can't be removed.

6.3.2 Developing a robust optimisation technique for cloud data centre scenarios.

From observation, the optimisation technique has always been a way toward complex calculated situations that cloud energy-related challenges fall into the category. Therefore, investigating more diverse optimisation aid goals could enhance the applicability of the mobile agent-based technique. Agent-based can be applied to optimisation goals such as evaluating load balancing, complicated cost-aware resource scheduling scenarios, and more QoS metrics can be applied with a mobile agent. Secondly, the power consumption of the data centre can

be optimised by redesigning the entire data centre. The optimisation process will use some aspect of the genetic algorithm due to its multivariable attribute, which will shuttle the servers' architecture and the settings of the VMs for a more adaptive and intelligent way of operation. Finally, the optimisation technique can ensure a sustainable cloud infrastructure with reduced carbon emissions for green computing.

6.3.3 Combination of prediction using spike neural network with agent technique

Neural network can be used to predict past occurrences of switches, hosts or servers that frequently consume high levels of energy during runtime; hence the integration of agent technology with the neural network will empower the agent with a more active historical feature to operate with. Therefore, through the historical report, the agent ensures the same pattern does not often repeat by continually monitoring the regions' or equipment performance state. The agent will immediately trigger an alarm on observing any abnormality or upcoming flaws, which will enable the system engineers to prepare for adequate event handling and avoid system sudden disruption.

6.3.4 Machine Learning Techniques can be Explore on the Cloud data centre

In cloud energy-related issues, the machine learning algorithm is yet to be used efficiently to ascertain the best decision time and execution time to trigger agent-based scenario, especially to accurately deactivate and activate data centre components based on the criticality of its operation at a given time. For instance, future work can investigate using the K-mean, which is one of the machine learning algorithms as an addition to mobile agent feature based on response time set thresholds, and the task can be clustered for further execution on the same kind of component for resource usage enhancement

6.3.5 Integration of agent approach with other existing energy-efficient approaches

Since a mobile agent can successfully transport itself on the network in the data centre will be worth exploring integrating the agent technology with other existing approaches to achieving more efficient and effective cloud resource management. It will be reasonable to expect better

results when a mobile agent's method is mixed with other existing energy efficiency techniques, such as software-defined-network (SDN) or energy-aware fuzzy framework. The mixed process will form a hybrid solution for optimising energy consumption from a network and application traffics perspective.

6.3.6 Develop new cost model

Cloud vendors will be keen to have a workable integrated cost model. The already existing cost models need a high level of improvement and should consider more business analytical cost consuming parameters, not a shallow approach with no emphasis on electricity costs, budget, cooling costs and carbon emissions. Based on the cost models, the service providers could optimise their resource provisioning and maximise profits. Therefore, an agent-based mechanism can be used to investigate a fine-grained cost model, such as the energy and time costs of the mobile agent activation/deactivation operations and energy consumption of turning on/off servers.

6.3.7 Investigating centralise controlled scheduling

Scalability and the ability to intelligently reduce redundancy should be considered when designing a resource scheduling approach in future work, which will support handling system components power consumption while there are increased number of requests on a global set energy consumption threshold. In this work, a distributed approach was used, where agent-based scenarios come into play once the system energy consumption remains on the range of the set threshold. Therefore, it is worth trying a fixed approach with agent-based scenarios. Once the agent finishes the allocation based on the data centre available power, it automatically deactivates all the non-critical components until there is a refreshed set rule. This will probably reduce the number of vast system redundancies.

6.4 Summary

Cloud computing technology is a business-oriented tool that is attractive; unfortunately, its enabling backbone (data centre) consumes a lot of energy, which is a significant challenge and threatens this striving innovation's sustainability. This thesis will explore using an agent-based

approach to fill in the gaps left by existing methods toward minimising the cloud data centre's energy consumption rate. It proposed the architecture that enabled the agent-based techniques in the cloud data centre, designed an algorithm for minimising energy usage rate while maintaining an acceptable QoS, and finally built from scratch the prototype system used to evaluate agent performance findings in a cloud data centre environment. The research outcome of this thesis produced a promising energy-efficient cloud data centre. The approach can be adopted by cloud service providers and implemented on their operating data centre to reduce the operational cost of running their data centre and increase their system network performance with minimal energy usage strategy. Therefore, seeing the improvement bought by this approach by saving the power consumption rate of the data centre's power consumption rate, future work should investigate other aspects of agents approach not yet implemented.

References

1. A. Bieszczad, B. Pagurek, T. White, “Mobile Agent for Network Management” IEEE communication Surveys,1, 2-9, 1998.
2. A. M. Al-Qawasmeh, S. Pasricha, A. A. Maciejewski, and H. J. Siegel, “Power and thermal-aware workload allocation in heterogeneous data centres,” IEEE Transactions on Computers, vol. 64, no. 2, pp. 477–491, 2015.
3. A. Verma. Ahuja, and A. Neogi, “Mapper: Power and migration cost aware application placement in virtualized systems”, in Proceedings of the 9th ACM/IFIP/USENIX International Conference on Middleware. Springer-Verlag New York, Inc., 2008, pp. 243–264.
4. A. S. Andrae and T. Edler, “On global electricity usage of communication technology: trends to 2030,” Challenges, vol. 6, no. 1, pp. 117–157, 2015.
5. A. Wierman, L. L. Andrew, and A. Tang, “Power-aware speed scaling in processor sharing systems”, in Proceedings of the 28th Conference on Computer Communications (INFOCOM 2009), Rio, Brazil, 2009. 43
6. Alder, J.L., Satapathy, G., Manikonda, V., Bowels, B., Blue, V.j., 2005. “A Multi-agent approach to cooperative traffic management and route guidance”, Transportation Research Part B 39(4), 297-318.
7. Ali Pahlevan, Yasir Mahmood Qureshi and Marina Zapater.” Energy Proportionality in near-threshold Computing Server and Cloud Data Centres: Consolidating or Not? Proceeding, IEEEExplore.ieee.org 2018
8. Andreas Berl1, Erol Gelenbe, Marco di Girolamo, Giovanni Giuliani, Hermann de Meer1, Minh Quan Dang and Kostas Pentikousis, “Energy-Efficient Cloud Computing” published by Oxford University Press on behalf of The British Computer Society. Advance Access publication on August 19, 2009.
9. Arianya E., Tahrir H, Sharifian S, “Novel Energy and SLA efficient resource management heuristics for consolidation of virtual machines in cloud data centres”, Comput Elect Eng. 2015
10. Beloglazov, “Optimal Online Deterministic Algorithm and Adaptive Heuristics for Energy and performance efficient dynamic consolidation of the virtual machine in the cloud Data centre”, Volume 24.

-
11. Brown, R., “Report to congress on server and data centre energy efficiency public law 109-431,” U.S. Environmental Protection Agency, Washington, DC, USA (2007).
 12. C. H. Hwang and A. C. Wu, “A predictive system shutdown method for energy saving of event-driven computation”, *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, vol. 5, no. 2, p. 241, 2000.
 13. C. Pettey, “Gartner estimates ICT industry accounts for 2 percent of global CO₂ emissions”, <http://www.gartner.com/it/page.jsp?id=503867>, 2007.
 14. Cisco, “Cisco Global Cloud Index: Forecast and Methodology,2012-2017”, White paper, 2013.
 15. Chabrol, M., Sarramia, D., Tchernev, N, “Urban traffic system modelling methodology” *International Journal of Production Economics* 99(1-2),156-176, 2006
 16. Cisco.<http://www.cisco.com> Verma A, Ahuja P, NeogiA.pMapper, “Power and migration cost aware application placement in virtualized systems” proceedings of the 9th ACM/IFIP/USENIX international conference on middleware, Springer, Leuven, Belgium, 2008;243-264.
 17. Claudio Fiandrino, DzmitryKliazovich, Pascal Bouvry and Albert Zomaya. “Performance and Energy Efficiency Metric for Communication Systems Of Cloud Computing Data Centers” *IEEE Transactions on Cloud* Vol.5 No.4, 2017.
 18. Cui, J., Liu, S.f, Zeng, B., Xie, N.M, “A novel grey forecasting model and its optimization. *Applied Mathematical Modelling*” 37(6), 4399-4406(2013).
 19. D. Kusic, J. O. Kephart, J. E. Hanson, N. Kandasamy, and G. Jiang, “Power and performance management of virtualized computing environments via lookahead control”, *Cluster Computing*, vol. 12, no. 1, pp. 1–15, 2009.
 20. E. Elnozahy, M. Kistler, and R. Rajamony, “Energy-efficient server clusters, Power-Aware Computer Systems”, pp. 179–197, 2003.
 21. E. Pinheiro, R. Bianchini, E. V. Carrera, and T. Heath, “Load balancing and unbalancing for power and performance in cluster-based systems”, in *Proceedings of the Workshop on Compilers and Operating Systems for Low Power*, 2001, pp. 182–195.
 22. E. Faller, C. Morin, D.Leprince, “State of the Art Saving in Clusters and Results from the EDF Case Study”, INRIARennes- France,2010
 23. F. Alvares, G. Delaval, E. Rutten, and L. Seinturier, “Language support for modular autonomic managers in reconfigurable software components,” in *Proceedings of the*

-
- 2017 IEEE International Conference on Autonomic Computing. IEEE, 2017, pp. 271–278.
24. F. Douglass, P. Krishnan, and B. Bershad, “Adaptive disk spin-down policies for mobile computers”, *Computing Systems*, vol. 8, no. 4, pp. 381–413, 1995.
25. G. Buttazzo, “Scalable applications for energy-aware processors, in *Embedded Software*”, 2002, pp. 153–165. [29] M. Weiser, B. Welch, A. Demers, and S. Shenker, —Scheduling for reduced CPU energy, *Mobile Computing*, pp. 449–471, 1996.
26. Gartner Press Release, “Gartner Estimates ICT Industry Accounts for 2 percent of global CO₂ Emissions”, <http://www.gartner.com/it/page.jsp?id=503867> April 26, 2007
27. Hu B, Lei Z, Lei Y, Xu D, “A time series based pre-copy approach for live migration of virtual machines”, In *Parallel and Distributed System(ICPADS)2011 IEEE 17th international conference on*. IEEE, pp 947-952, 2011
28. Ismaeel S, Miri A, “Using ELM technique to predict data centre VM request”, In *proceeding of the 2nd IEEE international conference on cybersecurity and cloud computing (CS Cloud 2015)*.IEEE, New York,pp80-86, 2015
29. J. R. Lorch and A. J. Smith, “Improving dynamic voltage scaling algorithms with PACE”, *ACM SIGMETRICS Performance Evaluation Review*, vol. 29, no. 1, p. 61, 2001.
30. K. Govil, E. Chan, and H. Wasserman, “Comparing algorithm for dynamic speed-setting of a low-power CPU”, in *Proceedings of the 1st Annual International Conference on Mobile Computing and Networking (MobiCom 2005)*, Berkeley, California, USA, 1995, p. 25.
31. K. Y. Bae, H. S. Jang, and D. K. Sung, “Hourly solar irradiance prediction based on support vector machine and its error analysis”, *IEEE Transactions on Power Systems*, vol. 32, no. 2, pp. 935–945, 2017.
32. Kusic D, Kephart J O, Hanson JE, Kandasamy N, Jiang G., “Power and Performance management in virtualised computing environments via lookahead control”, *Cluster Computing* 2009,12(1):1-15.
33. L. Baresi, S. Guinea, G. Quattrocchi, and D. A. Tamburri, “Microcloud: A container-based solution for efficient resource management in the cloud,” in *Proceeding of the IEEE International Conference on Smart Cloud*. IEEE, 2016, pp. 218–223.

-
34. L. Benini, A. Bogliolo, and G. D. Micheli, "A survey of design techniques for system-level dynamic power management", *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 8, no. 3, pp. 299–316, 2000.
 35. L. L. Andrew, M. Lin, and A. Wierman, "Optimality, fairness, and robustness in speed scaling designs", in *Proceedings of ACM International Conference on Measurement and Modeling of International Computer Systems (SIGMETRICS 2010)*, New York, USA, 2010.
 36. Liu, J., Zhao, F., Liu, X., He, W., "Challenges Towards Elastic Power Management in Internet Data Centres", *Proc 2nd international workshop on cyber-physical systems (WCPS)*, in conjunction with *ICDCS Montreal*, Canada, 2009
 37. Li B., et al., "EnaCloud: An energy-saving application live placement approach for cloud computing environments", *Proc of international conf on cloud computing(2009)*.
 38. Ling Zeng, Tonglin Zhu, Xin Dan and Xiaodong Xu, "Study on Construction of University Course Ontology", in the context of continuing Education, 2009
 39. M. S. Aslanpour, M. Ghobaei-Arani, and A. Nadjaran Toosi, "Auto-scaling web applications in clouds," *Journal of Network Computer Applications*, vol. 95, pp. 26–41, 2017.
 40. M. Avgerinou, P. Bertoldi, and L. Castellazzi, "Trends in data centre energy consumption under the European code of conduct for data centre energy efficiency," *Energies*, vol. 10, no. 10, p. 1470, 2017.
 41. Nathuji R, Schwan K., "Virtualpower: coordinated power Management in virtualized enterprise systems" *ACM SIGOPS Operating Systems Review* 2007;41(6):265-278.
 42. OgechukwuOkonor, Mo Adda, "Intelligent Approach to Minimizing Power Consumption in Cloud-Based System Collecting Sensor data and Monitoring the Status of Powered Wheelchair", *Intelligent system conference* 1, 2019.
 43. OgechukwuOkonor, Mo Adda, "Power Optimisation Model for Leveraging Cloud System", *IEEE Conference*, 2019.
 44. Okonor, Mo Adda, "Intelligent Approach to Minimising Power Consumption in Cloud-Based System Collecting Sensor data and Monitoring the Status of Powered Wheelchair", *Intelligent system conference* 1, 2019
 45. Oro E, Depoorter V, Garcia A, Salom J., "Energy efficiency and renewable and energy integration in data centres", *Strategies and modelling Review. Renewable and Sustainable Energy Review* 2015;429-445

-
46. P. Arcaini, E. Riccobene, and P. Scandurra, "Modeling and analyzing maps-k feedback loops for self-adaptation," in Proceedings of the 10th International Symposium on Software Engineering for Adaptive and Self-Managing Systems, 2015, pp. 13–23.
 47. P. Arroba, J. M. Moya, J. L. Ayala, and R. Buyya, "DVFS-aware consolidation for energy-efficient clouds," in Proceedings of the International Conference on Parallel Architecture and Compilation. IEEE, 2015, pp. 494–495.
 48. R. Bellman, "Dynamic programming and Lagrange multipliers," Proceedings of the National Academy of Sciences, vol. 42, no. 10, pp. 767–769, 1956. 194
 49. S. Albers, "Energy-efficient algorithms", Communications of the ACM, vol. 53, no. 5, pp. 86– 96, 2010.
 50. S. Belaid and A. Mellit, "Prediction of daily and mean monthly global solar radiation using support vector machine in an arid climate," Energy Conversion and Management, vol. 118, pp. 105–118, 2016.
 51. S. Devadas and S. Malik, "A survey of optimization techniques targeting low power VLSI circuits", in Proceedings of the 32nd ACM/IEEE Conference on Design Automation, 1995, pp. 242–247.
 52. S. K. Garg, C. S. Yeo, A. Anandasivam, and R. Buyya, "Environment-conscious scheduling of HPC applications on distributed cloud-oriented data centres", Journal of Parallel and Distributed Computing, 2010.
 53. Sebastian Tramp, Philipp Frischmuth, Timofey Ermilov, Saeedeh Shekarpour and Soren Auer, "An architecture of a distributed semantic social network, 2012."
 54. Sharma. A, Y Yao, L Huang- Proceeding,2012- IEEEExplore.ieee.org
 55. Simona Elena and Varlan, "Advantages of Semantic web Technologies in the Knowledge-based Society", 2010
 56. SPEc power benchmarks, Standard Performance EvaluationCorporation. <http://www.spec.org/benchmarks.html#power>.
 57. T. Adhikary, A. K. Das, M. A. Razzaque, M. Alrubaian, M. M. Hassan, and A. Alamri, "Quality of service-aware cloud resource provisioning for social multimedia services and applications," Multimedia Tools and Applications, vol. 76, no. 12, pp. 14 485–14 509, 2017.
 58. T. Bawden. (2016) Global warming, "Data centres to consume three times as much energy in the next decade", experts warn. [Online]. Available:

<http://www.independent.co.uk/environment/global-warming-datacentres-to-consume-three-times-as-much-energy-in-next-decade-experts-warna6830086.html>

59. Tenna Mathew, KC Sekaran, J Jose, “Study and analysis of various task scheduling algorithm in the cloud computing environment”, 2014 International Conference, 2014 - ieeexplore.ieee.org
60. V. Tiwari, P. Ashar, and S. Malik, “Technology mapping for low power”, in Proceedings of the 30th Conference on Design Automation, 1993, pp. 74–79.
61. V. Pallipadi and A. Starikovskiy, “The on-demand governor”, in Proceedings of the Linux Symposium, vol. 2, 2006.
62. NIKAEIN, N.(1999) Reactive Autonomous Mobile Agent.
63. 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. 25, no. 6, pp. 599–616, 2009.
64. M. B. Srivastava, A. P. Chandrakasan, and R. W. Brodersen, “Predictive system shutdown and other architectural techniques for energy-efficient programmable computation”, IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 4, no. 1, pp. 42–55, 1996.
65. R. Nathuji and K. Schwan, “Virtualpower: Coordinated power management in virtualized enterprise systems”, ACM SIGOPS Operating Systems Review, vol. 41, no. 6, pp. 265–278, 2007.
66. R. Raghavendra, P. Ranganathan, V. Talwar, Z. Wang, and X. Zhu, “No power struggles: Coordinated multi-level power management for the data centre”, SIGARCH Computer Architecture News, vol. 36, no. 1, pp. 48–59, 2008.
67. M. Stillwell, D. Schanzenbach, F. Vivien, and H. Casanova, “Resource allocation using virtual clusters”, in Proceedings of the 9th IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGrid 2009), Shanghai, China, 2009, pp. 260–267.
68. M. Cardoso, M. Korupolu, and A. Singh, “Shares and utilities based power consolidation in virtualized server environments”, in Proceedings of the 11th IFIP/IEEE Integrated Network Management (IM 2009), Long Island, NY, USA, 2009.
69. Nathuji R, Schwan K., “Virtualpower: coordinated power Management in virtualized enterprise systems”, ACM SIGOPS Operating Systems Review 2007;41(6):265-278.

-
70. Oro E, Depoorter V, Garcia A, Salom J., “Energy efficiency and renewable and energy integration in data centres”, Strategies and modelling Review. Renewable and Sustainable Energy Review 2015; 429-445
 71. OgechukwuOkonor, Mo Adda, “Intelligent Approach to Minimizing Power Consumption in Cloud-Based System Collecting Sensor data and Monitoring the Status of Powered Wheelchair” Intelligent system conference 1, 2019.
 72. OgechukwuOkonor, Mo Adda, “Power Optimisation Model for Leveraging Cloud System”, IEEE Conference, 2019.
 73. P. Arcaini, E. Riccobene, and P. Scandurra, “Modeling and analyzing maps-k feedback loops for self-adaptation,” in Proceedings of the 10th International Symposium on Software Engineering for Adaptive and Self-Managing Systems, 2015, pp. 13–23.
 74. P. Arroba, J. M. Moya, J. L. Ayala, and R. Buyya, “DVFS-aware consolidation for energy-efficient clouds,” in Proceedings of the International Conference on Parallel Architecture and Compilation. IEEE, 2015, pp. 494–495.
 75. PARK., A. S.-B, 2004, “A service-based agent system supporting mobile computing”, Heinisch-Westfalisch Technischen Hochschule.
 76. PHAM, V. A. & KARMOUCH, A., “Mobile software agent: an overview”, Communication Magazine, IEEE, 36, 26-37. 1998
 77. Verma A, Ahuja P, Neogi A. pMapper, “Power and migration cost aware application placement in virtualized systems”, Proceedings of the 9th ACM/IFIP/USENIX international conference on middleware, Springer, Leuven, Belgium, 2008; 243-264.
 78. X. Fan, W.D. Weber, L. Barroso, “Power provisioning for a warehouse-sized computer”, in proc. International Symposium on the computer, Architecture (ISCA07), June 2007, pp. 13-23, DOI: 10.1109/ISPASS.2011.5762739
 79. Y. Song, H. Wang, Y. Li, B. Feng, and Y. Sun, “Multi-Tiered On-Demand resource scheduling for VM-Based data centre”, in Proceedings of the 9th IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGrid 2009), Shanghai, China, 2009, pp. 148–155.
 80. Zhang, Q., Cheng, L., Boutaba, R., “Cloud computing: state-of-the-art and research challenges”, Published online: The Brazilian Computer Society (2010).

Appendix

Appendix1: Symbols and Definitions

Variable	Description
P_i	Ports
P	Power
E	Energy
μ	Utilisation
S_{rv}	Server
R	Rate
N_c	network configuration
V	set of network device
\mathcal{E}	set of links
S_{wi}	Switches
P_{cc}	Computing cost

Appendix2: Simulation code sample

```
import jdk.jshell.execution.Util;
import org.apache.commons.math3.util.Pair;
import org.cloudbus.cloudsim.Log;
import simulations.
import utils.Utils;
import utils.Vars;
import spec.Vars
import java.io.File;
import java.io.FileWriter;
import java.io.IOException;
import java.lang.reflect.InvocationTargetException;
import java.util.ArrayList;
import java.util.List;
import java.util.Map;

import static java.lang.System.exit;
import static java.lang.System.out;

public class EvaluateResult {
    /**
     * Nothing to change here, please edit the simulation.ini file
     * @param args
     */

    public static void main(String[] args){

        String[] detection_types = {"DaemonBased","TimeBased"};

        for(String detection_type: detection_types) {

            SimulationRunner simulationRunner = null;

            //TimeBased Simulation
            String outpath = "cloudsim-agent-master/res/"+detection_type+".csv";
            double repeatingTime = 5;
            boolean printDatacenter = false;
            double daemonMipsConsumption =100;
            double lower_bound = 0;
            double upper_bound = 0.7;
            String eval_name = detection_type+"simulation";
            Wini ini = new Wini();

            String[] result = null;
            List<String[]> resultList = new ArrayList<>();
            String[] headers = {"host_size", "switch_size","vm_size", "host_power",
"switch_power","num_cloudlet","cloudlet_load","actual_time","total_time"};
```

```

        for(int nbHost=10; nbHost<=100; nbHost+=10){
            for(int nbVms=1; nbVms<=10; nbVms++){
                for(int nbCloudlets=10; nbCloudlets<=100;
nbCloudlets+=10){
                    try {
                        ini.put("simulation","name",
eval_name);
                        ini.put("datacenter","nb_hosts", nbHost);
                        ini.put("datacenter","nb_vms", nbVms);
                        ini.put("cloudlets","nb_cloudlets", nbCloudlets);
                        ini.put("cloudlets","randomize_cloudlet_length",
true);
                        ini.put("cloudlets","mean_cloudlet_length",
100000);
                        ini.put("cloudlets","standard_cloudlet_deviation",
900);
                        ini.put("TimeBased","repeating_time",
repeatingTime);
                        ini.put("DaemonBased","mips_consumption",
daemonMipsConsumption);
                        ini.put("DaemonBased","lower_bound_ratio",
lower_bound);
                        ini.put("DaemonBased","upper_bound_ratio",
upper_bound);
                        ini.put("simulation","print_datacenter",
printDatacenter);

                        Class detectionClass =
Class.forName("simulations."+detection_type+"Simulation");

                        simulationRunner =
(SimulationRunner)detectionClass.getConstructor(ini.getClass()).newInstance(ini);
                    }
                    catch (ClassNotFoundException |
NoSuchMethodException | IllegalAccessException | InstantiationException |
InvocationTargetException e) {
                        e.printStackTrace();
                        exit(-1);
                    }

                    try {
                        simulationRunner.init();
                        result = simulationRunner.start();
                        resultList.add(result);
                    } catch (Exception e) {

```

```

        Log.println("An exception occurred during the
simulation !");
        e.printStackTrace();
    }
}

    }
    boolean b = writeResult(outpath, resultList, headers);
    if(b){
        System.out.println("\n\n successful saving output to file " + outpath);
    }
    else{
        System.out.println("\n\n error saving output to file " + outpath);
    }
}

public static boolean writeResult(String outPath, List<String[]> resultList, String[] header)
{
    CSVWriter writer;
    try {
        writer = new CSVWriter(new FileWriter(new File(outPath)));
        writer.writeNext(header);
        writer.writeAll(resultList);
        writer.close();
        return true;
    } catch (IOException e) {
        return false;
    }
}
}


```

FORM UPR16

Research Ethics Review Checklist

Please include this completed form as an appendix to your thesis (see the Research Degrees Operational Handbook for more information)



Postgraduate Research Student (PGRS) Information		Student ID:	837637
PGRS Name:	Ogechukwu Mercy Okonor		
Department:	School of Computing	First Supervisor:	Dr Mo Adda
Start Date: (or progression date for Prof Doc students)	1 st Feb. 2017		
Study Mode and Route:	Part-time <input type="checkbox"/> Full-time <input checked="" type="checkbox"/>	MPhil <input type="checkbox"/> PhD <input checked="" type="checkbox"/>	MD <input type="checkbox"/> Professional Doctorate <input type="checkbox"/>
Title of Thesis:	Energy Efficiency in Cloud Computing with an Intelligent Mobile Agent		
Thesis Word Count: (excluding ancillary data)	31566		
<p>If you are unsure about any of the following, please contact the local representative on your Faculty Ethics Committee for advice. Please note that it is your responsibility to follow the University's Ethics Policy and any relevant University, academic or professional guidelines in the conduct of your study</p> <p>Although the Ethics Committee may have given your study a favourable opinion, the final responsibility for the ethical conduct of this work lies with the researcher(s).</p>			
UKRIO Finished Research Checklist: (If you would like to know more about the checklist, please see your Faculty or Departmental Ethics Committee rep or see the online version of the full checklist at: http://www.ukrio.org/what-we-do/code-of-practice-for-research/)			
a) Have all of your research and findings been reported accurately, honestly and within a reasonable time frame?	YES <input checked="" type="checkbox"/> NO <input type="checkbox"/>		
b) Have all contributions to knowledge been acknowledged?	YES <input checked="" type="checkbox"/> NO <input type="checkbox"/>		
c) Have you complied with all agreements relating to intellectual property, publication and authorship?	YES <input checked="" type="checkbox"/> NO <input type="checkbox"/>		
d) Has your research data been retained in a secure and accessible form and will it remain so for the required duration?	YES <input checked="" type="checkbox"/> NO <input type="checkbox"/>		
e) Does your research comply with all legal, ethical, and contractual requirements?	YES <input checked="" type="checkbox"/> NO <input type="checkbox"/>		
Candidate Statement:			
I have considered the ethical dimensions of the above named research project, and have successfully obtained the necessary ethical approval(s)			
Ethical review number(s) from Faculty Ethics Committee (or from NRES/SCREC):		ETHICS-10086	
If you have not submitted your work for ethical review, and/or you have answered 'No' to one or more of questions a) to e), please explain below why this is so:			
<div style="border: 1px solid black; height: 20px; width: 100%;"></div>			
Signed (PGRS):			Date: 1/02/2021