I.J. Information Technology and Computer Science, **2025, 3, 72-96**
Published Online on June 8, 2025 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijitcs.2025.03.06

Evaluating Energy-efficiency and Performance of Cloud-based Healthcare Systems Using Power-aware Algorithms: An Experimental Simulation Approach for Public Hospitals

Aschalew Arega*
Faculty of Computing and Software Engineering, Arba Minch University, Arba Minch, Ethiopia
E-mail: aschalew.arega1@gmail.com
ORCID iD: https://orcid.org/0000-0002-2734-5530
*Corresponding author

Durga Prasad Sharma
Arba Minch University, Ethiopia
E-mail: dp.shiv08@gmail.com
ORCID iD: https://orcid.org/0000-0002-1654-901X

Received: 15 November 2024; Revised: 21 January 2025; Accepted: 28 March 2025; Published: 08 June 2025

Abstract: The use of cloud computing, particularly virtualized infrastructure, offers scalable resources, reduced hardware needs, and energy savings. In Ethiopian public hospitals, the lack of integrated healthcare systems and a national data repository, combined with existing systems deficiencies and inefficient traditional data centers, contribute to energy inefficiency, carbon emissions, and performance issues. Thus, evaluating the energy efficiency and performance of a cloud-based model with various workloads and algorithms is essential for its successful implementation in healthcare systems and digital health solutions. The study experimentally evaluates a cloud-based model's energy efficiency and performance for smart healthcare systems, employing descriptive and experimental designs to simulate cloud infrastructure. Simulations are conducted on diverse workloads in CloudSim using power-aware (PA) algorithms (along with VmAllocationPolicy and VmSelectionPolicy), and dynamic voltage frequency scaling (DVFS). Results reveal that the number of VMs and their migrations significantly impact energy consumption, with some algorithms achieving notable energy savings. Lr/Lrr-based algorithms are particularly energy-efficient, with LrMc and LrrMc saving 29.36% more energy than IqrMu at 55 VMs, and LrrRs saving 30.20% more at 1,765 VMs. DVFS adjusts energy consumption based on the number of VMs, while non-power-aware (NPA) consumes maximum energy based on hosts, regardless of the number of VMs. VM migrations, energy consumption, and average SLAV are positively correlated, while SLA is negatively correlated with these factors. In PlanetLab, energy consumption and average SLAV show a strong positive correlation (0.956) at Workload6, while SLA at Workload2 and average SLAV at Workload1 show a weak negative correlation (-0.055). Excessive migrations can disrupt the system's stability/performance and cause SLA violations. Task completion time is influenced by VM processing power and cloudlet length, being inversely proportional to VM processing power and directly proportional to cloudlet length. Overall, the findings suggest that cloud virtualization and energy-efficient algorithms can enhance healthcare systems performance, patient care, and operational sustainability.

Index Terms: Cloud Computing, Cloudsim Simulations, Energy Consumption, Energy Efficiency, Healthcare Information Systems, Power-aware Algorithms, Virtualization.

1. Introduction

Cloud computing offers on-demand access to internet-based computer resources like data storage and computing power, enabling faster innovation, flexible resources, and improved energy and cost efficiency[1–7]. Studies [2,7–10] reported that cloud computing is enabled by several key technologies: virtualization for resource partitioning, load balancing [10] for distributing workloads across multiple nodes, replication for data redundancy and disaster recovery, identity and access management (IAM) for secure user authentication and authorization, and service-level agreements

(SLA) that defines a customer's service quality expectations with cloud service provider. SLA is a formal contract agreement that define expected service levels [8,9], such as uptime, response times, and performance metrics, while service- level agreement violation (SLAV) [9,11] refers to the failure to meet the terms of an SLA, which can lead to penalties, customer dissatisfaction, and a loss of trust [7,12]. Ethiopian healthcare settings face unique infrastructure challenges, including limited IT resources, poor internet connectivity, financial constraints, and unreliable power supply, especially in remote areas, distinguishing them from global counterparts. A case study [13] in Ethiopia finds Ethio-Telecom's readiness to launch infrastructure as a service (IaaS), offering virtual computing resources.

In cloud computing, a data center contains many physical machines (PMs) that host multiple virtual machines (VMs). A study [6] explains that to host a VM, a PM must provide necessary resources, such as CPU, memory, storage, and network bandwidth. Virtualization allows multiple VMs on a single PM, enabling scalable resource provisioning and efficient resource management, as illustrated in Fig.1. VMs share host's resources based on VM allocation [14] and VM selection policies [6,15], enabling cost-effective, energy-efficient resource sharing that allows multiple customers to use a single physical resource or application simultaneously [4,16]. Hypervisors (type-I or type-II) allow multiple operating systems to run concurrently on the same hardware. However, overloading a PM with too many VMs is not advisable, and running multiple VMs on a single host requires careful consideration of the host's specifications for optimal performance.

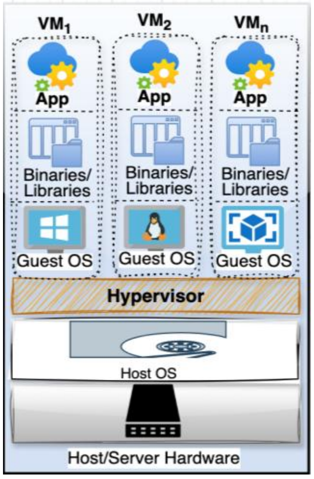

Fig.1. Virtualization [4,16]

A cloud-based contextualized model was proposed to improve healthcare services, emphasizing cloud computing's role in efficient data management, storage, real-time access, and scalable infrastructure. Despite these advantages, the model's smartness and energy efficiency must be evaluated in a cloud environment before implementation. However, conducting repeatable benchmarking experiments on real cloud infrastructure is costly (incurs high costs) and challenging for academic research [5,6,17,18]. According to a study [19], the pay-per-use cloud model incurs high costs for repeatable experiments in real cloud environments, making simulation environments like CloudSim an ideal alternative for evaluating resource management policies and ensuring repeatability and clarity in experimental results. Cloud simulators provide a viable alternative for repeated simulation and evaluation of cloud infrastructure. Thus, CloudSim was chosen as the simulation platform in this study to ensure repeatability and clarity in the experimental analysis results.

CloudSim is an open-source framework licensed under Apache 2.0 for modeling and simulating cloud infrastructures and services[4,20], allowing users to test various cloud configurations, performance metrics, and resource utilization algorithms [19] without incurring high costs, while modeling parameters such as data centers, hosts, VMs, cloudlets, service brokers, compute resources, energy efficiency, and resource provisioning techniques. In CloudSim, the data center acts as the resource provider [4,21]. As shown in Fig.2., the CloudSim architecture consists of three layers: the user code layer (defining basic entities like hosts, applications, VMs, cloudlets, and broker scheduling policies), the CloudSim layer (modeling and simulating virtualized cloud-based data center environments with management interfaces for VMs, memory, storage, and bandwidth), and the core simulation engine layer (modeling queuing and communication between components) [16,18,21,22]. Our study considers the CloudSim components with a shaded blue background, as shown in Fig.2, for evaluating a cloud-based model for healthcare systems. The CloudSim was selected after the parameter-based comparison of other alternative methods, as shown in Table 2 and Table 3. While CloudSim provides a valuable platform for simulating cloud environments. Also, we acknowledge its limitations in real-world validation due to real-time cost constraints. For a researcher, access to real-world systems for experimentation and analysis with real-world systems data and environment is a tough task in a country where the adoption of cloud computing is very limited, especially in healthcare. And therefore the study chose the simulated environment only with the mentioned limitation. In future work, we can conduct real-world validation of our model to determine how it is theoretically sound and practically applicable in the Ethiopian healthcare sector.

The existence of paper-based methods and multiple isolated units (non-integrated systems) in Ethiopian public hospitals has led to several critical problems, including remotely inaccessible patient histories in case of emergency, high costs, scalability issues, disorganized health data, energy inefficiency, performance issues, carbon emissions, and poor resource allocation and sharing of government-owned healthcare resources. To minimize reliance on manual processes and multiple systems within healthcare, adopting a cloud-based model is essential. However, the model lacks experimental analysis to efficiently and effectively address the need for smart healthcare information systems. Cloud-based solutions, with their potential for scalable resources, reduced hardware resource needs, and improved energy efficiency, can offer a promising alternative to address these challenges and improve healthcare service delivery.

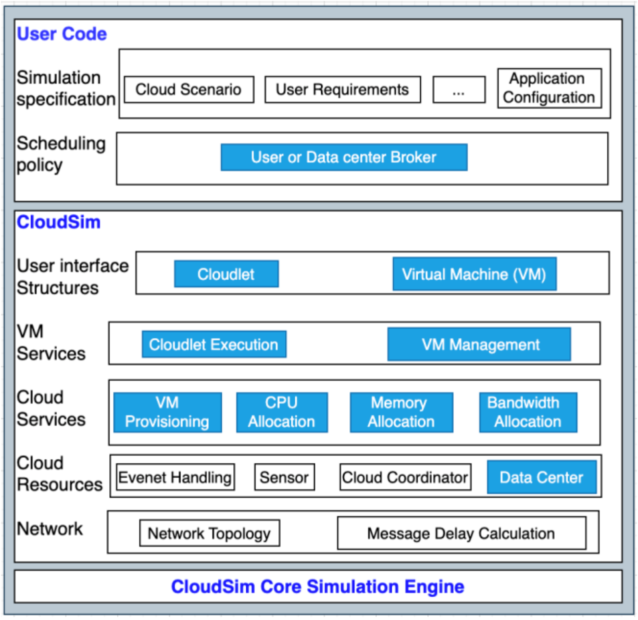

Fig.2. CloudSim Layered Architecture [16,18,21]

Research question (R.Q): How can we evaluate/test a cloud-based contextualized model for smart and energy-efficient healthcare information services? That was a significantly important concern in the present study. The prime objective of this paper is to experimentally evaluate a cloud-based model's energy-efficiency and performance for smart healthcare information systems using power-aware algorithms.

This paper evaluates the proposed model's energy efficiency, scalability, and data consolidation using PlanetLab and random workloads. Energy efficiency in IT refers to optimizing resource usage to minimize energy consumption without compromising performance [11,23]. The key parameters include data centers, hosts, VMs, energy consumption (kWh), SLAs, average SLAV [9], VM migrations [11], and cloudlets. Metrics like processing power (MIPS– million instructions per second) and task completion time are used to assess computing system performance. The study analyzes real-world workloads from PlanetLab [24] and random workloads based on hospital numbers. Both workloads are tested using various algorithms, including power-aware (PA), DVFS, and non-power-aware (NPA) to measure energy consumption and identify optimal algorithms for a power-efficient cloud data center. The study conducted extensive experimental simulations with these algorithms focusing on VM allocation and selection policies. It also analyzes the correlation between VM migrations, energy consumption, SLAs, and average SLAVs across selected workloads.

2. Literature Review and Related Works

Recent advancements in cloud computing research have focused on virtualized servers, developing energy-saving techniques and power-aware algorithms to optimize resource allocation, utilization, and efficiency in large data centers. Studies [1,2] emphasize that cloud computing is key to building smart, sustainable healthcare systems, offering benefits like virtualization, energy efficiency, multi-tenancy, data consolidation, automation, and reduced greenhouse gas emissions and paper use. It provides on-demand access to computing resources [3] such as CPU, memory, storage, and bandwidth, enabling scalable and efficient solutions. Studies[2,3,6], considered the key parameters for analyzing cloud system's behavior and performance using CloudSim including data centers, VMs, workloads, networks, energy consumption, cloud services, resource utilization, and response time. In the simulated experimentations, these parameters can be adjusted to assess different scenarios, evaluating performance, efficiency, and reliability of cloud systems. Simulating data centers (DCs) enables the analysis of scalability and workload distribution. VM simulations assess resource allocation, scheduling, and performance. Workload simulations are essential for performance evaluation and resource management, reflecting real-world diversity and variability, and reproducibility. Energy consumption simulations model power usage based on workload and resource utilization, using dynamic metrics like VM migrations, Cloudlets, processing power, and time. Cloud services simulations evaluate resource accessibility, scalability, SLA compliance, interoperability, collaboration, and performance.

Data centers are the backbone of cloud computing, housing IT infrastructure for storage, processing, and communication [3]. Studies [3,6,11] highlight that the rapid growth in data center energy consumption raises significant economic and environmental challenges, contributing 0.5% to 1% of global CO_2 emissions. As noted in [25], the data centers' continuous operation of numerous active hosts, servers, and networking devices to meet users' demand leads to substantial energy consumption. According to [6], the expansion of data centers to support high-performance computing and large-scale storage increases both energy consumption and carbon emissions, resulting in high operating costs [26]. In cloud computing [3], a data center consists of multiple PMs, each hosting several VMs. Efficient VM placement on PMs is essential for optimizing resource utilization, reducing costs, and meeting application performance requirements. According to a study [3], server consolidation involves mapping multiple VMs onto PMs to share resources such as CPU, storage, bandwidth, and memory. This involves identifying an optimal mapping of VMs to PMs to maximize resource utilization [19], minimize active hosts, and balance the workload across the data center.

Studies [2,27] focus on enhancing cloud data center performance to meet growing computing and storage demands, marking the importance of effective task scheduling [19] for optimizing resource use, reducing energy consumption, minimizing response time, and maximizing efficiency. As reported by[19,27] task scheduling is vital for enhancing cloud data center performance in the face of increase service demand. However, increasing energy consumption remains a challenge, driven by inefficient cooling, network equipment, and idle servers [27]. Other studies [15,26] revealed that large data centers, which contain thousands of servers, networking devices, racks, and cooling systems, consume significant amounts of energy. As per the studies[11,15], to reduce energy consumption and carbon footprint, virtual machine (VM) consolidation is promising; however, unwanted VM migrations during dynamic consolidation can create processing overhead and costs, ultimately impacting data center performance and resource utilization.

A study [2] underscores the need for adaptable resource allocation in cloud computing and proposes Knowledge-Based Flower Pollination Algorithm (KB-FPA) for efficient resource management. The algorithm is evaluated using three performance metrics: resource utilization, execution time, and energy consumption. However, the study primarily focuses on VM resource allocation and tests only three workloads. Several studies [11,15,19] indicate that CPU usage is more closely linked to energy consumption than other factors like network bandwidth or memory usage. CloudSim models energy consumption based primarily on CPU utilization, which has the highest energy impact. Study [11] emphasizes that VM consolidation reduces VM migrations, SLA breaches, and energy consumption, making it crucial for cloud resource optimization. Additionally, [28] highlights the importance of reducing energy use, VM migrations, and SLA violations while maintaining QoS in cloud data centers.

According to a study [14], virtualization-based software solutions are key to energy-efficient cloud computing, employing consolidation techniques to switch off idle servers and migrate VMs between underutilized, overutilized, and moderately utilized servers [24] for optimal utilization. VM consolidation is a key strategy for reducing energy consumption and optimizing resource utilization in cloud data centers [5,6]. According to a study [5], algorithms must be tested on multiple workloads to determine its optimality. A study [29] states that VM consolidation is widely used in cloud data centers to optimize resource utilization and reduce energy consumption. However, it [29] warns that sudden resource utilization bursts and insufficient analysis of historical resource utilization can lead to host overloading and SLA violations. To save energy, studies [6,14,28], highlight, that over-loaded hosts should migrate VMs to idle hosts, while under-loaded hosts should migrate VMs to suitable hosts to prevent SLA violations. Study [10], emphasizes that load balancing is crucial for cloud-based systems, ensuring no VMs are idle, underutilized, or overloaded during operation. Load balancing distributes workloads across multiple nodes for efficient service delivery and infrastructure integrity.

In the review of [30], cloud computing conserves energy consumption in data centers through shutdown, hibernation, and low-power modes. While most research relies solely on CPU utilization for host overload detection, the study [6] proposes a multiple regression algorithm incorporating CPU, memory, and bandwidth utilization, and developed multiple regression host overload detection (MRHOD) and hybrid local regression host overload detection (HLRHOD) algorithms to improve VM consolidation. A study [4] simulates a smart grid cloud model using CloudSim to analyze how parameters like VMs, image size, RAM, bandwidth, and cloudlet length affect processing costs and completion times. In a time-shared allocation policy, multiple VMs share the host's resources concurrently, while in a space-shared allocation policy, only one VM runs its cloudlet at a time. Accordingly, cloudlet completion time varies significantly with VM bandwidth. Consequently, in the time-shared policy, increasing cloudlets extends the task completion time for each cloudlet.

According to a study [22], companies like Google and Amazon operate large global data centers with virtual computers, leading to high power costs and rising energy demand in IT firms. The study [22] uses CloudSim to simulate two power consumption approaches: DVFS and NPA. In the NPA model, hosts consume maximum power, resulting in higher service costs and loss of profits for providers. In contrast, DVFS saves more energy and leads to fewer SLA violations. Research studies [2,14,22] recommend using power-aware techniques to address energy-saving challenges. To understand and analyze the state-of-the-art energy efficiency issues and concerns in the data centers using various algorithms and techniques, Table 1 summarizes the problems, objectives, contributions, and limitations identified in the review of related works.

Table 1. Summary of problems, objectives, contributions, and limitations observed in the review of related works

A study [Ref.]	Problems/Issues	Focus/objective: To-	Contribution(s)	Scope and Limitations
[3]	VM placement problem (VMPP)	Reduce energy consumption in data centers	Designed a Duelist Algorithm for VMPP	Considers only CPU allocation policy
[2]	Resource allocation adaptability and energy efficiency	Substantially reduce energy consumption	Proposed a novel Algorithm (KB-FPA)	Focus solely on the VM resource allocation, and test very few workloads(3)
[28]	Data centers (DC) consume significant energy	Reduce energy consumption, VM migrations, and SLAVs	Proposed energy-efficient VM consolidation Framework	Tested with only one dataset (PlanetLab)
[4]	Inefficiencies in time-shared and space-shared resource allocation	Evaluate the impact of allocation policies on cloudlet completion times at host and VM	Proposed a smart grid cloud model	Used a few hosts (2) and VMs to simulate the model
[5]	Increasing energy consumption in computing data centers	Test proposed algorithms on multiple datasets for optimal assessment	Comparison of various algorithms and policies across multiple datasets	The numbers of hosts, VMs, and workloads used are unclear/ unspecified
[6]	High energy consumption and carbon emissions	Improve resource utilization and energy efficiency in cloud DC	Developed a Multiple Regression algorithm for VM consolidation	Considers only three parameters and VM selection policies
[27]	Rising energy consumption by data centers	Minimize energy use: optimizing VM scheduling & reducing migration time	Proposed a QoS-aware fault-tolerant task scheduling algorithm	Neglects the balance between energy reduction, VM migrations, & SLAVs. Also uses a single dataset
[10]	Load balancing among VMs across geographically dispersed DC.	Enhance the efficiency of cloud-based computing systems	Proposed component-based Throttled load balancing algorithm	Utilized a few VMs and parameters, with unspecified host status and unknown datasets

Cloud simulators: Selecting appropriate cloud simulator is a crucial task to address gaps and evaluate a cloud-based model, prompting a review of common cloud simulators. According to [17], cloud computing, driven by virtualization and pay-per-use services, is costly and challenging for real-world experimentation, leading to a focus on simulation frameworks. A study [18] stated that conducting research in real cloud environments is challenging for individuals and small institutions due to the high costs associated with cloud setup. As shown in Table 2, selecting the appropriate simulation tool requires a comparative analysis of available simulators using their features.

Table 2. Comparative analysis of major cloud simulation tools and their limitations

Simulation Tool [Ref.]	Type	Purpose/Focus-	Modeling /Simulating	Features	Limitations and Scope
CloudSim [17,18,22,31]	Framework	Provides a flexible and extensible framework (General-purpose)	Data center, VMs, brokers, and resource provisioning, & workloads	Scalability, time-shared, & space-shared policies, & support for virtualization	Limited network model, & delays when application load exceeds host capacity
CloudAnalyst [17,18,31]	Simulation tool	Evaluate large-scale distributed apps (Specific-purpose)	Large-scale distributed applications	Provides location info., (DC, users), and GUI	Limited support for power modeling
EMUSIM [17,18,31]	Framework	Evaluate performance of cloud apps (special purpose)	Information accessibility	Exploits information from the application	Limited scalability (emulation and hardware)
GreenCloud [17,18,31]	Framework	Simulating energy modeling (Specific-purpose)	A fine-grained model: Network equipment	Cloud network & energy monitoring	Scalability issues arise (simulation time & space)
iCanCloud [17,18,31]	Framework	Architecture & Configurations	Large-size cloud environments	Complete hypervisor, GUI, & define storage capabilities	Overlooks a complete energy utilization model
GroundSim [17,18,31]	Framework	For scientific applications (both grid & cloud environments)	Network resources of cloud and grid	Scientific workflow application, Extensible	Presents only basic network, and experiences performance degradation

Choosing the right tool is challenging due to the variety of simulators for specific tasks. Therefore, we compared major cloud simulators (Table 3) based on key parameters to identify the most suitable option for our simulation experiments [17,18,31].

Table 3. Comparison of major cloud simulators for optimal selection

Parameters	CloudSim	CloudAnalyst	EMUSIM	GreenCloud	iCanCloud	GroudSim
Availability	Open source	Open source	Open source	Open source	Open source	Open source
Support Infrastructure(IaaS)	√	√	√	√	√	√
Application model	√	√	√	√	√	√
Resource allocation & load balancing	√	√	√	√	√	√
Energy Model	√	√	√	√	×	×
VM migration policy	√	×	×	×	×	×
Network topology model	√	√	√	√	√	×
SLA support	×	×	×	√	×	×
Cost Model	√	√	√	×	√	√
Platform Portability	√	√	√	×	√	√
Scalability	√	√	√	√	√	√

Based on the mentioned comparative analyses in Table 3, we have chosen CloudSim. CloudSim, developed by the CLOUDS laboratory at the University of Melbourne [14,20], is an extensible simulation framework for modeling, simulating, and experimenting with cloud computing infrastructures and services [4,21]. It supports the modeling of data centers, hosts, VMs, cloudlets, brokers, and resource provisioning policies, enabling detailed simulations of virtualized environments and resource management [5,6]. While CloudSim's energy consumption models focus only on CPU utilization, other simulators like GreenCloud and iCanCloud account for additional factors such as network, storage, and memory [17]. Despite this limitation, CloudSim is preferred for its generalized framework and comprehensive modeling capabilities. Numerous studies [17,18,22,31] have shown that most cloud simulators are built upon extensions of CloudSim.

Based on the review of related works and the feature-based comparative analysis of the simulators in Table 3, this study identifies the following research scope and features limitations:

- Lack of suitable/appropriate cloud simulation for smart hospital applications in developing countries like Ethiopia.
- Absence of empirical analysis on power-aware, DVFS, and non-power-aware algorithms for smart healthcare information systems.
- Neglect the balance and correlation among VM migrations, energy consumption, and SLAs.
- Insufficient exploration of how cloud parameters—such as hosts, VMs, and cloudlets—affect task completion times.
- Reliance on a single dataset, a few host/VM usage, and limited workloads.

The identified gaps in terms of efficiency and performance of the existing systems in the healthcare sector underline vital areas where cloud-based solutions can be explored to investigate and employ toward improvements for the development of the next-generation and green healthcare systems. Developing countries like Ethiopia face challenges in adopting smart hospital applications and cloud solutions. Empirical analysis of PA algorithms is essential for reducing energy consumption, while balancing energy reduction, VM migrations, and SLAs is crucial for efficient cloud resource management. The impact of task completion times is underexplored, potentially hindering system performance. Relying on limited datasets and workloads restricts the scalability and adaptability of cloud-based solutions in real-world scenarios. Addressing these gaps through cloud-based models can enhance efficiency, scalability, and sustainability in healthcare systems.

3. Research Methodology

According to a study [32], most healthcare organizations use cloud-based applications to improve service delivery; however, designing such systems requires in-depth knowledge of both cloud services and the healthcare environment. This study employs a mixed-methods approach, combining descriptive and experimental designs. It conducts experimental simulations of cloud infrastructure for smart and energy-efficient healthcare information services, evaluating a cloud-based contextualized model through power-aware, DVFS and non-power-aware algorithms across various workloads using CloudSim.

The rationale behind the choice of algorithms and policies is based on CloudSim's intrinsic support for various VM allocation and selection policies, which enables the simulation of power-aware algorithms in cloud environments. VM allocation policies determine how VMs are assigned to hosts, while VM selection policies are responsible for selecting the appropriate VM from a pool to be scheduled on a host. The paper applies two key policies—VmAllocationPolicy and

VmSelectionPolicy—simulating 20 distinct power-aware algorithms configurations by combining five VM allocation policies (IQR, LR, LRR, MAD, THR) with four VM selection policies (MC, MMT, MU, RS) [5,6,24]. The combination of these policies influences the overall efficiency of VM, particularly in terms of power-aware decisions.

VmAllocationPolicy:	**VmSelectionPolicy:**
Inter Quartile Range (IQR)	Maximum Correlation (MC)
Local Regression (LR)	Minimum Migration Time (MMT)
Local Regression Robust (LRR)	Minimum Utilization (MU)
Median Absolute Deviation (MAD)	Random Selection (RS)
Threshold (THR)	

4. Experimental Simulation Modeling with CloudSim

Hospitals in Ethiopia, under the Ministry of Health (MoH), urgently need digital transformation and digital health solutions [33] to enhance healthcare systems. Digital transformation has significantly enhanced technology adoption in healthcare, resulting in remarkable advancements through their synergy [34]. However, comprehensive experimental analysis and judicious adoption of emerging technologies like cloud computing are largely missing. This study simulates a cloud-based contextualized model, CloudCareEthiopia, as illustrated in Fig.3.

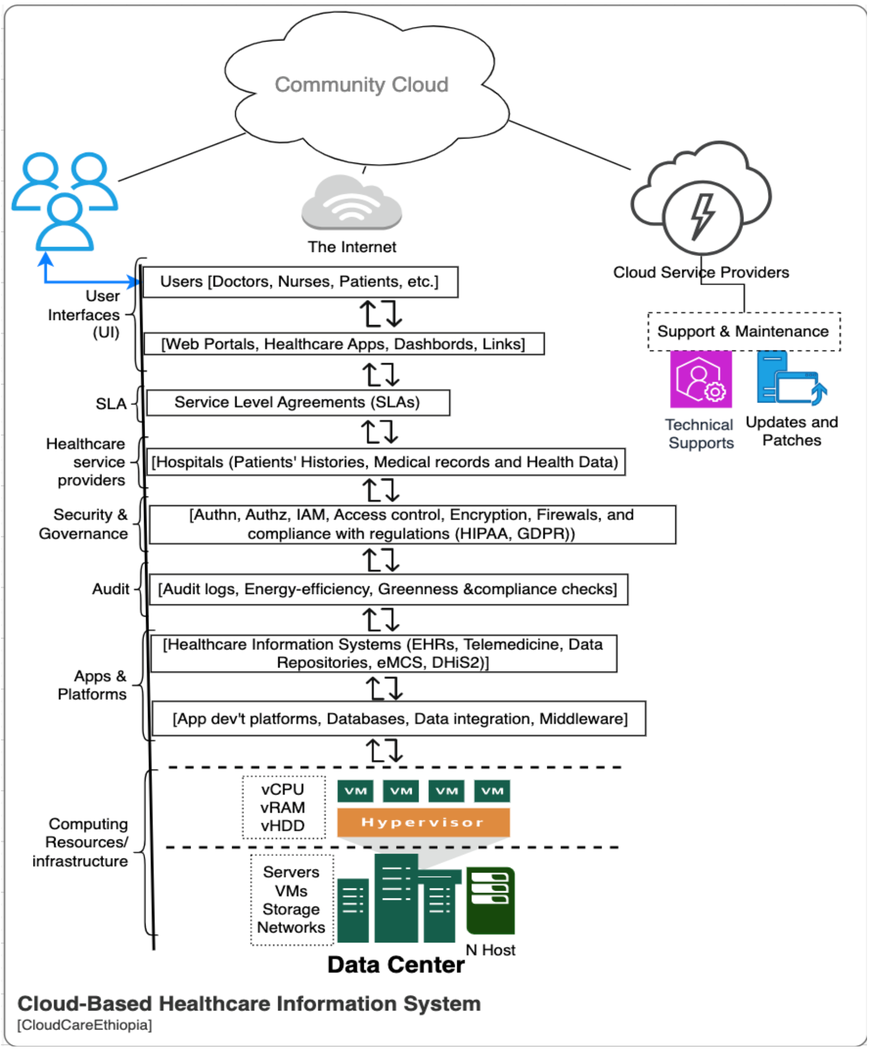

Fig.3. The proposed system view

Classes in CloudSim are vital for simulating cloud infrastructure, representing components like data centers, hosts, VMs, policies, and cloudlets. These classes work together to enable testing and analysis of cloud environments. Studies [4,20], highlight the key classes for modeling cloud infrastructure, as shown in Table 4. A concrete class provides complete functionality and can be instantiated directly without extension.

Table 4. Major classes in cloudsim considered in our experiments

CloudSim Class name	Class Type	Responsibility	Key resources characteristics
Datacenter (DC)	Superclass	Creating core cloud infrastructure	Hosts, Storage, CPU, Memory, Networks
Datacenter Charactersitics	Concrete	Defines attributes & capabilities of a data center– configuration information	Architecture, OS, hypervisor/VMM, hostList, time_zone, cost
Host	Concrete	Providing resources (CPU, RAM, Storage, Bw) to VMs	Amount of storage, RAM, type of processor (multi-core), MIPS, Bw
Cloud Coordinator	abstract	Coordinating tasks and managing interactions for efficient execution.	Allocation, task scheduling, load balancing, & component interaction (VMs, Hosts, DC).
Vm	Concrete	Enabling realistic modeling of cloud scenarios(virtualized env't)	VMs (mips, pesNumber /CPU, RAM, Bw, hypervisor)
Vm Allocation Policy	abstract	Allocate & evaluate performance of VM allocation techniques.	Policy (IQR, LR, LLR, MAD, THR)
Vm Selection Policy	abstract	Defining VM selection based on certain policies	Policy (MC, MMT, MU, RS)
Vm Scheduler	abstract	Manages policies for resource allocation to VMs on a specific host	Scheduling strategies: time-sharing, and space-sharing
Cloudlet	User-defined tasks	Modeling and simulating tasks/workloads for performance analysis	Cloudlet length, pesNumber, inpuFileSize, outputFileSize, utilization model,
Bw Provisioning	abstract	Defines the bandwidth (Bw) provisioning policy for VMs.	Allocate Bw in bits per second (bps), (Kbps/Mbps/Gbps)
Cloudlet Scheduler	abstract	Determines scheduling & execution of cloudlets on VMs	Policies (time-shared and space-shared) scheduling

4.1. Experimental Simulation Setup

The simulation experiments were conducted in a local cloud environment, with Java, Eclipse IDE (version 2024-03), and CloudSim installed on a MacBook Pro (Intel Core i5, 3.1GHz, 8GB RAM, macOS 13.6.6). The process involves setting parameters, applying algorithms, running them as a Java application, and analyzing the collected results. The data center in this simulation uses two server configurations with dual-core CPUs: the HP ProLiant ML110 G4 (Intel Xeon 3040, 2 cores, 1860 MHz, 4 GB RAM) and the HP ProLiant ML110 G5 (Intel Xeon 3075, 2 cores, 2660 MHz, 4 GB RAM) [6,14,21,22]. Hosts manage VM creation, migration, destruction, and provisioning, with the number of hosts and VMs determined by workload characteristics and total workloads [4].

Simulations are conducted using both PlanetLab and random workloads. These workloads are selected to represent a diverse range of usage patterns and test scenarios. PlanetLab workloads, derived from 800 global VM servers, provide real-world data on system behavior under realistic usage patterns. While not specific to Ethiopian healthcare IT, this data offers valuable insights into system behavior but generously reflecting real-world usage patterns. Conversely, random workloads generate arbitrary workloads to simulate unpredictable usage patterns, such as random CPU utilization for cloudlets (tasks). However, these workloads may not fully capture the complexities of real-world healthcare IT systems, which is beyond the scope of this paper.

A. Setting Parameters for PlanetLab Workloads

Using real-world data traces, like PlanetLab [6,14,28], in CloudSim provides realistic workloads, ensuring valid and credible simulations. These traces, collected from over 1,000 VMs across 500+ global locations, include ten days of CPU utilization data for ten distinct workloads, with host monitoring every five minutes [24,28,35]. This approach ensures more credible and diverse simulation scenarios. In our PlanetLab simulation, we modeled a data center with 800 heterogeneous nodes and 898 to 1,516 VMs using multiple algorithms/policies.

Table 5. The number of hosts and VMs for PlanetLab at each workload (WL)

	WL1	Wl2	WL3	WL4	WL5	WL6	WL7	WL8	WL9	WL10
Hosts	800	800	800	800	800	800	800	800	800	800
VMs	1052	898	1061	1516	1078	1463	1358	1233	1054	1033

B. Setting Parameters for Random Workloads

In our Random workloads simulation, the number of hosts and VMs was based on the 353 public hospitals under the Ministry of Health [36] and the 11 surveyed hospitals, used to allocate workloads and evaluate energy consumption. However, the number of hosts required in a data center for a hospital varies based on factors like hospital size, IT infrastructure complexity, application types, and data storage needs. This section estimates a minimum of five hosts per hospital for random workloads: 1) application server, 2) database server, 3) storage server, 4) network services server, and 5) backup/disaster recovery server. Table 6 shows the various hosts and VMs considered for two distinct cases.

Table 6. The number of hosts and VMs for Random workloads in two distinct cases

If a minimum of 5 hosts are needed per hospital (Virtualized)					
Case 1: if 1Vm per host (possible)			Case 2: If 2VMs per host (possible)		
No. of Hospitals	Hosts	VMs	No. of Hospitals	Hosts	VMs (VM instances)
11	55	55	11	28	55
353	1765	1765	353	884	1765

The number of VMs per host depends on host specifications (CPU, RAM, storage, bandwidth), VM resource requirements (vCPUs, RAM, storage, bandwidth), and the hypervisor type. Studies [5,14] revealed that hypervisors and CloudSim consume system resources, reducing CPU and RAM for VMs.

4.2. General Simulation Steps and Pseudo Code

In CloudSim, the data center (DC) acts as an IaaS provider, housing multiple PMs that run several VMs. Data centers, brokers, and cloud information services (CIS) are interconnected entities. Fig.4 depicts the basic CloudSim framework during simulation, where the DC and its characteristics are registered in the CIS. When a cloudlet is submitted to a broker, the broker interacts with the CIS and the datacenter to assign the task to a selected VM based on the VM allocation [14], and selection policies. Depending on VM policies, a single VM can execute multiple cloudlets.

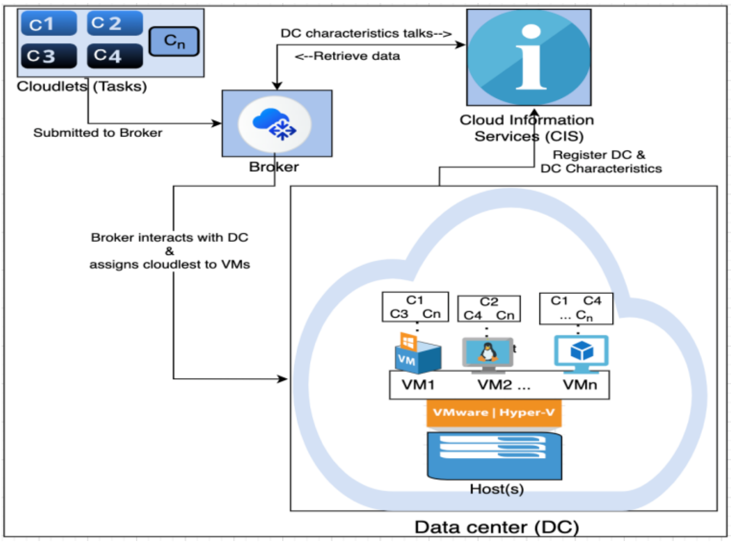

Fig.4. Interactions in the basic CloudSim Framework during simulation

Fig.5 illustrates the CloudSim simulation process: configure the local environment (e.g., IDE), initialize CloudSim, and define a data center with resources (hosts, PEs, RAM, storage, bandwidth). Instantiate a broker to manage VMs and cloudlets, create and submit VMs to the broker. Define and submit cloudlets with attributes to the broker, which assigns them to appropriate data centers. After running the simulation, stop it and retrieve results for analysis.

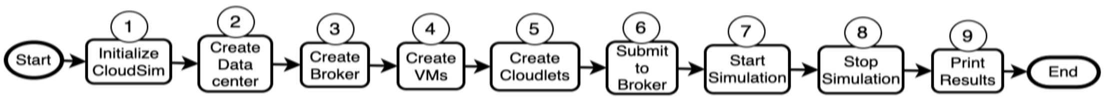

Fig.5. The general steps of the CloudSim simulation process

```
BEGIN CloudSim_Simulation {
int num_user = 1; Calendar calendar = Calendar.getInstance(); boolean trace_flag = false;
        CloudSim.init(num_user, calendar, trace_flag);
        HOST = new Host(hostCharacteristics);
Datacenter datacenter = create Datacenter("Datacenter");
DatacenterBroker broker = createBroker();
        vmlist = new ArrayList<Vm>();
        Vm vm = new Vm(vmCharacteristics);
                vmlist.add(vm);-
                broker.submitVmList(vmlist);
        cloudletList = new ArrayList<Cloudlet>();
```

```
        Cloudlet cloudlet = new Cloudlet(Cloudlet properties);
        cloudlet.setUserId(brokerId);
        cloudletList.add(cloudlet);
        broker.submitCloudletList(cloudletList);
CloudSim.startSimulation();
CloudSim.stopSimulation();
Results = broker.getCloudletReceivedList();
        PrintResults(Results);
} END CloudSim_Simulation.
```

5. Results and Discussions

Using PlanetLab and random workloads, we simulated all combinations of five host overloading detection algorithms [5,14,24] VM allocation policy and four VM selection policies, resulting in 20 combinations. Also, we tested the DVFS as the state-of-the-art algorithm and NPA algorithms on both workloads, yielding the following results:

5.1. Using PlanetLab Workloads

We evaluated 20 algorithms (5 VM allocation policies × 4 VM selection policies) across 10 workload traces, resulting in 200 simulations (20 algorithms × 10 traces) in CloudSim. The simulation parameters include the number of hosts, VMs, energy consumption, VM migrations, SLA, average SLAVs, VM processing power (MIPS), and task completion time. This section uses various VMs, as shown in Fig.6, while maintaining a constant number of 800 hosts.

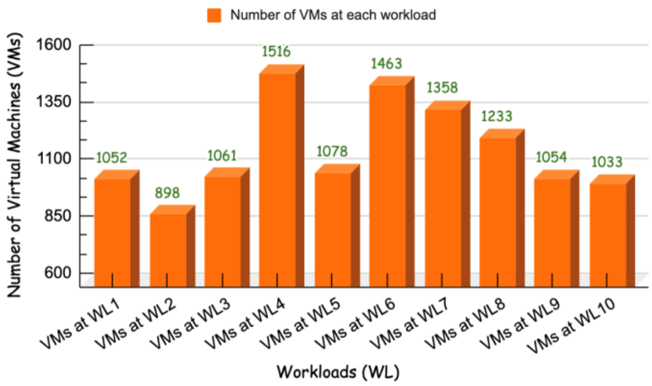

Fig.6. The number of VMs used in each PlanetLab workload

Energy consumption across all workloads using all power-aware algorithms together: In our experiments, all power-aware algorithms show the highest energy consumption at workload six (WL6), as shown in Fig.7. WL6 has the highest VM migrations and contains 1,463 VMs, making it the second-largest in the PlanetLab workload. This experiment shows that both the number of VMs and VM migrations significantly increase energy consumption.

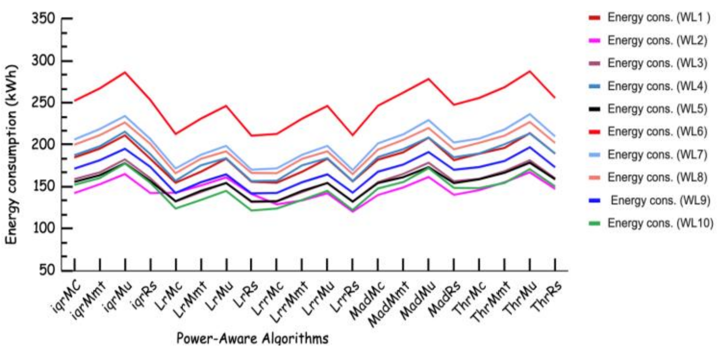

Fig.7. PA- algorithms energy-consumption at all workloads

Energy consumption across all workloads using all power-aware algorithms together: In our experiments, all power-aware algorithms show the highest energy consumption at workload six (WL6), as shown in Fig. 7. WL6 has the highest VM migrations and contains 1,463 VMs, making it the second-largest in the PlanetLab workload. This experiment shows that both the number of VMs and VM migrations significantly increase energy consumption.

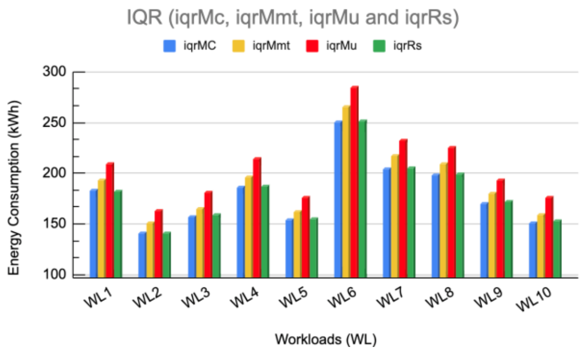

Fig.8. IQR across all workloads

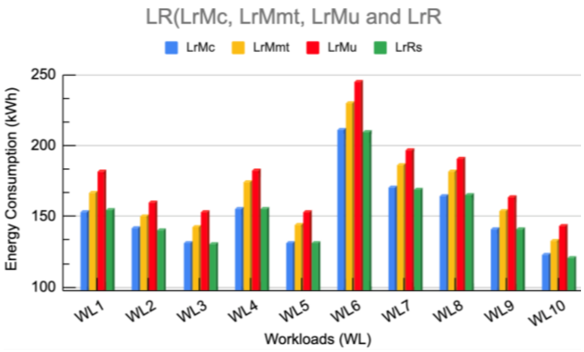

Fig.9. LR across all workloads

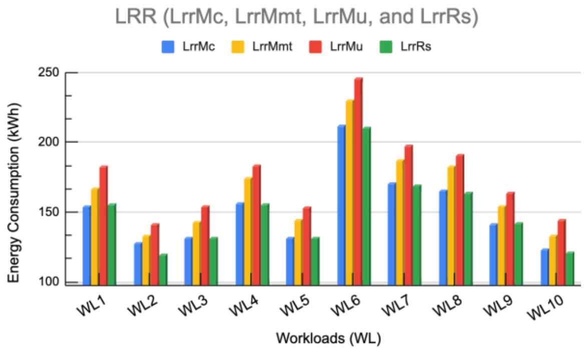

Fig.10. LRR across all workloads

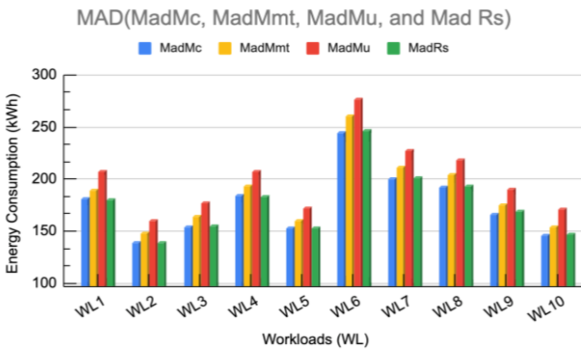

Fig.11. MAD across all workloads

Experimental results (Fig. 8–12) show that energy consumption is highest in workload six (WL6) across all algorithms, primarily due to two factors: (1) a high number of VM migrations and (2) the number of VMs. In contrast, workload two (2) exhibits low energy consumption, associated with its lower number of VMs and VM migrations, indicating that both factors significantly impact energy consumption.

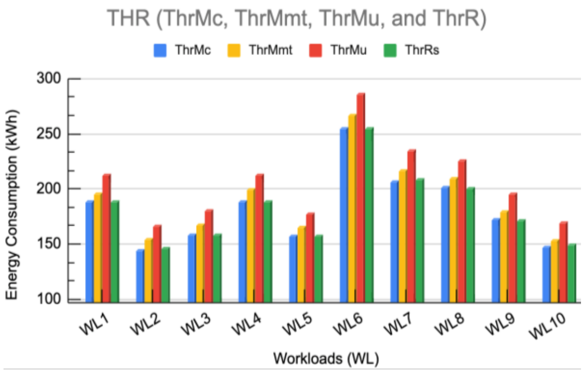

Fig.12.THR across all workloads

Energy consumption across selected workloads using all power-aware algorithms: We selected five workloads (WL1, WL2, WL4, WL6, and WL8) based on the numbers of VMs and average VM migrations for further simulations, as shown in Table 7. These workloads are used to simulate VM migrations and analyze the correlation between VM migrations, energy consumption, SLA, and average SLAVs. Studies [37,38] suggest that during VM migrations, the number of migration operations can exceed the actual number of VMs on hosts due to factors like multiple migration phases, iterative migrations, multiple network paths, and partial migrations or failures.

Table 7. Selected workloads based on the number of VMs and average VM migrations

In terms of VM numbers:			In terms of average VM Migrations:		
Workload (WL)	Number of VMs	Remarks	Workload (WL)	Avg. no. of VM migrations	Remarks
WL2	898	Minimum	WL2	21144.1	Minimum
WL8	1233 (approx.)	Average	WL1	26610.23	Average
WL4	1516	Maximum	WL6	36296.3	Maximum
			WL 4	28685.65	Unclassified

The experimental simulation results show a significant difference in energy consumption between workload six (WL6) and workload two (WL2). As illustrated in Fig.13, workload two (WL2) exhibits low energy consumption, while workload six (WL6) shows high energy consumption across all PA algorithms. WL2 has a minimum of 898 VMs, compared to 1,463 VMs at WL6. Additionally, WL2 has fewer VM migrations, whereas WL6 has the highest VM migrations, as depicted in Fig.14. WL4 has the highest number of VMs (1516) but has fewer VM migrations and lower energy consumption than WL6.

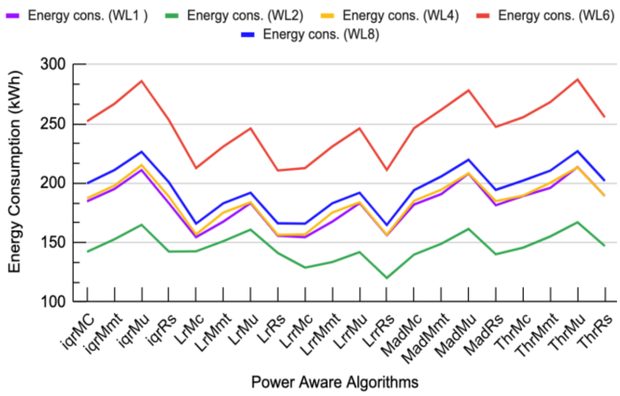

Fig.13. Energy consumption across selected workloads

The results (Fig.13 - Fig.15) clearly demonstrate that both the number of VMs and VM migrations affect the amount energy consumption. The simulation scenarios indicate that energy consumption increases with the number of VMs and is further amplified by VM migrations across all power-aware algorithms. However, our extensive simulations show that the LR and LRR algorithms are more energy-efficient than the others.

To validate the reliability of the results, we conducted a statistical test to analyze the impact of independent variables (VM numbers and VM migrations) on the dependent variable (energy consumption in kWh), with a 95% confidence level and a 0.05 significance level.

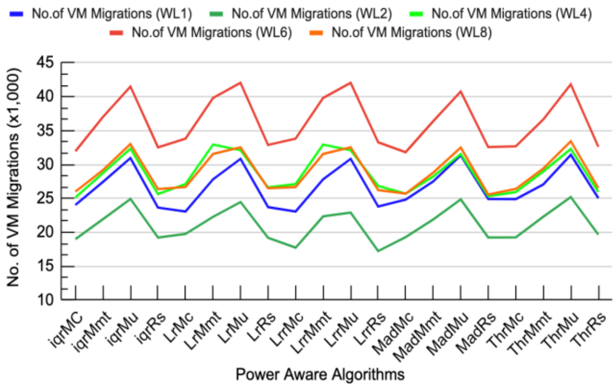

Fig.14. VM migrations across selected workloads

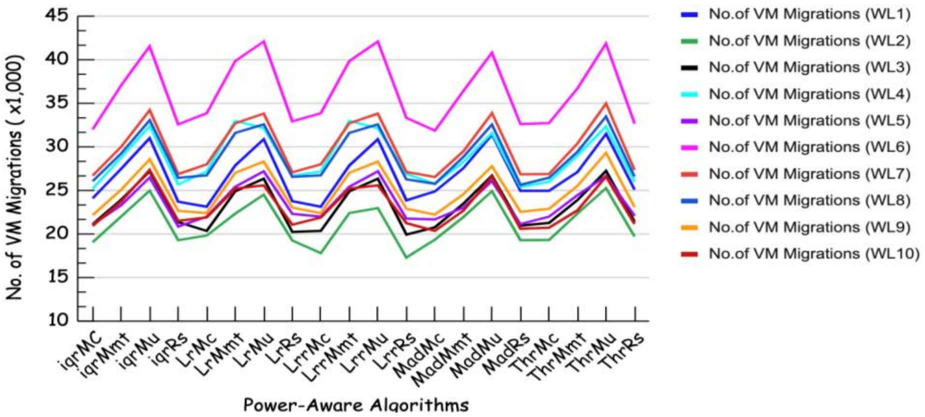

Fig.15. VM migrations across all workloads

Table 8. Tests of between-subjects effects

Dependent Variable: Energy Consumption

Source	Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared
Corrected Model	115940.549[a]	5	23188.110	88.509	.000	.825
Intercept	12253.257	1	12253.257	46.771	.000	.332
VM Migrations	8601.262	1	8601.262	32.831	.000	.259
VM Numbers	10567.899	4	2641.975	10.084	.000	.300
Error	24626.566	94	261.985			
Total	3809236.167	100				
Corrected Total	140567.115	99				

a. R Squared = .825 (Adjusted R Squared = .815)

Based on the results in Table 8, the p-values (sign.) for both VM numbers and migrations were below 0.05, suggesting that both factors have a statistically significant effect on energy consumption.

DVFS and NPA algorithms: In DVFS, as shown in Fig.16, an increase in VMs is associated with higher energy consumption, while a decrease in VMs leads to reduced energy consumption, i.e., dynamically adjusting energy consumption based on the number of VMs. Energy consumption is affected by the number of VMs while maintaining a constant number of hosts. DVFS considers the energy consumption of VMs, illustrating a direct proportional relationship between the number of VMs and energy consumption. Conversely, our experiment results show that the NPA algorithm consistently consumes the maximum energy, regardless of the number of VMs, while maintaining a constant number of hosts, as shown in Fig.17. The NPA policy ensures that all hosts operate at their peak power levels, consuming the highest amount of energy. This indicates that the NPA algorithm focuses on host energy consumption rather than VMs and hence they are not green in the contextual scenario.

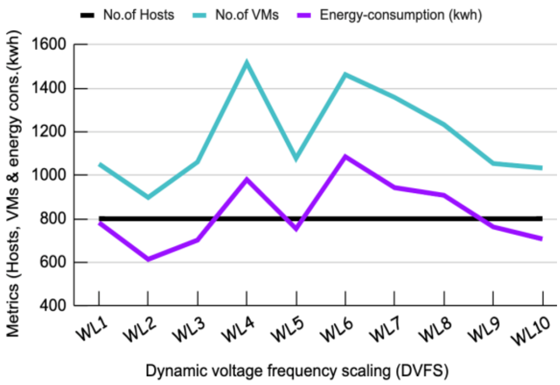

Fig.16. DVFS across all workloads

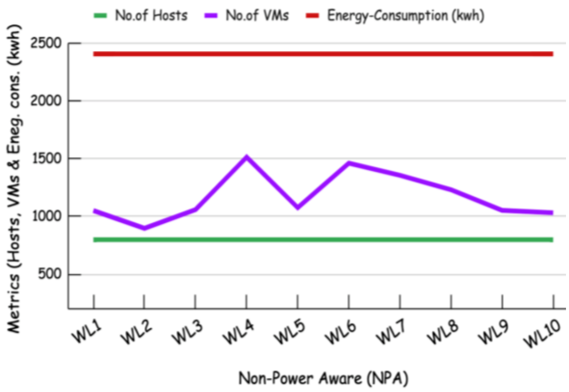

Fig.17. NPA across all workloads

Based on extensive experimental analysis, this study confirms that power-aware algorithms optimize energy consumption by: i) dynamically adjusting resource allocation, ii) shutting down idle VMs and hosts, and migrating workloads to balance host utilization. DVFS targets the VM's energy consumption, while NPA algorithms focus on host energy consumption. In all scenarios— energy consumption across all workloads with all PA algorithms together (Fig.7.), across all workloads with each PA algorithms separately (Fig.9 and Fig.10), and across selected workloads with all PA algorithms (Fig.13)—LR/LRR-based power-aware algorithms consume less energy and outperform the others.

SLA and SLAVs: In CloudSim, SLAs define service performance and resource metrics, such as availability and energy efficiency, enabling the simulation of resource management strategies to analyze performance-energy trade-offs. Fig.18 shows that SLAs are defined for each workload, with workload 10 having a notably higher SLA, while the SLAs for the other workloads are closely aligned. However, SLAV [9,11] occurs when a provider fails to meet the SLA or a customer breaches it, potentially leading to penalties, compensation claims, or a reassessment of the service relationship, affecting trust and customer satisfaction. Our simulation confirmed SLAVs in each workload (Fig.19), with workload 10 showing a notably higher average SLAV than the others.

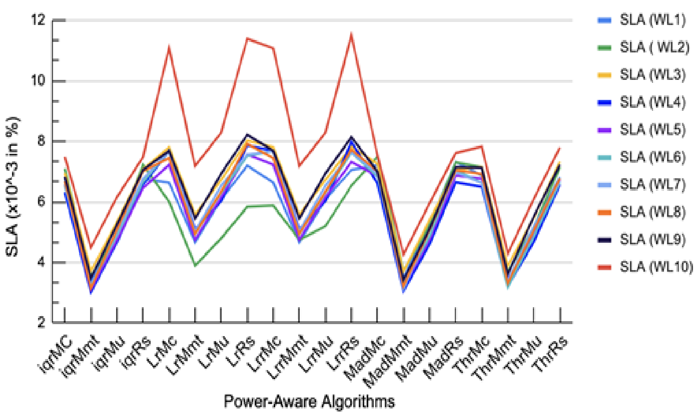

Fig.18. SLAs for PA algorithms across all workloads

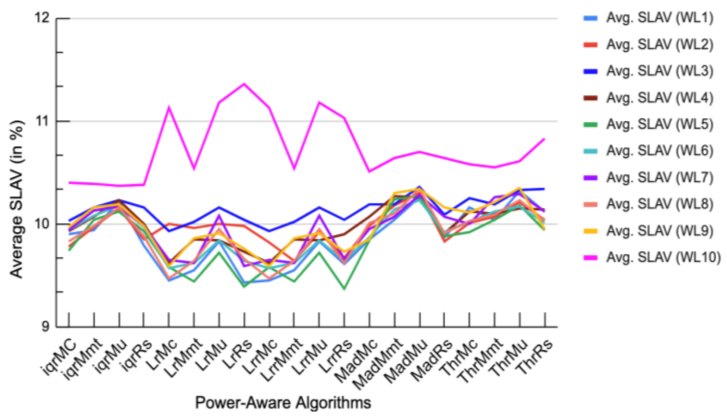

Fig.19. SLAVs of PA algorithms across all workloads

The results reveal that the Minimum Utilization (Mu) VM selection policy (iqrMu, LrMu, LrrMu, MadMu, ThrMu) resulted in high average SLAVs across all PA algorithms (Fig.19). However, SLAV is lower in LR/LRR-based algorithms than in others for all PlanetLab workloads, except workload 10, where SLAV is significantly higher.

Correlation: The selected workloads (Table 7) are used in experimental simulations to analyze the correlation between VM migrations, energy consumption, SLA, and average SLAV. A positive correlation means variables move in the same direction, while a negative correlation means they move in opposite directions. The experimental results (Table 9) show a positive correlation between VM migrations and energy consumption, with energy consumption increasing or decreasing in sync with VM migrations. Notably, workload 2 shows a strong positive correlation (0.845) between VM migrations (VMM) and energy consumption, while weak positive correlation exists between VMM at Workload 4 and energy consumption at Workload 8. In most cases, the correlation is moderate, indicating that both variables increase or decrease together.

Table 9. Correlation between VMM and Energy consumption across selected workloads

	VMM (WL1)	Energy (WL1)	VMM (WL2)	Energy (WL2)	VMM (WL4)	Energy (WL4)	VMM (WL6)	Energy (WL6)	VMM (WL8)	Energy (WL8)
VMM (WL1)	1									
Energy (WL1)	0.679	1								
VMM (WL2)	0.958	0.685	1							
Energy (WL2)	0.74	0.804	0.845	1						
VMM (WL4)	0.868	0.311	0.862	0.521	1					
Energy (WL4)	0.677	0.991	0.698	0.801	0.342	1				
VMM (WL6)	0.942	0.443	0.921	0.617	0.972	0.457	1			
Energy (WL6)	0.651	0.995	0.672	0.798	0.299	0.997	0.424	1		
VMM (WL8)	0.949	0.502	0.942	0.664	0.97	0.518	0.993	0.485	1	
Energy (WL8)	0.636	0.992	0.661	0.793	0.291	0.997	0.409	0.999	0.474	1

Our simulations, presented in Table 10 and Fig.20, reveal a positive correlation among VM migrations, energy consumption, and average SLAV, while SLA shows a negative correlation with these variables. Accordingly:

- Energy consumption and average SLAV at WL6 have a strong positive correlation of 0.956.
- VM migrations and energy consumption at WL2 also show a strong positive correlation of 0.845.
- Average SLAV and VM migrations at WL2 exhibit a strong positive correlation of 0.629.

In contrast:

- SLA and energy consumption at WL6 have a strong negative correlation of -0.665.
- SLA at WL2 and VM migrations at WL6 show a strong negative correlation of -0.695.

- SLA and average SLAV at WL6 have a strong negative correlation of -0.614, while SLA at WL2 and average SLAV at WL1 show a very weak negative correlation of -0.055.

Table 10. Correlation between VMM, energy consumption, SLA, and average SLAV across selected workloads

	VMM (WL1)	*Energy (WL1)*	*SLA (WL1)*	*Avg. SLAV (WL1)*	*VMM (WL2)*	*Energy (WL2)*	*SLA (WL2)*	*Avg. SLAV (WL2)*	*VMM (WL6)*	Energy (WL6)	*SLA (WL6)*	*Avg. SLAV (WL6)*
VMM (WL1)	1											
Energy (WL1)	0.679	1										
SLA (WL1)	-0.541	-0.442	1									
Avg. SLAV (WL1)	0.557	0.955	-0.303	1								
VMM (WL2)	0.958	0.685	-0.624	0.533	1							
Energy(WL2)	0.74	0.804	-0.495	0.698	0.845	1						
SLA (WL2)	-0.585	-0.213	0.91	-0.055	-0.649	-0.448	1					
Avg. SLAV (WL2)	0.548	0.682	-0.382	0.674	0.629	0.827	-0.338	1				
VMM (WL6)	0.942	0.443	-0.569	0.29	0.921	0.617	-0.695	0.39	1			
Energy(WL6)	0.651	0.995	-0.484	0.939	0.672	0.798	-0.236	0.662	0.424	1		
SLA (WL6)	-0.588	-0.628	0.96	-0.503	-0.657	-0.592	0.798	-0.465	-0.527	-0.665	1	
Avg. SLAV (WL6)	0.508	0.955	-0.42	0.951	0.521	0.71	-0.157	0.69	0.244	0.956	-0.614	1

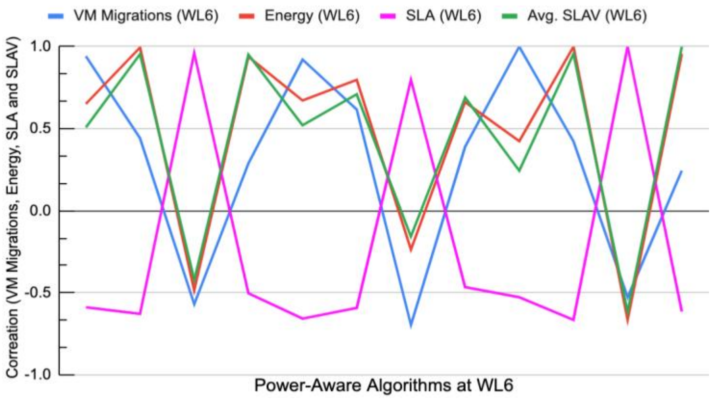

Fig.20. Correlation among VM migrations, energy consumption, SLA, and avg. SLAV at WL6

SLA violations can jeopardize the security and privacy of individuals' healthcare data. SLAVs may disrupt critical healthcare operations, leading to disastrous outcomes in emergencies, delays in patient care, increased operational costs for hospitals, reduced patient trust, delays in accessing patient data and medical services, and potential additional security risks. Thus, balancing VM migrations, energy efficiency and performance while regularly reviewing and ensuring SLA compliance is essential for successful cloud operations in healthcare systems.

VM's processing power, Cloudlets, and Task completion time: The processing power (MIPS) and task completion time are key metrics for evaluating computing performance in cloud simulations. MIPS measures CPU speed, while task completion time indicates the total time to finish a workload in seconds [4,17].

Based on experimental results, variations in processing power (MIPS) and Cloudlet length affect task completion time. When both VM processing power and Cloudlet length decrease (Fig.21), task completion time increases exponentially. Likewise, a decrease in MIPS with an increase in Cloudlet length also leads to exponential increases in completion time (Fig.22).

Conversely, increasing processing power while decreasing Cloudlet length leads to exponential decreases in completion time (Fig.23). When both processing power and Cloudlet length increase, task completion time decreases (Fig.24.). Thus, these results show that both processing power and Cloudlet length affect task completion time; however, processing power has a significant influence.

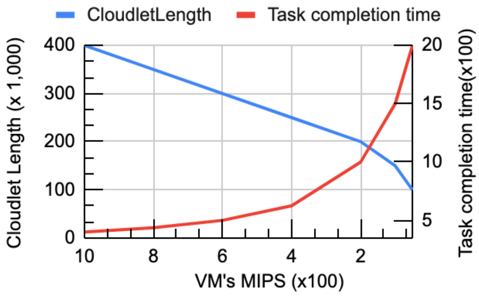

Fig.21.Task completion time increases exponentially as both MIPS and Cloudlet length decrease

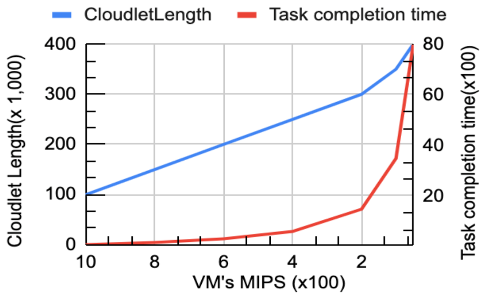

Fig.22. Decreasing MIPS and increasing Cloudlet length lead to exponential increases in completion time

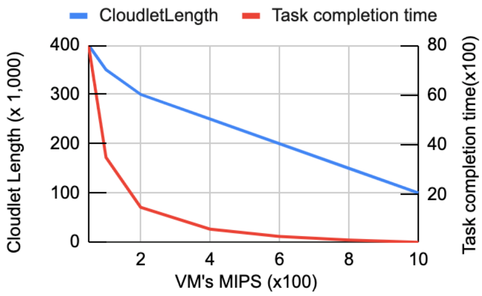

Fig.23. Increasing MIPS and decreasing Cloudlet length result in exponential decreases in completion time

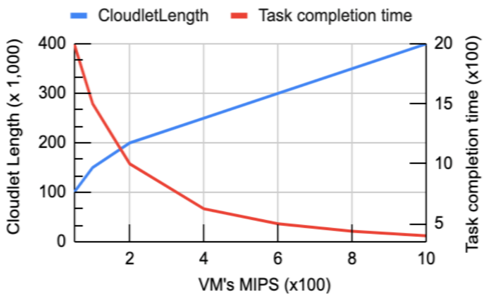

Fig.24. Task completion time decreases exponentially as both MIPS and Cloudlet length increase

When assigning tasks to VMs, it's crucial to consider the VM's MIPS, ensuring that the hosts' MIPS exceed the VMs' MIPS to avoid automatic VM shutdowns. Task completion time is directly proportional to Cloudlet length: as Cloudlet length increases, task completion time rises slowly, and vice versa. In contrast, as MIPS increases, task completion time decreases exponentially and vice versa, indicating an inversely proportional relationship.

5.2. Using Random Workloads

We evaluated 20 algorithm combinations (5 VMs allocation policies × 4 VMs selection policies) across 4 random workloads, totaling 80 simulations (20 x 4) in CloudSim. This section simulates PA, DVFS, and NPA algorithms with varying numbers of hosts and VMs, as detailed in Table 6. As shown in Fig.25, with 55 hosts and VMs, the highest energy consumption is 53.12 kWh at IqrMu (minimum utilization), while the lowest is 36.75 kWh at LrMc and LrrMc (maximum correlation). Likewise, with 1,765 hosts and VMs, the highest recorded energy consumption is 1,672.29 kWh at IqrMu, and the lowest is 1,111.85 kWh at LrrRs, as depicted in Fig.26.

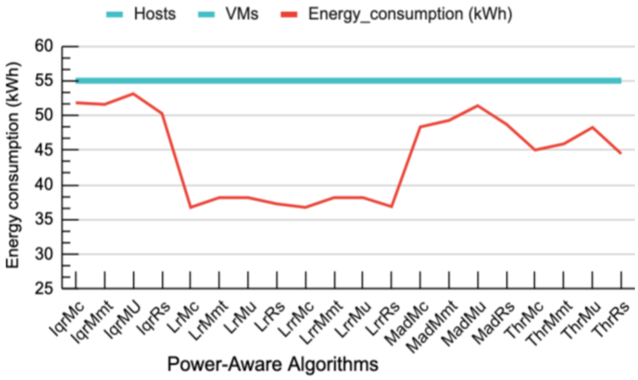

Fig.25. Energy consumption with the same number of hosts and VMs (small scale)

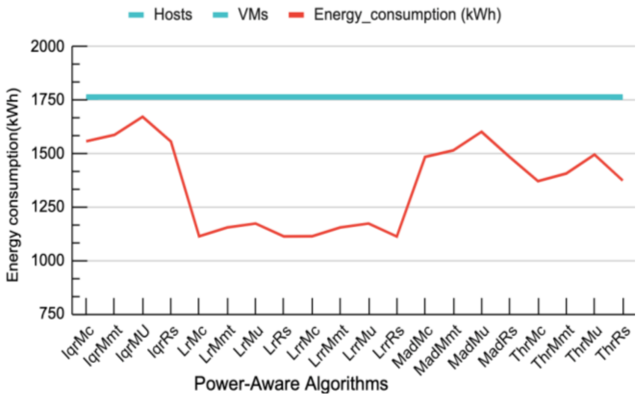

Fig.26. Energy consumption with the same number of hosts and VMs (relatively large scale)

As illustrated in Fig. 27 and Table 11, with 28 hosts supporting 55 VMs, the highest energy consumption is 50.3 kWh under IqrMu, while the lowest is 35.54 kWh for LrMc and LrrMc. When using a relatively large and varied number of hosts and VMs (i.e., 884 hosts hosting 1,765 VMs), the highest recorded energy consumption is 1,566.9 kWh under IqrMu, while the lowest is 1,093.55 kWh under LrrRs, as shown in Fig. 28 and Table 12.

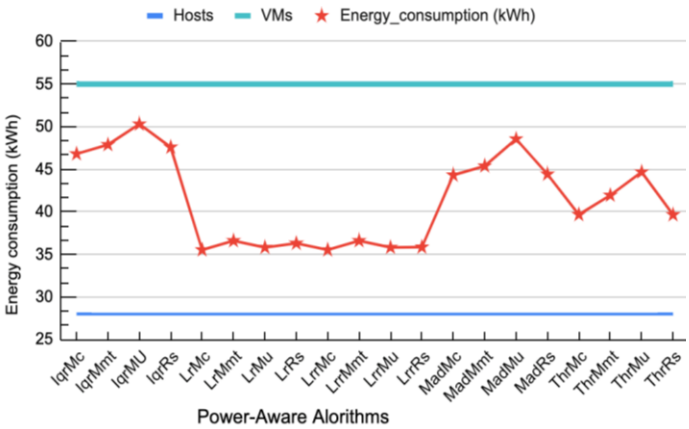

Fig.27. Energy consumption with varying numbers of hosts and VMs (small scale)

Fig. 27 and Table 10 show that with 55 VMs, the LrMc and LrrMc algorithms have the lowest energy consumption, while IqrMu has the highest. LrMc and LrrMc are 29.36% more energy-efficient compared to IqrMu. When the number of VMs increases from 55 to 1,765, the LrrRs algorithm outperform, saving 30.20% energy compared to IqrMu, as illustrated in Fig. 28 and Table 11. Our experiments demonstrate that Lr/Lrr-based power-aware algorithms reveal less energy consumption than other power-aware algorithms, and are categorized under the top five (5) in ranking these algorithms, as presented in Tables 10 and 11. This paper focuses on 20 intrinsic PA algorithms aimed at energy efficiency, as shown in Tables 10 and 11, and uses DVFS as the state-of-the-art algorithm in CloudSim to validate the superiority of the optimal algorithms. Numerous studies [39,40] have explored other state-of-the-art scheduling algorithms (extrinisic), but their practical scenarios, application areas, simulation environment, and objectives differ, such as energy consumption reduction, minimize average execution time, cost reduction, or minimizing schedule length, etc. We acknowledge this limitation, as our study is confined to intrinsic algorithms; however, extrinsic algorithms can be explored in future work and experimentation.

Table 11. Energy consumption with small host and VM numbers

Power-Aware Algorithms	Hosts	VMs	Energy_consumption (kWh)
LrMc	28	55	35.54
LrrMc	28	55	35.54
LrMu	28	55	35.83
LrrMu	28	55	35.83
LrrRs	28	55	35.86
LrRs	28	55	36.31
LrMmt	28	55	36.61
LrrMmt	28	55	36.61
ThrRs	28	55	39.66
ThrMc	28	55	39.68
ThrMmt	28	55	41.95
MadMc	28	55	44.32
MadRs	28	55	44.42
ThrMu	28	55	44.68
MadMmt	28	55	45.39
IqrMc	28	55	46.81
IqrRs	28	55	47.59
IqrMmt	28	55	47.89
MadMu	28	55	48.56
IqrMU	28	55	50.31

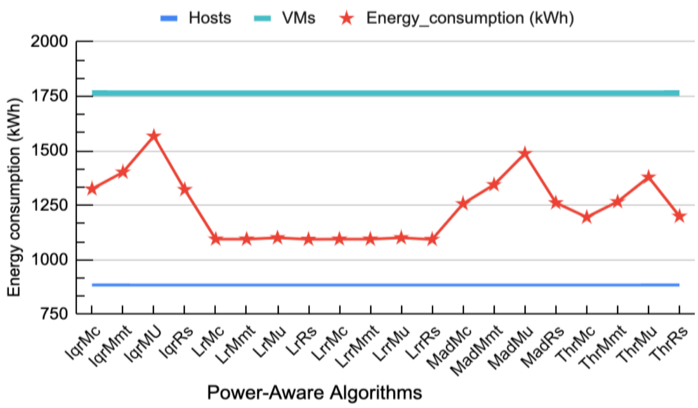

Fig.28. Energy consumption with varying numbers of hosts and VMs (relatively large scale)

In the case of DVFS, the experimental results in Fig. 29 and Fig. 30 clearly show that energy consumption remains the same as the number of hosts varies (55, 1765, 28, and 884) while keeping the number of VMs fixed (55 and 1765). This indicates that DVFS considers the energy consumption of VMs instead of hosts. In contrast, when the number of hosts varies while keeping the number of VMs the same in both cases, energy consumption fluctuates; specifically, it increases with more hosts and decreases with fewer hosts, as shown in Fig. 31 and Fig. 32. Nevertheless, the NPA algorithm utilizes the maximum energy or power levels.

Table 12. Energy consumption with large numbers of hosts and VMs

Power-Aware Algorithms	Hosts	VMs	Energy_consumption (kWh)
LrrRs	884	1765	1093.55
LrRs	884	1765	1094.38
LrMmt	884	1765	1094.82
LrrMmt	884	1765	1094.82
LrMc	884	1765	1095.13
LrrMc	884	1765	1095.13
LrMu	884	1765	1100.9
LrrMu	884	1765	1100.9
ThrMc	884	1765	1195.47
ThrRs	884	1765	1199.82
MadMc	884	1765	1257.16
MadRs	884	1765	1261.75
ThrMmt	884	1765	1266.49
IqrRs	884	1765	1322.13
IqrMc	884	1765	1324.95
MadMmt	884	1765	1344.5
ThrMu	884	1765	1379.49
IqrMmt	884	1765	1402.14
MadMu	884	1765	1487.59
IqrMU	884	1765	1566.9

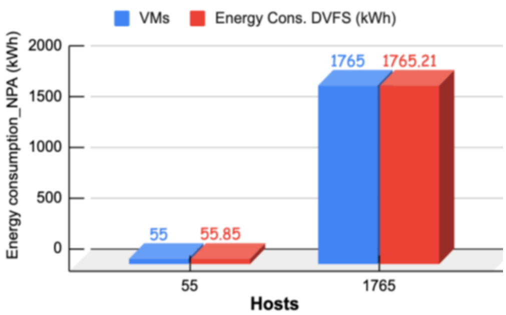

Fig.29. DVFS energy consumption when the number of hosts and VMs is equal

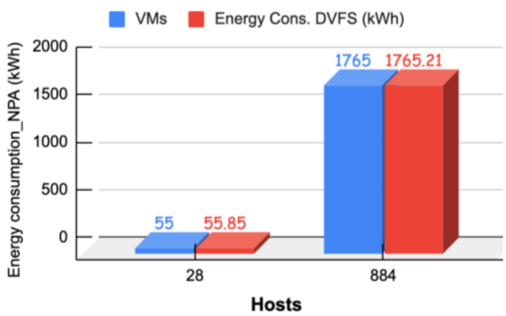

Fig.30. DVFS energy consumption when the number of hosts and VMs differ

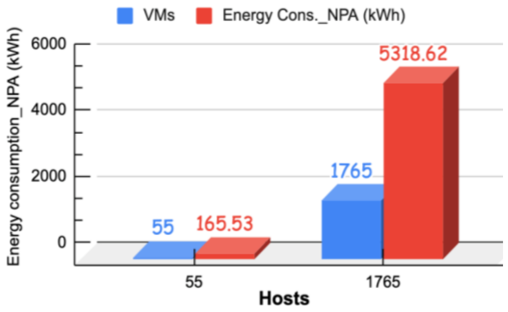

Fig.31. NPA energy consumption when the number of hosts and VMs is equal

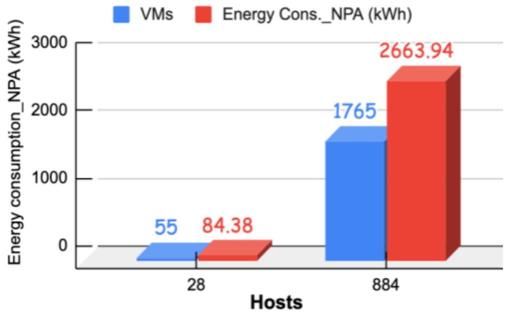

Fig.32. NPA energy consumption when the number of hosts and VMs differ

Thus, power-aware algorithms prioritize the energy consumption of VMs rather than hosts. Our experiment results revealed that DVFS emphasizes the energy consumption of VMs to enhance energy efficiency, whereas the NPA algorithm depends on the number of hosts, potentially resulting in maximum energy consumption.

In the context of Ethiopia as a case context, the hospitals have numerous limitations with existing information system management. In the current state of art practices, the paper-based methods are not only error-prone, but fragmented, and inaccessible as well. The independent operations of traditional IT data centers lead to high costs and significant energy consumption. Numerous studies show that cloud data centers provide a more sustainable and cost-effective alternative. Implementing a cloud data center with virtualization for hospitals significantly differs from traditional on-premises infrastructure, offering lower initial investment (OpeX), greater flexibility, and scalability while minimizing the need for multiple PMs. By consolidating resources, hospitals can lower costs, reduce energy consumption, and enhance data interoperability and collaboration, ultimately improving patient care. Overall, a cloud-based approach can be explored as an alternative option for energy saving.

The main contributions of this study are as follows:

- Advocating for cloud computing: highlights the crucial role of cloud-based solutions in developing sustainable healthcare information systems within resource-constrained environments like Ethiopian public hospitals.
- Evaluation of Energy-efficiency and VM behavior in cloud-based healthcare systems: Through simulations of cloud infrastructure with varied workloads and PA algorithms, the study provides key insights into optimizing energy consumption in the healthcare sector via cloud computing technologies.
- Analyzing the impact of VM Migrations: It examines how VM migrations across different workloads influence energy consumption and the performance.
- Identifying optimal energy-saving algorithms: It conducts empirical analyses of various energy-saving techniques, including PA algorithms, DVFS, and NPA algorithms, to identify effective strategies for minimizing energy consumption in healthcare clouds.

However, the paper does not explicitly address the cost of transitioning to cloud-based healthcare systems in the Ethiopian context. Current challenges include managing hardcopy records, scattered data, and security concerns. The transition demands substantial investment in infrastructure, training, and data migration. Future work can focus explicitly on cost-benefit analysis during real-world deployment phases.

6. Conclusions

Cloud-based systems in data centers substantially lower infrastructure costs, carbon emissions, and energy consumption. PA VM allocation algorithms improve resource utilization and energy efficiency. Balancing VM migrations is crucial for maintaining energy efficiency and performance while ensuring SLA compliance for effective cloud operations in healthcare systems. Experimental results demonstrate that energy consumption increases with the number of VMs and VM migrations, with higher migrations leading to higher energy consumption. Among other PA algorithms, Lr/Lrr-based methods were more energy-efficient for the tested workloads. In random workloads, LrMc and LrrMc were 29.36% more energy-efficient than IqrMu with 55 VMs, while LrrRs saved 30.20% more energy than IqrMu with 1,765 VMs. DVFS saves energy by dynamically adjusting consumption based on the number of VMs, whereas NPA algorithms utilize the maximum host power regardless of VM count. The Minimum Utilization (Mu) VM selection policy achieves high average SLAVs across all PA algorithms. VM migrations significantly impact energy consumption and SLA performance, with positive correlations to energy consumption and average SLAV, and a negative correlation with SLA. Task completion time is influenced by VM processing power (MIPS) and cloudlet length, with higher MIPS reducing completion time. It is inversely proportional to MIPS and directly proportional to cloudlet length. Cloud data centers with virtualization can offer hospitals lower initial investment, greater scalability, energy efficiency, improved data interoperability, reduced reliance on multiple PMs, and ultimately enhanced patient care. Power-aware techniques in cloud-based solutions can reduce energy consumption and enhance healthcare information systems. Adopting cloud-based systems in Ethiopian public hospitals could enhance accessibility and address existing systems deficiencies. Although focused on Ethiopian, the findings on cloud-based systems, energy efficiency, and PA algorithms are relevant to global healthcare settings. Future research can explore integrating cloud-based systems with national data repositories.

Actionable recommendations and suggested roadmap:

Actionable recommendation for policymakers or healthcare organizations

- Design a federal-level cloud service policy for the migration and unionization of healthcare institutions' information services for improved security, privacy, trust, performance, energy efficiency, interoperability of services, etc.
- Developing an adoption strategy for a cloud-based healthcare solution frameworks or models in public hospitals. Policymakers should prioritize the adoption of cloud-based systems in healthcare settings and encourage the transition to cloud-based systems to minimize infrastructure costs and improve healthcare service delivery through better data management and accessibility.
- Planning investments in cloud-based infrastructures for healthcare: Prioritize investments in national cloud data centers that ensure scalability and enhance interoperability to effectively address the increasing demands of healthcare systems.
- A Shift from CapEx to OpEx: The Ethiopian government can establish a unionized cloud data center at the federal level, allowing public healthcare institutions to subscribe to cloud-based virtualized data centers under the OpEx model
- Implement and evaluate the proposed solutions using power-aware algorithms and DVFS: To achieve smart and energy-efficient healthcare systems, organizations should adopt power-aware (PA) algorithms and DVFS techniques to optimize resource utilization. Policymakers should actively support the development and deployment of these solutions.
- Integrate cloud solutions with emerging digital health interventions using standards: Ensure seamless integration of cloud solutions with other digital health technologies by adopting standardized protocols.

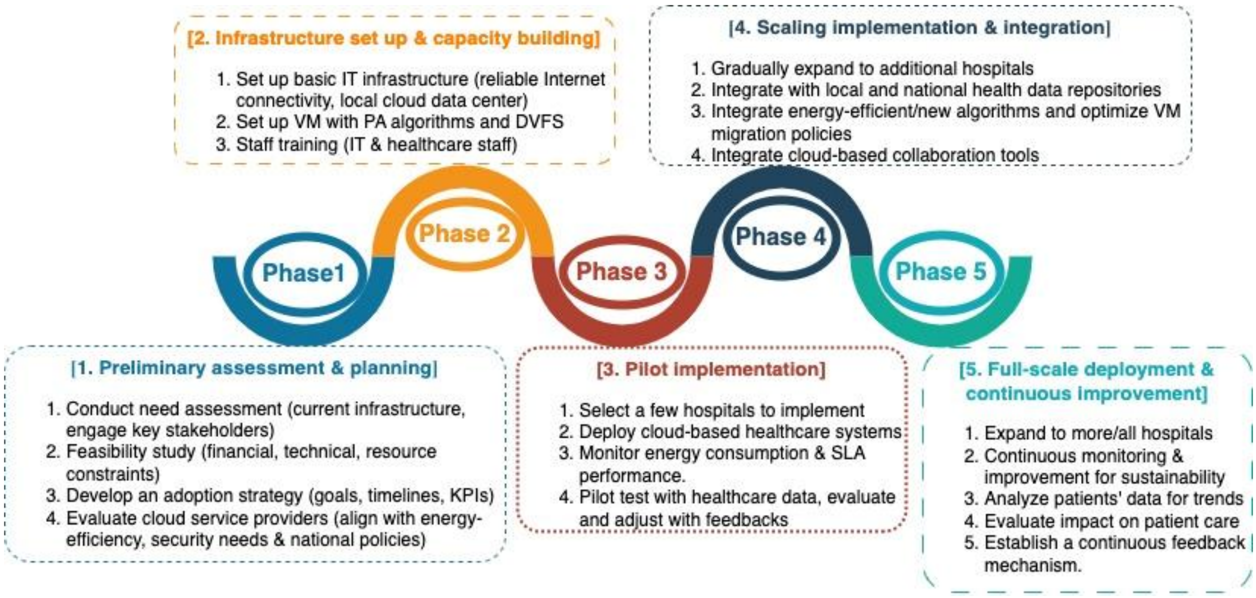

Fig.33. A roadmap for adopting cloud-based healthcare systems in ethiopian public hospitals

Suggested a roadmap for adopting cloud-based systems in Ethiopian public hospitals

We suggested a five-phase roadmap for adopting cloud-based systems in Ethiopian public hospitals, focusing on key milestones from preliminary assessment and planning to full-scale deployment and continuous improvement, as detailed in Fig.33.

Ethical Considerations

Adopting cloud-based healthcare information systems, particularly in settings like Ethiopian public hospitals, offers numerous potential benefits. However, careful attention to the ethical issues related to data privacy and security, equity in access to technology, algorithmic fairness, accountability and transparency, stakeholder engagement and environmental impact is crucial to ensure that cloud solutions are implemented in a manner that is beneficial, sustainable, and aligned with the values of both healthcare providers and patients. Ethical principles must guide the design, deployment, and operation of cloud-based systems to ensure they enhance the healthcare ecosystem, minimize harm, and promote fairness.

References

[1] A. Arega and D. P. Sharma, "Towards Smart and Green Features of Cloud Computing in Healthcare Services: A Systematic Literature Review," J. Inf. Syst. Eng. Bus. Intell., vol. 9, no. 2, pp. 161–180, Nov. 2023, doi: 10.20473/jisebi.9.2.161-180.

[2] N. Chauhan, N. Kaur, K. S. Saini, S. Verma, R. A. Khurma, and P. A. Castillo, "Maximizing Resource Efficiency in Cloud Data Centers through Knowledge-Based Flower Pollination Algorithm (KB-FPA)," Comput. Mater. Contin., vol. 79, no. 3, pp. 3757–3782, Jun. 2024, doi: 10.32604/cmc.2024.046516.

[3] A. V. D. K. Amol, "Simple and efficient duelist algorithm variations for energy-aware virtual machine placement in cloud data centers ," Decis. Sci. Lett., vol. 13, no. 3, pp. 751–766, Mar. 2024, doi: 10.5267/dsl.2024.3.004.

[4] S. Mehmi, H. K. Verma, and A. L. Sangal, "Simulation modeling of cloud computing for smart grid using CloudSim," J. Electr. Syst. Inf. Technol., vol. 4, no. 1, pp. 159–172, May 2017, doi: 10.1016/j.jesit.2016.10.004.

[5] Nagma, J. Singh, and J. Sidhu, "Comparative analysis of VM consolidation algorithms for cloud computing," Procedia Comput. Sci., vol. 167, pp. 1390–1399, 2020, doi: 10.1016/j.procs.2020.03.350.

[6] A. Abdelsamea, A. A. El-Moursy, E. E. Hemayed, and H. Eldeeb, "Virtual machine consolidation enhancement using hybrid regression algorithms," Egypt. Informatics J., vol. 18, no. 3, pp. 161–170, Nov. 2017, doi: 10.1016/j.eij.2016.12.002.

[7] D. A. Shafiq, N. Z. Jhanjhi, and A. Abdullah, "Load balancing techniques in cloud computing environment: A review," Journal of King Saud University - Computer and Information Sciences, vol. 34, no. 7. King Saud bin Abdulaziz University, pp. 3910–3933, Jul. 01, 2022. doi: 10.1016/j.jksuci.2021.02.007.

[8] I. Odun-Ayo, O. Ajayi, and N. Omoregbe, "Cloud Service Level Agreements – Issues and Development," in 2017 International Conference on Next Generation Computing and Information Systems (ICNGCIS), Dec. 2017, pp. 1–6. doi: 10.1109/ICNGCIS.2017.18.

[9] A. Badshah, A. Ghani, S. Shamshirband, G. Aceto, and A. Pescapè, "Performance-based service-level agreement in cloud computing to optimise penalties and revenue," IET Commun., vol. 14, no. 7, pp. 1102–1112, Apr. 2020, doi: 10.1049/iet-com.2019.0855.

[10] D. Mekonnen, A. Megersa, R. K. Sharma, and D. P. Sharma, "Designing a Component-Based Throttled Load Balancing Algorithm for Cloud Data Centers," Math. Probl. Eng., vol. 2022, pp. 1–12, Oct. 2022, doi: 10.1155/2022/4640443.

[11] S. Singhal and A. Sharma, "An energy-aware migration framework using metaheuristic algorithm in cloud computing," Knowl. Inf. Syst., Oct. 2024, doi: 10.1007/s10115-024-02224-1.

[12] I. Z. Yakubu, Z. A. Musa, L. Muhammed, B. Ja'afaru, F. Shittu, and Z. I. Matinja, "Service Level Agreement Violation Preventive Task Scheduling for Quality of Service Delivery in Cloud Computing Environment," Procedia Comput. Sci., vol. 178, pp. 375–385, 2020, doi: 10.1016/j.procs.2020.11.039.

[13] G. H. Mengesha, P. F. Musa, S. Negash, R. Leuleged, and G. Munchus, "Assessment of readiness for launching cloud computing services: The case of <scp>Ethio-Telecom</scp> in Ethiopia," Electron. J. Inf. Syst. Dev. Ctries., vol. 88, no. 4, Jul. 2022, doi: 10.1002/isd2.12210.

[14] Z. Á. Mann, "Cloud simulators in the implementation and evaluation of virtual machine placement algorithms," Softw. Pract. Exp., vol. 48, no. 7, pp. 1368–1389, Jul. 2018, doi: 10.1002/spe.2579.

[15] I. Hussain, A. Khalique, F. Alvi, F. Siddiqui, and S. Zafar, "Reducing Energy Consumption in Cloud Data Centers via Predictive Virtual Machine Consolidation," in Conference Proceedings of Interdisciplinary Research for Sustainable Development, May 2023. [Online]. Available: https://www.researchgate.net/publication/381323219

[16] Namrata Bisht, "Virtualization in Cloud Computing and Types," GeeksforGeeks, Jul. 13, 2024. https://www.geeksforgeeks.org/virtualization-cloud-computing-types/ (accessed Oct. 07, 2024).

[17] N. Mansouri, R. Ghafari, and B. M. H. Zade, "Cloud computing simulators: A comprehensive review," Simul. Model. Pract. Theory, vol. 104, p. 102144, Nov. 2020, doi: 10.1016/j.simpat.2020.102144.

[18] P. S. Suryateja, "A Comparative Analysis of Cloud Simulators," Int. J. Mod. Educ. Comput. Sci., vol. 8, no. 4, pp. 64–71, Apr. 2016, doi: 10.5815/ijmecs.2016.04.08.

[19] A. Hussain, M. Aleem, M. A. Iqbal, and M. A. Islam, "Investigation of Cloud Scheduling Algorithms for Resource Utilization Using CloudSim," Comput. Informatics, vol. 38, no. 3, pp. 525–554, 2019, doi: 10.31577/cai_2019_3_525.

[20] Rajkumar Buyya, Remo Andreoli, Jie Zhao, and Tharindu Bandara, "The Cloud Computing and Distributed Systems (CLOUDS) Laboratory, University of Melbourne," The University of Melbourne, Australia. http://www.cloudbus.org/cloudsim (accessed Oct. 12, 2024).

[21] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. F. De Rose, and R. Buyya, "CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," Softw. Pract. Exp., vol. 41, no. 1, pp. 23–50, Jan. 2011, doi: 10.1002/spe.995.

[22] B. Ahmad, S. Mcclean, D. Charles, and G. Parr, "Analysis of Energy Saving Technique in CloudSim Using Gaming Workload," in The Ninth International Conference on Cloud Computing, GRIDs, and Virtualization, Feb. 2018.

[23] D.-M. Bui, Y. Yoon, E.-N. Huh, S. Jun, and S. Lee, "Energy efficiency for cloud computing system based on predictive optimization," J. Parallel Distrib. Comput., vol. 102, pp. 103–114, Apr. 2017, doi: 10.1016/j.jpdc.2016.11.011.

[24] A. Beloglazov and R. Buyya, "Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in Cloud data centers," Concurr. Comput. Pract. Exp., vol. 24, no. 13, pp. 1397–1420, Sep. 2012, doi: 10.1002/cpe.1867.

[25] J. Shuja et al., "Greening emerging IT technologies: techniques and practices," J. Internet Serv. Appl., vol. 8, no. 1, p. 9, Dec. 2017, doi: 10.1186/s13174-017-0060-5.

[26] B. V and V. joe S, "GREEN CLOUD COMPUTING SOLUTION FOR OPERATIONAL COST EFFICIENCY AND ENVIRONMENTAL IMPACT REDUCTION," J. ISMAC, vol. 01, no. 02, pp. 40–48, Jun. 2019, doi: 10.36548/jismac.2019.2.005.

[27] S. Rostami, A. Broumandnia, and A. Khademzadeh, "An energy-efficient task scheduling method for heterogeneous cloud computing systems using capuchin search and inverted ant colony optimization algorithm," J. Supercomput., vol. 80, no. 6, pp. 7812–7848, Apr. 2024, doi: 10.1007/s11227-023-05725-y.

[28] J. Wang, H. Gu, J. Yu, Y. Song, X. He, and Y. Song, "Research on virtual machine consolidation strategy based on combined prediction and energy-aware in cloud computing platform," J. Cloud Comput., vol. 11, no. 1, p. 50, Sep. 2022, doi: 10.1186/s13677-022-00309-2.

[29] Y. Chen, Z. Zhang, Y. Deng, G. Min, and L. Cui, "A Combined Trend Virtual Machine Consolidation Strategy for Cloud Data Centers," IEEE Trans. Comput., pp. 1–14, 2024, doi: 10.1109/TC.2024.3416734.

[30] A. H. A. AL-Jumaili, Y. I. Al Mashhadany, R. Sulaiman, and Z. A. A. Alyasseri, "A Conceptual and Systematics for Intelligent Power Management System-Based Cloud Computing: Prospects, and Challenges," Appl. Sci., vol. 11, no. 21, p. 9820, Oct. 2021, doi: 10.3390/app11219820.

[31] Anand Nayyar, "The Best Open Source Cloud Computing Simulators," opensource for you, Nov. 07, 2016. https://www.opensourceforu.com/2016/11/best-open-source-cloud-computing-simulators/ (accessed Oct. 18, 2024).

[32] M. P. R. Mohemmed Sha M, "Cloud-based Healthcare data management Framework," KSII Trans. Internet Inf. Syst., vol. 14, no. 3, Mar. 2020, doi: 10.3837/tiis.2020.03.006.

[33] MoH (Health of Ministry), "Directory of digital health solutions in Ethiopia," Iternational Institute for Primary Healtcare-Ethiopia (IPHCE), Addis Ababa, 2022. Accessed: Oct. 25, 2024. [Online]. Available: http://repository.iphce.org/handle/123456789/2633

[34] P. T. Aji and L. Ramadani, "Patients' Acceptance of Telemedicine Technology: The Influence of User Behavior and Socio-Cultural Dimensions," J. Inf. Syst. Eng. Bus. Intell., vol. 10, no. 1, pp. 81–93, Feb. 2024, doi: 10.20473/jisebi.10.1.81-93.

[35] K. Park and V. S. Pai, "CoMon: a mostly-scalable monitoring system for PlanetLab," ACM SIGOPS Oper. Syst. Rev., vol. 40, no. 1, pp. 65–74, Jan. 2006, doi: 10.1145/1113361.1113374.

[36] Ministry of health of Ethiopia, "Ethiopia Health Sector Transformation Plan (HSTP-II)," 2021. Accessed: Oct. 24, 2023. [Online]. Available: https://www.globalfinancingfacility.org/sites/default/files/Ethiopia-HSTP-II.pdf

[37] A. Choudhary, M. C. Govil, G. Singh, L. K. Awasthi, E. S. Pilli, and D. Kapil, "A critical survey of live virtual machine migration techniques," J. Cloud Comput., vol. 6, no. 1, p. 23, Dec. 2017, doi: 10.1186/s13677-017-0092-1.

[38] T. He and R. Buyya, "A Taxonomy of Live Migration Management in Cloud Computing," ACM Comput. Surv., vol. 56, no. 3, pp. 1–33, Mar. 2024, doi: 10.1145/3615353.

[39] P. Verma, A. K. Maurya, and R. S. Yadav, "A survey on energy-efficient workflow scheduling algorithms in cloud computing," Softw. Pract. Exp., vol. 54, no. 5, pp. 637–682, May 2024, doi: 10.1002/spe.3292.

[40] A. Hussain, M. Aleem, M. A. Islam, and M. A. Iqbal, "A Rigorous Evaluation of State-of-the-Art Scheduling Algorithms for Cloud Computing," IEEE Access, vol. 6, pp. 75033–75047, 2018, doi: 10.1109/ACCESS.2018.2884480.

Authors' Profiles

Aschalew Arega Ademe earned his M.Sc. in Information Systems from the Kobe Institute of Computing, Graduate School of Information Technology, KOBE, Japan, in 2017. He completed a B.Sc. in Information Systems from Addis Ababa University, Ethiopia, in 2012. He also holds a diploma in Computer Applications from Alpha College, Hawassa, Ethiopia, in 2008. Currently, he is pursuing a Ph.D. in Computing and Information Technology at the Faculty of Computing and Software Engineering, Arba Minch University, Ethiopia. His dedication to the academic community has been recognized with four Reviewers' certificates from prestigious Scopus-indexed journals, awarded in 2023 and 2024. His prior experience includes serving as an Assistant Lecturer from 2012 to 2015 and a lecturer from 2018 to 2020 at Madda Walabu University, where he was also the Head of the Information Systems Department during two separate terms—2013-2015 and 2019-2020. His research interests encompass cloud computing, information technology, computer applications technology, and information security.

Professor Durga Prasad Sharma is a distinguished Indian computer scientist and academician. He received a Ph.D. from the University of Rajasthan, a Master of Computer Application (MCA), a Master of Technology (M.Tech.) in Information Technology, and a B.Sc. degree from Government College Dholpur, India. In 2000, Sharma became an Assistant Professor of Computer Science & Informatics and was later promoted to Associate Professor and Full Professor. Among his achievements, in 2005, he developed a job search engine for people with disabilities. In 2019, based on Sharma's recommendation, the Government of Rajasthan started establishing a Technical University in Jaipur, Rajasthan, to provide technology-enabled training, education, and rehabilitation to people with disabilities. Currently, he is serving Arba Minch University as an Expat Research Adviser under an International assignment. Professor Sharma has authored or co-authored 27 books, published 161 research articles and 5 patents, received 62 Awards, and edited/reviewed 605 articles in several conferences, Journals, and editorial publications. For more detail, https://dpsharma.org

How to cite this paper: Aschalew Arega, Durga Prasad Sharma, "Evaluating Energy-efficiency and Performance of Cloud-based Healthcare Systems Using Power-aware Algorithms: An Experimental Simulation Approach for Public Hospitals", International Journal of Information Technology and Computer Science(IJITCS), Vol.17, No.3, pp.72-96, 2025. DOI:10.5815/ijitcs.2025.03.06