ISSN 2278-3091

Volume 8, No.1.4, 2019

International Journal of Advanced Trends in Computer Science and Engineering

Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse5281.42019.pdf

https://doi.org/10.30534/ijatcse/2019/5281.42019

The Optimization of Resources Allocation to Minimize Energy Consumption Within Data Center in Cloud

Fahrul Hakim Ayob[1], **Punaya Jayatis**[2], **Shamala Subramaniam**[3]
UPM, Malaysia,
[1]fahrulhakim@gmail.com
[2]zenith72@gmail.com
[3]drshamala@gmail.com

ABSTRACT

Cloud computing is a ubiquitous technology in computer science which provides some advantages to current users of modern computing. The advantages of these services are extremely significant, but the data centers that offer this kind of service are suffering from over power consumption that impose serious threats to the environment due to the emission of carbon dioxide to the atmosphere. Because of some disadvantages in Dynamic Voltage Frequency Scheduling (DVFS), this paper proposes a cloud scheduling algorithm adapted from Dynamic Resource Allocation Method (DRA) to be developed with the goal of dynamically virtual machines (VMs) migration when resource utilization reaches above or below threshold readings in order to optimize the power consumption towards energy saving in cloud data center. The advantage of having VMs migration mechanism is to have the capability to consolidate VMs in order to optimize the capacity of resources utilization. Thus, at the same time it has the capability to switch off some under-loaded hosts which are below the threshold reading. So in this research, our proposed algorithm is going to optimize the total running machine by switching off the hosts with below 30% of CPU power consumption and migrate all the VMs run under it to other hosts. But, for over-utilized hosts with more than 90% of computing resources consumption, the migration of some VMs run under it will be carried out. When comparing with DVFS in a set of experiment, the proposed algorithm is better in terms of lower power consumption.

Key Words: cloud, DRA, DVFS, virtual machines, power consumption, VM migration

1. INTRODUCTION

Nowadays, cloud computing has become a new buzzword in offering file hosting and data sharing services through data centers [1]. Cloud computing also provides various capabilities of services and solutions to users which incredibly reduce the operation cost as compared to the traditional way [2]. With the existence of commercial cloud providers such as Google, Amazon, Alibaba, and Apple *icloud*, it allows even a small start-up business organization or non-commercial one to gain access to a large powerful computing resource without the need to set up their own data center at first, which lead to the higher costs of operation and maintenance [3][4]. For examples the organizations' staff payroll cost may increase a lot due to the higher number of manpower recruitments as well as the cost of electric bills due to the higher power consumption which coming from those powerful computing machines that constantly burdening the entire organization. Apart from that, excessive power consumption might also hazardous to the environment because it will excessively release the carbon footprint to the atmosphere and thus increases the greenhouse effect that lead to the global warming [5]-[7].

By hiring cloud infrastructure services, these responsibility and burden could be delegated to cloud providers with specialization in taking care of everything such as data hosting, data sharing, file backup-storage as well as productivity software handling service[8]-[10]. Due to the increasing demands of cloud services, the high usage level of computing resources to satisfy the needs of applications that runs on cloud platform are badly required which leads to the increasing energy consumption that becomes a grave problem in cloud computing environment [11][12]. Therefore, there is a crucial necessity to minimize the energy or power consumption with energy-efficient resource management technique [12][13].

Data centers in cloud environment are not only facing with excessive power consumption but also suffering from uneven distribution of workload among the VM hosts that could result to underutilization of the resources that worsen the power consumption problem [14]. According to some studies, in 2001, the worldwide data centers' power consumption had reached 6358 hundred million kWh with 568 hundred million kWh in China alone. And in 2012, the energy consumption increased to 7202 hundred million kWh with 664 hundred million kWh in China alone [15]. While in Japan, their report suggests that its data centers would consume the entire electricity supply by the year 2030 if the growth continues at today's rate [16]. In addition to energy costs, cooling costs for data centers are also continue to soar because the ever-increasing of power consumption by VM servers that lead to the increase of heat dissipation [17]. So, this research proposes a scheduler based on DRA to be developed to manage the load balancing among VMs host in order to perform its migration process dynamically across the data center network towards minimal energy consumption.

1.1 Problem Statements

There are quite a number of methods and techniques that have been suggested to reduce power consumption in data centers and to efficiently utilize available resources using scheduler algorithms in cloud environment. For instance, [18] in their recent paper highlighted a few scheduling algorithm on power usage efficiency. One of the popular scheduling algorithms recommended in [18] is Dynamic Voltage Frequency Scheduling (DVFS). This scheduling algorithm uses a technique that dynamically change the voltage and frequency of host based on its load. However there are few issues arise such as:

- DVFS does not have a dynamic migration mechanism. Idle machines and under-utilize VM hosts are still in unnecessary power-ON state which excessively consume the electricity.
- DVFS dynamically changes the voltage and frequency merely based on CPUs load. What if the actual load was caused by RAM or Bandwidth utilization? Does it will affect the overall result of power usage?

So, the main objective of this research is to develop a scheduling algorithm based on DRA which could minimize the power consumption in a cloud data center by optimizing the number of host used. This could be done by implementing a dynamic migration process to increase the efficiency of resource allocation among the VM hosts in order to avoid the waste of energy across the data center network. The output of this study is subject to the simulation of cloud environment that leads to the conservation of the energy consumption in cloud data center. The results of this study will be compared with DVFS.

This paper is organized as follows. Section 2.0 discusses on related works which mainly focus on scheduling methods in cloud computing including DRA and DVFS which have been the base of this work. Section 3.0 explains the methodology of our proposed algorithm. All the criteria to be evaluated in this experiment with its settings are discussed in Section 4.0. Results and Discussions are explained in Section 5.0. The whole research ideas with its findings and contributions are concluded in Section 6.0.

2. RELATED WORKS

There are multiple methods suggested and has been implemented to optimize cost and resources utilization in cloud environment. One of the methods is to build dynamic resource scheduling algorithm which is able to monitor resource usage and perform necessary steps to ensure resources are evenly distributed. The methods suggested improved the efficiency and helped reduced the overall cost of managing resources (minimize idle time and maximize resources accordingly) and reduced overall power consumption of data centers. The main purpose of scheduling algorithm is to ensure a fair allocation of task over available resources and an evenly distribution of workload to all resources available. To solve the problem of scheduling of tasks over available resources, a large number of algorithms have been introduced. However many of those algorithms are not applicable in large scale distributed systems such as cloud environment or grid environment due to high communication cost [19].

Dynamic Resource Allocation (DRA) method and Energy Saving method have been introduced in [20]. Their main research problem is to find ways to save and use the idled resources on low loading Virtual Machines (VMs) effectively and to save energy consumption on servers. The proposed system is built based on the software Open Stack, an infrastructure platform for the cloud. From the research problem, researchers in the paper introduced two algorithms to improve traditional DRA and energy saving methods. The proposed DRA algorithm, manage to reduce idled resource on the VMs while the improved energy saving algorithm manage to reduce the energy consumption of the entire cloud cluster [20]. The researchers claims to achieve goal by using the DRA method to reduce the idled resources on the VM and using energy saving method to save the energy consumption of servers. With the DRA method introduced in the paper, it is not only can reduce the waste of the idled resources on VMs, but also can increase allocation of resources to any VM that needs extra resources to meet the demand. The energy saving method worked mainly through shutting idle physical machines to save energy consumption with the technique call “live migration”. The “Live migration” refers to the process of moving a running VM or application between different physical machines without disconnecting the client or application [20]. Memory, storage, and network connectivity of the VM are transferred from the original guest machine to the destination. There is no downtime needed to perform “live migration” and this is one of the most added advantage of having cloud environment implemented in any organizations. It almost guarantees to provide 99.9% of uptime which allows business to continue as usual during this process of “live migration” [20].

An enhanced algorithm based upon the existing algorithm with practical implementation to carry out several tests and execution plans has been proposed in [21]. Based upon the tests, they introduced a technique which is to allocate each VM with a part of the total requested power according to its power factor. The research evaluates the performance of the enhanced algorithm and conducts a comparative study over the other algorithms based on Particle Swarm Optimization

(PSO), Genetic Algorithm (GA), and First Come First Serve (FCFS) with respect to the execution time and the price of execution using Amazon Elastic Compute Cloud (Amazon EC2), and Google pricing models. Based upon their experimental results, it proves the efficiency of their enhanced algorithm by minimizing the make-span or the time taken for a process to be completed [21].

The increase functions of cloud data centers have proved them to become the main energy consumer in today's modern computing. Research in seeking a various way to reduce energy consumption and to increase its usage efficiency has become a new trend. This trend leads to the improvement and the creation of new algorithms that will improve the current power consumption. Energy-aware tool is one of the solutions that could be used which allows the minimization of the power consumption in data centers. [22] address this issue by analyzing three common energy-aware tools that focuses on hosts virtualization. The first solution, calls ON/OFF method, which turns off the underutilized hosts and switches them on again if necessary (according to the threshold readings). All processes running on an underutilized host are moved to other hosts, and then the underutilized host is switched off. Conversely, when there are hosts are over-utilized, some switched off hosts will be switched on again. This process has only one goal, which is to reduce and to optimize the number of switched-on hosts at a given time [22].

In contrast to traditional data centers, cloud computing model allows greater energy efficiency to be achieved by using techniques such as virtualization and consolidation workload, providing better control of computing resources in relation to their demand, better security management and of course avoiding the waste of resources from just being idled for most of the time [23]. The energy costs of data centers are rapidly increasing each year. It is estimated that the cost would be doubled by the year 2020. However with the numerous efforts that have been taken, they help power usage to be maintained in much slower pace compared to the previous decades. Based upon the report as in [24], energy used by data centers only grew 4% between 2010 and 2014 which contrast to the 90% growth between 2000 and 2005, and 24% growth between 2005 and 2010. The report predicts that data center energy use will only grow another 4% between 2014 and 2020 [24]. For a company like Google, it has initially adopted energy efficiency software in its data centers which intelligent enough to automatically switch off some VM hosts when lower-power consumption is detected.

One of the scheduler algorithms which use CPU frequency to perform calculation and decision is Dynamic Voltage and Frequency Scaling (DVFS) [18]. DVFS, being widely applied in modern processors, is recognized as an effective technique for achieving the balance between system performance and energy saving [25]. With DVFS, the processor could dynamically alter the working frequency, which leads to different power consumption. Nevertheless, "lower frequency fewer energy consumption" is not always true because low frequency increases the task execution time as well as energy depends both on power and execution time [25].

3. THE PROPOSED ALGORITHM

This research is focused on how the resource allocation could be efficiently utilized towards power saving in a cloud environment. The simulation work will be done by using CloudSim as this platform offers framework for modeling and simulation of cloud computing infrastructures and services. CloudSim provides provisioning capabilities for tasks execution (cloudlets) that run on VMs that in turn could be run on servers that form a data center. CloudSim is fully developed using JAVA platform. For measuring power consumption, it is calculated using the existing policy which has been implemented in CloudSim. This will provide an accurate result since our proposed method will be calculated using the same metrics as the existing one. There are three phases in this study:

First phase: This phase would require extensive amount of time and experiments as it involves simulating the existing work from the previous research as in [18] and developing the proposed one.

- To identify the characteristics and attributes used
- To identify the jobs that need to be executed during simulation and the scheduler algorithm used in the previous research.

Second phase:

- Simulate the presented algorithm with the same set of characteristics and attributes.
- Simulate the proposed algorithm with the same set of characteristics and attributes.

Third phase:

This is crucial part of the research work as it will determine the results which concludes the outcome of the experiment and the observation that have been made throughout the work.

- Tabulate and compare the results obtained.

In an attempt to reduce the energy consumption, overloaded and under-loaded hosts must first be identified in data center. Once identified, the next step is to decide which VMs are to be migrated and to which host. Optimum selection of the right VMs and replacing them on the appropriate hosts could technically reduce the energy consumption. Our work focuses on the migration of VMs which depends on the characteristics of the host which based on these three (3) conditions as follows:

- high workload host (overloaded host with and over 90% workload)
- non-high workload host (host with 31% - 89% workload)
- low workload host (under-loaded host with 0% - 29% workload)

Based on these three conditions, the decision is made to dynamically perform migration of virtual machines (VMs). The aim of this decision is to minimize the power consumption by minimizing total running machine by switching off some of the under-utilized or under-loaded host.

Figure 1 is the algorithm to determine host utilization in processing time interval. Host utilization is calculated based on RAM utilization, Million instruction per seconds (MIPS) utilization and Bandwidth utilization (BW) which are all derived from Equations (1) to (3) respectively as follows.

RAM utilization (*ramUtilization*) percentage ratio is derived by dividing the amount of RAM in every virtual machine (RAM_{vmi}) with the total RAM allocated in the physical machine (RAM_{pm}) as shown in Equation (1) as follows.

$$ramUtilization = \sum_{i=0}^{n} RAM_{vmi} \,/\, RAM_{pm} \qquad (1)$$

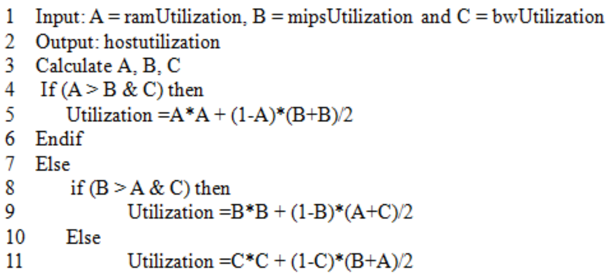
The percentage of million instructions per second utilization (*mipsUtilization*) is calculated in Equation (2) as follows. In this equation, MIPS of each virtual machine ($MIPS_{vmi}$) is divided with the total allocated MIPS of the physical machine ($MIPS_{pm}$).

$$mipsUtilization = \sum_{i=0}^{n} MIPS_{vmi} \,/\, MIPS_{pm} \qquad (2)$$

Equation (3) on the other hand, is to find the percentage ratio of the current bandwidth utilization (*bwUtilization*). This can be done by dividing the bandwidth of each virtual machine (BW_{vmi}) with the total allocated bandwidth of the physical machine (BW_{pm}). The equation is as follows.

$$bwUtilization = \sum_{i=0}^{n} BW_{vmi} \,/\, BW_{pm} \qquad (3)$$

Figure 1 shows the proposed algorithm to calculate the host utilization based on the utilization of RAM, MIPS and bandwidth (BW) as described in Equation (1)-(3).



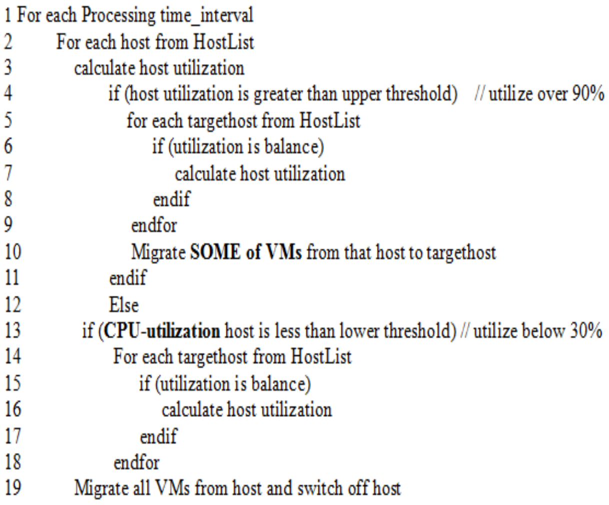
```
1  Input: A = ramUtilization, B = mipsUtilization and C = bwUtilization
2  Output: hostutilization
3  Calculate A, B, C
4   If (A > B & C) then
5       Utilization =A*A + (1-A)*(B+B)/2
6  Endif
7  Else
8       if (B > A & C) then
9            Utilization =B*B + (1-B)*(A+C)/2
10      Else
11           Utilization =C*C + (1-C)*(B+A)/2
```

Figure 1: Calculation of Host Utilization

Algorithm

For example, the values (in percentage) A, B and C are given as 70%, 50% and 50% respectively. From the *if-else* statement algorithm in Figure 1 above, the highest ratio will be selected as it is bigger than that of B and C. Then utilization statement in line 5 (Figure 1) is executed which yields the parameter value of 0.64. This parameter value then will be used as the benchmark to find and to perform the virtual machine migration based on the dynamic resource allocation method.

Algorithm in Figure 2 is then to perform the VM migrations from the host that exceeds the upper threshold size (utilize over 90%). It will select the VMs to be migrated based upon the highest ratio determined from the previous algorithm in Figure 1.



```
1 For each Processing time_interval
2     For each host from HostList
3       calculate host utilization
4           if (host utilization is greater than upper threshold)   // utilize over 90%
5             for each targethost from HostList
6                 if (utilization is balance)
7                     calculate host utilization
8                 endif
9             endfor
10            Migrate SOME of VMs from that host to targethost
11          endif
12          Else
13       if (CPU-utilization host is less than lower threshold) // utilize below 30%
14            For each targethost from HostList
15                if (utilization is balance)
16                    calculate host utilization
17                endif
18            endfor
19      Migrate all VMs from host and switch off host
```

Figure 2: VMs Migration Algorithm

If the hosts utilization exceeded the upper threshold level (over 90%), it is considered as over utilized and the VMs will then be migrated to a host which is not over utilized (below 90%). The hosts might be overloaded with over consumption of resources such as memory, MIPS or bandwidth. If this occurred, the algorithm will then search for the target host with lower utilization for the migration process. On the other hand, if the host with CPU utilization were detected as under-utilized, (below the threshold level of 30%) then all the VMs from the host will be migrated and the host will be switched off. In this research we only focus on under utilized host with lower threshold CPU consumption to be migrated in order to limit the number of host being switched off.

Setting at 90% as an upper threshold indicator (line 4 in Figure 2) has been considered as the universal benchmark value which indicates that the host is overloaded. Setting at 100% on the other hand, is considered as a very high risk since it may crash and damage the hosts and VMs that are running under it.

Figure 3 explains the overall process of the proposed scheduler execution which is based on DRA. It starts with the execution of the proposed scheduler Java file which will invoke all the random runners to generate and assign variables and parameters to the cloudlets (tasks), VMs, host and the data centre. Once the data centre has been created with host and VMs, it will then set two policies which are referred as the VM migration policy and VM selection policy. These policies will perform the selection and the migration processes of the VMs based on the threshold size. The cloudlets will assign 25 I/O tasks and 25 CPU tasks to each VM in the data centre. The algorithm (as in Figure 1) will perform the calculation to determine the utilization of the host during the processing interval time. The second algorithm (as in Figure 2), will decide which host is overloaded and under-loaded. So, based upon this criterion, VM migration policy and VM selection policy will be executed thus dynamically trigger the VMs migration process and powers down the under-utilized hosts.

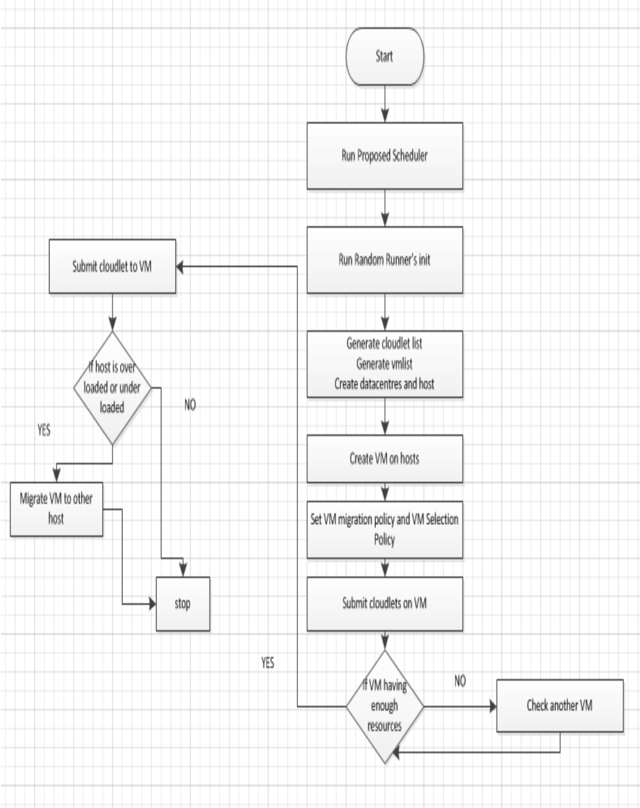


Figure 3: Proposed Scheduler Flowchart

4. EXPERIMENTAL DESIGN & EVALUATION

The experiment performed in this study was carried out by using the CloudSim simulator. CloudSim is an extensible simulation toolkit that enables modeling and simulation of Cloud Computing systems and application provisioning environments. The CloudSim toolkit supports the Clouds components such as data centers, virtual machines (VMs), and resource provisioning policies. On the other hand, the Cloudsim could implement generic application provisioning techniques that can be extended easily with limited efforts.

Figure 4 shows the cloudsim framework. There are three main components which consists of Cloudlets, Broker and Cloud Information Services (CIS). CIS is an integral part of Cloudsim which creates and allocates resources to data centres upon receiving requests from broker. It also ensures that all entities in the data centres are registered with CIS. While Broker focuses on the communication between CIS and Cloudlets. It will submit resources to CIS upon receiving requests from Cloudlets on the number of tasks. Cloudlets on the other hand, allow the creation of processes. Each process, no matter whether it is either a CPU or an I/O, will need to be initialized as a cloudlet. Cloudsim uses three kinds of policy which are referred as VM Allocation Policy used by data centre, VM Scheduling Policy utilized by hosts and Cloudlet Schedule Policy implemented by the VMs.

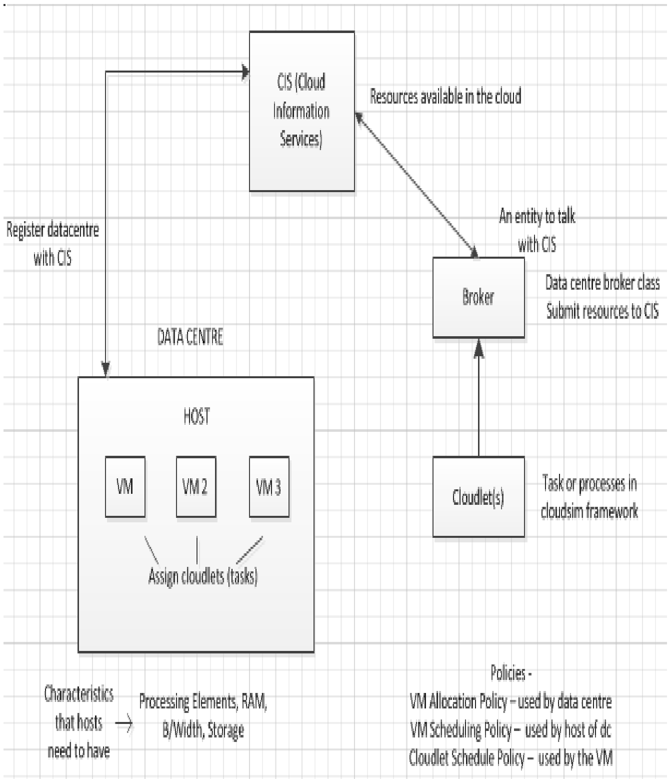

CIS (Cloud Information Services)
Resources available in the cloud
Register datacentre with CIS
An entity to talk with CIS
Broker
Data centre broker class
Submit resources to CIS
DATA CENTRE
HOST
VM
VM 2
VM 3
Assign cloudlets (tasks)
Cloudlet(s)
Task or processes in cloudsim framework
Characteristics that hosts need to have →
Processing Elements, RAM, B/Width, Storage
Policies -
VM Allocation Policy – used by data centre
VM Scheduling Policy – used by host of dc
Cloudlet Schedule Policy – used by the VM

Figure 4: CloudSim Framework

For the implementation, some features of the environment to be assessed were set. Thus, each simulation considers:

- a single data center.
- 2 scheduler methods: DVFS as implemented in [18] and our proposed scheduling algorithm based on DRA method.
- 50VMs
- 50 tasks (one to be executed on each VM), 25 CPU-bound and 25 I/O-bound.
- The scheduler time interval (round) of 60th sec min,225th sec , 445th sec, 900th sec, 1800th sec and 3600th sec.

Table 1 and Table 2 give the details on the specifications of host and VM respectively which appear in the simulation.

Table 1: Host Specifications

Processor	HP ProLiant ML110 G4 Xeon 3040 1860 MHz, 2 cores
RAM	80 GB
Bandwidth	2 Gbs
Storage	500 GB

Table 2: Virtual Machine Specifications

Processor	HP ProLiant ML110 G4 Xeon 3040 1860 MHz, 2 cores
RAM	4 GB
Bandwidth	100 Mbs
Storage	2.5 GB

To evaluate the performance of our research, the power consumption is used as the performance metrics. The simulation experiment is carried out on two entities which are the number of hosts and processing elements (PE). Each PEs is attached to the hosts. In a set of experiment, there are six (6) different time intervals to be scheduled and observed. The experiment is carried out to examine the performance of two algorithms (DVFS and our proposed algorithm) and to identify the best between these two in terms of power consumption. Both algorithms are tested with similar set of parameters. Each time interval consists of four segments parameters which are the number of hosts used to capture the power consumption readings. The purpose of the execution of experiment in repetitive manner is to acquire a better interpretation of results, thus implies this comparative study is in high precision. Our simulation experiment is conducted by using CloudSim framework, which did the job tremendously quite well.

5. RESULTS AND DISCUSSION

Inevitably power consumption has become a major concern in the modern era of cloud computing. Another major concern is the speed and adaptability of allocating resources when requested during the crucial time (when it is needed the most). Dynamic resources scheduler is the preferred solution adopted by most developers when designing an algorithm that could satisfy all the current requirement of cloud computing. Thus, understanding the different ways of resource allocation would benefit in terms of developing best suit model for current cloud computing environment. Therefore, this research highlights the performance comparison between DVFS (from the previous research) with our proposed scheduler adapted from DRA on some parameters and time interval.

Table 3 describes the interpretation of terms used in the simulation experiment. For better understanding, the graphs were plot according to each time interval scheduled. Total, there are six graphs which depict the overall results of each time interval.

Table 3: Interpretation of the terms used in the simulation experiment

Terms Used	**Interpretation (Meaning)**
Cloudlet	50 number of tasks (25 I/O bound and 25 CPU bound)
PE	Processing Element which refers to CPU assigned
Host	number of physical host
DVFS	Dynamic Voltge Frequency Scheduling - an energy conscious cloud computing scheduling from previous research which will be compared with our proposed scheduling algorithm.
Proposed Method	our proposed scheduling algorithm in this research which adapted from DRA method previously
60th Sec, 225th Sec, 445thSec, 900th Sec, 1800thSec , 3600thSec	Sequence of time intervals during the experiments which are going to be observed. Total, there are six time intervals considered to be observed in this experiment.

The rest of this section illustrates the acquired results in this research.

5.1 Time Interval at 60th Seconds Observation

Table 4 below is the result of time interval at 60th seconds. Graph in Figure 5 is based on Table 4.

Table 4: Results of time interval at 60th Seconds

PE	**Host**	**DVFS**	**Proposed Method**
		Energy Consumption	**Energy Consumption**
10	5	18.99	6.39
5	10	19.29	7.31
3	20	19.7	9.31
2	40	20.24	16.7

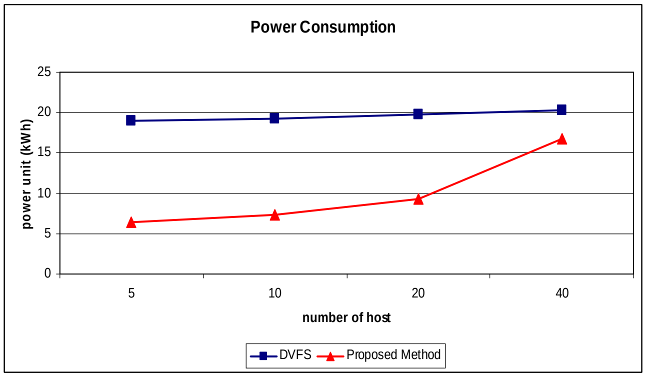


Figure 5: Graph Comparison of time Interval at 60th sec

According to Table 4 and Figure 5, our proposed algorithm consumes below half of the DVFS power consumption in the first three experiment segments with 5, 10 and 20 number of hosts. Only in the last experiment segment with 40 number of host, our proposed method utilizes more than half energy consumed by DVFS. Overall, our proposed method demonstrates significantly lower energy usage compared to DVFS.

5.2 Time Interval at 225th Seconds Observation

Table 5 below is the result of time interval at 225th seconds. Graph in Figure 6 is based on Table 5.

Table 5: Results of time interval at 225th Seconds

PE	Host	DVFS	Proposed Method
		Energy Consumption	Energy Consumption
10	5	19.1	6.51
5	10	19.4	7.48
3	20	19.81	9.27
2	40	20.35	16.52

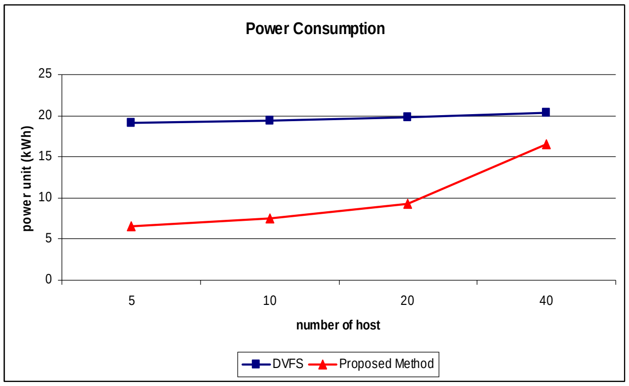


Figure 6: Graph Comparison of time Interval at 225th sec

According to the results as illustrates in Figure 6, our proposed method is still better in terms of minimizing the power consumption compared to DVFS. Apparently based on the graph in Figure 6, our proposed algorithm still could manage to maintain below half of DVFS power consumption in the first three segments with number of hosts 5, 10 and 20. Only for the last experiment segment with 40 number of hosts, our proposed method consumes higher energy than before (Table 5). However the reading is still lower than that of the DVFS which indicates that our proposed method is a way better off than DVFS in terms of energy saving.

5.3 Time Interval at 445th Seconds Observation

Table 6 below is the result of time interval at 445th seconds. Graph in Figure 7 is based on Table 6. Based on Table 6 and Figure 7, during the time interval at 445th seconds, our proposed method still consumes lower energy usage compared to DVFS in all the 4 segments of the experiments. Amazingly, our proposed algorithm also indicates that there is no change in terms of power consumption (refer Table 6) as compared to the recent experiment in Table 5.

Table 6: Results of time interval at 445th Seconds

PE	Host	DVFS	Proposed Method
		Energy Consumption	Energy Consumption
10	5	19.09	6.51
5	10	19.39	7.48
3	20	19.8	9.27
2	40	20.34	16.52

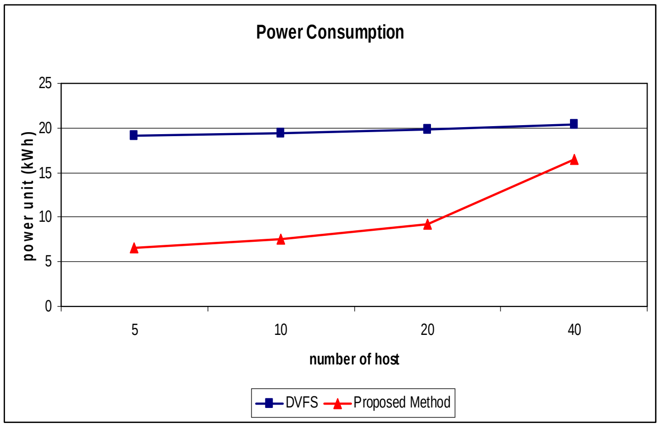


Figure 7: Graph Comparison of time Interval at 445th sec

5.4 Time Interval at 900th Seconds Observation

Table 7 below is the result of time interval at 900th seconds. Graph in Figure 8 is based on Table 7.

Table 7: Results of time interval at 900th Seconds

PE	Host	DVFS	Proposed Method
		Energy Consumption	Energy Consumption
10	5	19.39	6.85
5	10	19.68	8.11
3	20	20.1	9.91
2	40	20.63	17.61

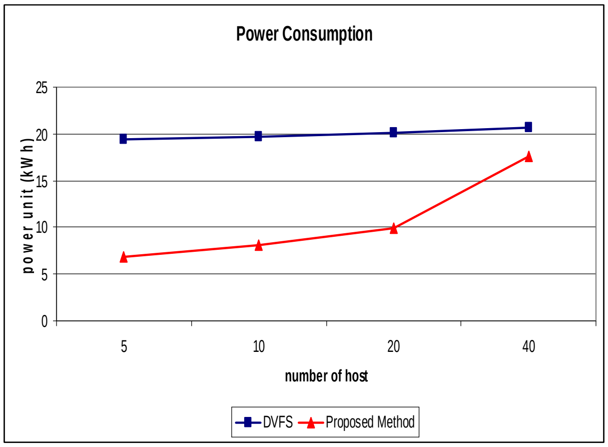


Figure 8: Graph Comparison of time Interval at 900th sec

During the time interval at 900th seconds, the readings obtains from our proposed method (Table 7) are almost similar (but with slightly higher) compared to the recent experiments which lead to the same pattern of graph as depicted in Figure 8. Overall, our proposed method is still relatively lower in power consumption compared to DVFS.

5.5 Time Interval at 1800th Seconds Observation

Table 8 below is the result of time interval at 1800th seconds. Graph in Figure 9 is based on Table 8.

During the time interval of 1800th seconds as depicted in Figure 9, our proposed method has demonstrated a lower power consumption in the first 3 segments (with number of host 5, 10 and 20). However only in the last segment of experiment with 40 hosts, surprisingly our proposed method and DVFS has recorded the same amount of power consumption which is 21.37kWh (refer Table 8).

Table 8: Results of time interval at 1800th Seconds

PE	Host	DVFS	Proposed Method
		Energy Consumption	**Energy Consumption**
10	5	20.13	7.31
5	10	20.43	8.74
3	20	20.84	9.92
2	40	21.37	21.37

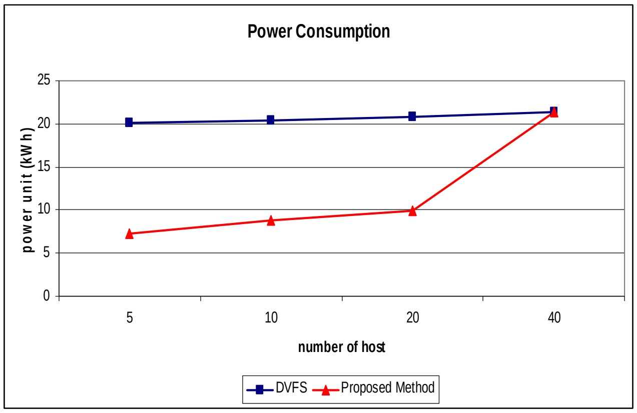


Figure 9: Graph Comparison of time Interval at 1800th sec

5.6 Time Interval at 3600th Seconds Observation

Table 9 below is the result of time interval during 3600th seconds. Graph in Figure 10 is based on Table 9. On the last set of experiment, during the time interval at 3600th seconds, our proposed algorithm still maintain better results compared to DVFS in terms of reducing energy consumption. In the first two experiment segments with number of host 5 and 10, our proposed method consumes below the half of DVFS power consumption. And the two remain experiment segments with number of host 20 and 40, our proposed algorithm still shows a better performance in terms of energy saving compared to DVFS with significantly lower power consumption.

Table 9: Results of time interval at 3600th Seconds

PE	Host	DVFS	Proposed Method
		Energy Consumption	**Energy Consumption**
10	5	20.23	7.87
5	10	20.53	9.89
3	20	20.94	11.11
2	40	20.97	18.33

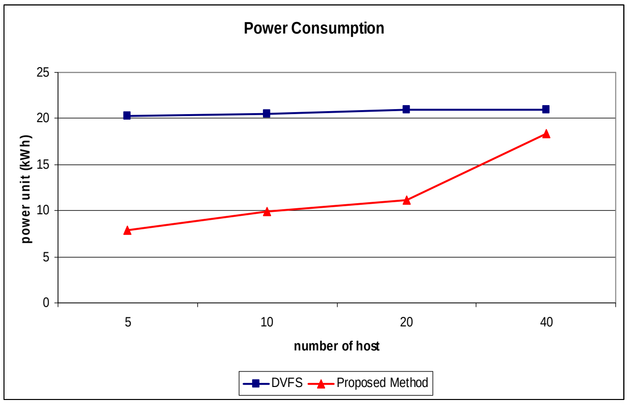


Figure 10: Graph Comparison of time Interval at 3600th sec

5.7 Interpretation of the Simulation Results

From the results that have been discussed in the previous sections, our proposed method which is based on Dynamic Resource Allocation Method (DRA) is able to achieve significantly lower power consumption as compared to DVFS scheduling algorithm. This is because, almost all the first three experiment segments in every set of simulation (with the number of host 5, 10 and 20) shows our proposed algorithms consumes below the half energy consumed by DVFS algorithm. This is indeed a remarkable finding which proves our proposed scheduler is not only capable in performing dynamic VMs migration, but at the same time it could significantly reduce the energy consumption. DVFS on the other hand, does not have a dynamic migration process mechanism which exposes their hosts with the unnecessary waste of energy. Technically, in this case, our proposed scheduler is able to achieve energy saving by consolidating hosts dynamically with migration process mechanism which indirectly encourages under-loaded host to be switched off.

6. CONCLUSION

This research highlights our proposed method in cloud scheduling adapted from DRA with the goal of the migration of VMs dynamically when resource utilization reaches above or below threshold readings in order to minimize the power consumption towards energy saving. The advantage of having a VMs migration mechanism is to have the capability of VMs consolidation in order to optimize the capacity of resources utilization. Thus, at the same time it has the capability to switch off the under-loaded hosts which are below the threshold reading. Our proposed scheduling algorithm takes into account the utilization of computing resources such as RAM, Bandwidth and CPU during processing time interval and chooses the highest parameter ratio. Instead of solely focuses on CPU, our proposed method also capable to determine other resources utilization level. With the over-utilization of resource identified among the hosts, the VMs migration process will be carried out. When comparing with DVFS in a set of experiment, we found that our proposed algorithm is better in terms of minimal power consumption.

REFERENCES

1. M. R. Mesbahi, A. M. Rahmani, & M. Hosseinzadeh. **Reliability and high availability in cloud computing environments: a reference roadmap**. *Human-centric Computing and Information Sciences*, 8(1), 20, 2018. https://doi.org/10.1186/s13673-018-0143-8
2. L. A. Barroso, U. Hölzle, & P. Ranganathan, . **The Datacenter as a Computer: Designing Warehouse-Scale Machines**. *Synthesis Lectures on Computer Architecture*, *13*(3), i-189, 2018. https://doi.org/10.2200/S00874ED3V01Y201809CAC046
3. M. Attaran, & J. Woods. **Cloud computing technology: improving small business performance using the Internet**. *Journal of Small Business & Entrepreneurship*, 1-25, 2018. https://doi.org/10.1080/08276331.2018.1466850
4. S. Liu, F. T. Chan, J. Yang, & B. Niu. **Understanding the effect of cloud computing on organizational agility: An empirical examination**. *International Journal of Information Management*, *43*, 98-111, 2018.
5. A. Jain, M. Mishra, S. K. Peddoju, & N. Jain. **Energy efficient computing-green cloud computing**. In IEEE *2013 International Conference on Energy Efficient Technologies for Sustainability*, April 2013 pp. 978-982.
6. W. Deng, F. Liu, H. Jin, B. Li, & D. Li. **Harnessing renewable energy in cloud datacenters: opportunities and challenges**. *IEEE Network*, *28*(1), 48-55, 2014. https://doi.org/10.1109/MNET.2014.6724106
7. L. N. Balasooriya, S. Wibowo, & M. Wells. **Green cloud computing and economics of the cloud: Moving towards sustainable future**. *GSTF Journal on Computing (JoC)*, *5*(1), 2016.
8. J. Singh, T. Pasquier, J. Bacon, H. Ko, & D. Eyers. **Twenty security considerations for cloud-supported Internet of Things**. *IEEE Internet of things Journal*, *3*(3), 269-284, 2016.
9. R. Beri, & V. Behal. **Cloud computing: A survey on cloud Computing**. *International Journal of Computer Applications*, *111*(16), 2015.
10. Z. Yan, X. Li, M. Wang, & A. V. Vasilakos. **Flexible data access control based on trust and reputation in cloud computing**. *IEEE transactions on cloud Computing*, *5*(3), 485-498, 2017.
11. Z. Zhou, J. Abawajy, M. Chowdhury, Z. Hu, K. Li, & F. Li. **Minimizing SLA violation and power consumption in Cloud data centers using adaptive energy-aware algorithms**. *Future Generation Computer Systems*, *86*, 836-850, 2018. https://doi.org/10.1016/j.future.2017.07.048
12. M. A. Haghighi, M. Maeen, & M. Haghparast. **An Energy-Efficient Dynamic Resource Management Approach Based on Clustering and Meta-Heuristic Algorithms in Cloud Computing IaaS Platforms**. *Wireless Personal Communications*, *104*(4), 1367-1391, 2019.
13. S. Kaushal, D. Gogia, & B. Kumar. **Recent Trends in Green Cloud Computing**. In *Proceedings of 2nd International Conference on Communication, Computing and Networking*, Springer, Singapore, 2019, pp. 947-956.
14. N. Thilagavathi, D. Dharani, R. Sasilekha, V. Suruliandi, & V. R. Uthariaraj. **Energy Efficient Load Balancing in Cloud Data Center Using Clustering Technique**. *International Journal of Intelligent Information Technologies (IJIIT)*, *15*(1), 84-100, 2019.
15. T. Yang, W. XIANG, F.E.N.G. Yingmin, P.E.N. Haibo, M. Xu, Y.O.U. Jinkuo, & H. Wang. **Method for solving the decomposition-coordination calculation based on**

Block Bordered Diagonal Form (BBDF) model using Data Center. *U.S. Patent Application No. 10/235,341*, 2019.

16. T. Bawden. **The Independent**. Available at: http://www.independent.co.uk/environment/global-warming-data-centres-to-consume-three-times-as-much-energy-in-next-decade-experts-warn-a6830086.html. 2017.
17. S.Akbar, S.U.R.Malik, S. U. Khan, R. Choo, A. Anjum, & N. Ahmad. **A Game-based Thermal-aware Resource Allocation Strategy for Data Centers**. *IEEE Transactions on Cloud Computing*, 2019. https://doi.org/10.1109/TCC.2019.2899310
18. F. Fernandes, D. Beserra, E.D. Moreno, B. Schulze, & R. C. G. Pinto. **A virtual machine scheduler based on CPU and I/O-bound features for energy-aware in high performance computing clouds**. *Computers & Electrical Engineering*, *56*, 854-870, 2016.
19. Mittal, S., & Katal, A. **An optimized task scheduling algorithm in cloud computing**. In *2016 IEEE 6th International Conference on Advanced Computing (IACC),* February, 2016, pp. 197-202. https://doi.org/10.1109/IACC.2016.45
20. C. C. Chen, P. L. Sun, C. T. Yang, J. C. Liu, S. T. Chen, & Z. Y. Wan, **Implementation of a Cloud Energy Saving System with Virtual Machine Dynamic Resource Allocation Method Based on OpenStack**. In *2015 IEEE Seventh International Symposium on Parallel Architectures, Algorithms and Programming (PAAP),* December 2015, pp. 190-196.
21. E., Ibrahim, N.A. El-Bahnasawy, & F. A. Omara. **Task scheduling algorithm in cloud computing environment based on cloud pricing models**. In *2016 IEEE World Symposium on Computer Applications & Research (WSCAR),* March 2016, pp. 65-71.
22. T. Guérout, T. Monteil, G. DaCosta, R. N. Calheiros, R. Buyya, & M. Alexandru. **Energy-aware simulation with DVFS**. *Simulation Modelling Practice and Theory*, *39*, 76-91, 2013. https://doi.org/10.1016/j.simpat.2013.04.007
23. K. Ye, D. Huang, X. Jiang, H. Chen, & S. Wu. **Virtual machine based energy-efficient data center architecture for cloud computing: a performance perspective**. In *Proceedings of the 2010 IEEE/ACM Int'l Conference on Green Computing and Communications & Int'l Conference on Cyber, Physical and Social Computing,* IEEE Computer Society, Dec 2010, pp. 171-178.
24. N. C. Horner, A. Shehabi, & I. L. Azevedo. **Known unknowns: Indirect energy effects of information and communication technology**. *Environmental Research Letters*, *11*(10), 103001, 2016. https://doi.org/10.1088/1748-9326/11/10/103001
25. S. Wang, Z. Qian, J. Yuan, & I. You. **A DVFS based energy-efficient tasks scheduling in a data center**. *IEEE Access*, *5*, 13090-13102, 2017. https://doi.org/10.1109/ACCESS.2017.2724598