MIT-WPU
॥ विश्वशान्तिर्ध्रुवं ध्रुवा ॥

Dr. Vishwanath Karad
MIT WORLD PEACE UNIVERSITY | PUNE
TECHNOLOGY, RESEARCH, SOCIAL INNOVATION & PARTNERSHIPS

Seminar Report

On

Survey of Energy Efficient Cloud Computing techniques

By

Aditya Vikramsinh Desai

PRN No. 1032170282

Under the guidance of

Prof. Rajendra G. Pawar

MIT-World Peace University (MIT-WPU)

Faculty of Engineering
School of Computer Engineering & Technology

*** 2019-2020 ***

MIT-World Peace University (MIT-WPU)

Faculty of Engineering
School of Computer Engineering & Technology

CERTIFICATE

This is to certify that Mr. Aditya Vikramsinh Desai of B.Tech., School of Computer Engineering & Technology, Trimester – IX, PRN. No. 1032170282 has successfully completed seminar on

Survey of Energy Efficient Cloud Computing techniques

To my satisfaction and submitted the same during the academic year 2019 - 2020 towards the partial fulfilment of degree of Bachelor of Technology in School of Computer Engineering & Technology under Dr. Vishwanath Karad MIT- World Peace University, Pune.

Prof. Rajendra G. Pawar

Seminar Guide

Prof. Dr. M.V. Bedekar

Head

School of Computer Engineering & Technology

List of Figures

Abbreviations

Sr. No	Abbreviation	Full Form
1	DC	Data Centers
2	VM	Virtual Machine
3	BP	Bin Packing
4	FFD	First Fit Decreasing
5	DVMC	Dynamic Virtual Machine Consolidation
6	OS	Operating System
7	CorHS	Correlation Host Selection
8	SLA	Service Level Agreement
9	DVFS	Dynamic Voltage and Frequency Scaling
10	MESF	Most Efficient Server First
11	QoS	Quality of Service
12	BF	Best - Fit
13	VNE	Virtual Network Embedding

ACKNOWLEDGEMENTS

I would like to express my gratitude towards my parents & faculty of MIT World Peace University, including Prof. Rajendra G. Pawar for guiding me in this topic, as well as Head of School, Dr Mangesh Bedekar and Seminar Coordinator, Prof. Jayshree Aher for their kind cooperation and encouragement which helped me in completing this seminar. My thanks and appreciations also go to my colleagues who helped me out with the best of their abilities to make this endeavour a success.

- Aditya Desai
1032170282
PC15, T. Y B.Tech CSE

INDEX

ABSTRACT

Cloud data centers consume large amounts of energy, generating a considerable amount of heat and CO2, even when most of the servers are idle and doing any processing. This is not an efficient way to manage the vast resources provided by the cloud. However, cloud services include not just computation servers, but also a broad variety of intra-cloud and inter-cloud network resources to be regarded. The rapid development in mobile and networking technology has led to comprehensive data-centered activities being carried out. This results in critical need for energy efficient task scheduling schemes for data centers. The more we send emails, watch online videos, do business online, and use social media such as Facebook, the greater the demand grows for data centers. The rising challenge is how to provide services effectively to meet such enormous demands with improved service quality, low energy usage and minimal greenhouse gas emissions. This paper will try to show a few methods to handle this challenge, and hence promote energy efficiency in cloud computing.

Keywords

Cloud Computing, Virtual Machining, Data Centers, Energy Efficiency, Processor Scheduling, Containers, Optimization, Resource Management, Dynamic Scheduling

1. INTRODUCTION

Cloud computing is a computing standard that delivers on demand a highly manageable, robust and flexible platform for various distributed computing applications through a set of geographically separate data centers [2]. The advent of cloud computing technology has turned the IT industry into a modernized environment, and massive cloud data centers are serving millions of internet-based demands for computing, storage and networking, and application resources based on the pay-for-use model [5]. Cloud computing is the most influential computing method in this modern age of information technology.

The data centers have expanded exponentially and along with that their power consumption. The energy consumption leads to high operating costs and great environmental impacts. Electricity demand for data centers is expected to increase by more than 66 percent over the period 2011–2035 [4]. As a result, significant research has been conducted on how to reduce the power consumption of data centers. Servers, communications network, and the cooling system are the major sources of power consumption in a data center. An idle server has been estimated to use about 70 percent of its peak capacity. In its latest report, Amazon estimated that their data centers consumed up to 52 per cent of the cloud system's overall energy and pushed the cloud maintenance expense to 42 per cent. Owing to the drastic effect of energy consumption on the scalability of data centers, businesses are focused on energy efficiency [2].

The graph shows all existing solutions and ways to manage the energy consumption of resources in cloud DCs.

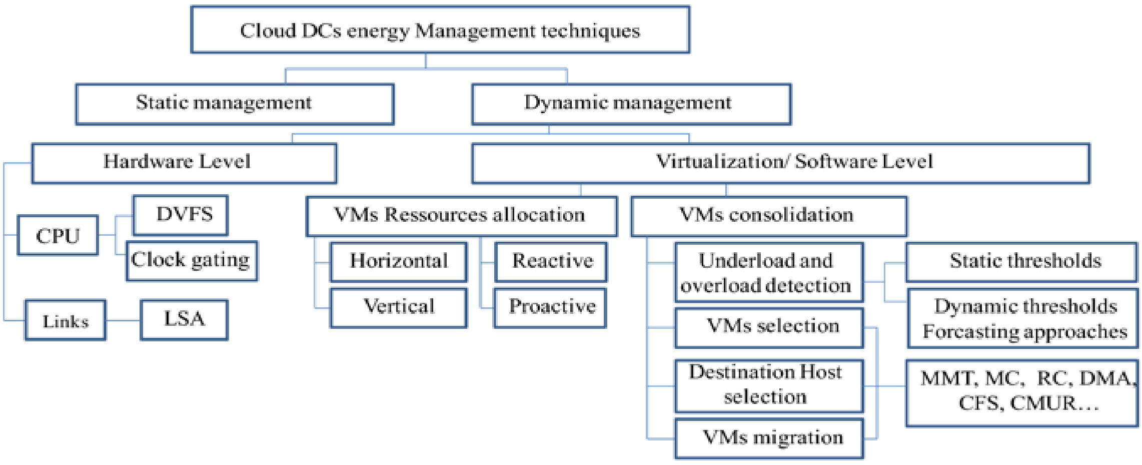

Cloud DCs energy Management techniques
Static management
Dynamic management
Hardware Level
Virtualization/ Software Level
CPU
DVFS
Clock gating
Links
LSA
VMs Ressources allocation
Horizontal
Vertical
Reactive
Proactive
VMs consolidation
Underload and overload detection
VMs selection
Destination Host selection
VMs migration
Static thresholds
Dynamic thresholds Forcasting approaches
MMT, MC, RC, DMA, CFS, CMUR…

Fig.1.1 Energy management techniques in cloud DCs

1.1 Methodologies included

I will be summarizing the following energy aware cloud computing techniques in the due course of this paper

1. Virtual resource management, talking about the effective management of available resources to achieve low power consumption when working with Virtual machines
2. Container resources management, which talks about reducing power usage by containerizing all the applications running on the cloud for faster and lighter execution
3. A dynamic and greedy approach for allocating tasks onto appropriate VMs based on various criteria for scheduling
4. A temporal load-aware strategy that depicts energy consumption as a function of time. VMs are assigned to jobs in slotted time intervals.
5. VM allocation in an untrustworthy environment. VMs are allocated to host machines only if the host machine can be trusted. Energy consumption reduced by minimization of number of migrations
6. A Best-fit scheduling algorithm that uses the optimized Bin Packing method for assigning jobs to VMs

And a few more techniques are discussed in brief at the end of this paper.

2. Virtual Resources Management

Recent emerging research focuses on reducing energy consumption while providing resources dynamically and without sacrificing the quality of data center services. Provisioning of dynamic resources is not a trivial problem to manage, since it requires several processes such as repeated migrations and reallocation of VMs. In general, dynamic techniques are the most popular due to the enormous variability of on-demand workloads.

Dynamic scheduling can be decomposed to a hardware or software level. Scheduling at virtualization or software level can be either [1]

- reactive: which means that planning takes place following the arrival of customer requests, or
- proactively: when we talk about the anticipated scheduling that predicts future demands and takes place in real time with customer demands
- horizontal planning: when it comes to creating multiple VMs and allocating resources to them,
- vertical scheduling: it refers to the reallocation of resources to a single VM as workloads change.

In this part, we focus on dynamic resource management approaches aimed primarily at minimizing energy consumption or carbon emissions as well. This section represents the most common technique used for efficient allocation and provision of VM resources for under-utilized DCs and servers.

2.1 Bin-Packing Technique

Bin-packing (BP) technique is commonly used to accomplish a main objective of consolidating VMs on the minimum number of servers while allocating resources to VMs. As shown in Fig. 2.1, this technique considers servers or hosts as bins, VMs as items to be packed, while physical resources (such as CPU, memory, bandwidth) are viewed as dimensions. [1]

Recent BP algorithms use or improve generally some classical BP heuristics such as First Fit Decreasing (FFD) or Best Fit Decreasing combined with various optimization methods. In this section, we will leverage the most relevant research in this field, OptiDiv.

2.1.1 OptDiv

Fares Alharbi in [10] proposed this approach to solving the bin-packing problem optimally in a polynomial time. He suggests an offline algorithm that extends the FFD heuristic with informed power consumption-related decisions, and the optimal bin selected is the one that has the minimum cost function.

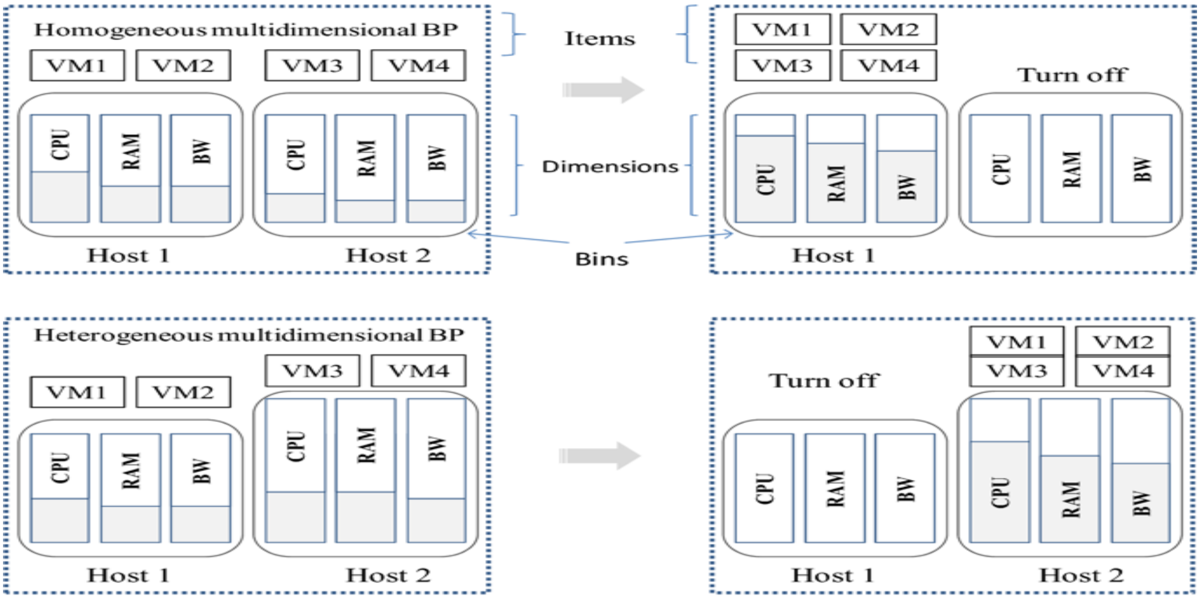

Homogeneous multidimensional BP
VM1
VM2
VM3
VM4
CPU
RAM
BW
Host 1
Host 2
Items
Dimensions
Bins
Turn off
Heterogeneous multidimensional BP

Fig 2.1 Homogeneous and heterogeneous multidimensional bin packing

This algorithm returns an optimal solution if and only if the input is weakly divisible (i.e., in a non-increasing order of item size, the last item size is a divisor of the size of all items previously packed) and the cost functions are monotonically increasing and concave. Thus, OptDiv leads to higher power consumption savings; however, compared to FFD and other heuristics, it is still not fast, and it is not clear whether it can perform heterogeneous binning.

2.2 Dynamic Virtual Machine Consolidation (DVMC)

In [11], the author proposes a new energy-aware DVMC dynamic VM consolidation mechanism which can be carried out by heterogeneous hosts and various types of VMs.

This process consists of four stages:

- A monitoring phase for gathering information related to resources, energy consumption, on-off status, and workload.
- Analysis and estimation of the workload for the identification of overloaded and light-loaded hosts using different policies as: interquartile range (IQR), robust local regression (LRR), local regression (LR), and median absolute deviation (MAD)
- VM migration selection decisions and placement phase of migrated VMs
- Actuation stage for performing the VM migration and turning on or off of the respective physical host(s) during this phase

Significant efforts have been made to develop cloud virtual resource allocation policies. Those policies, however, may or may not adapt to containers. Existing resource allocation policies for VMs might therefore need to be refined and further optimized for containers.

3. Container Resources Management

Container as a service is a new layer that has recently been largely deployed in the cloud environment for its significant energy efficiency benefit. Containers are also characterized, in the same way as VMs, by their requests for several resources. However, it usually takes only a few seconds to start the container, as the application in the container is partially disassociated from the operating system. They are therefore lighter and easier to migrate. [1]

But unlike VM migrations, migration of containers can be more complicated in practice as it requires a significant amount of OS state that must be copied along with the memory pages. The problem then arises as to how to configure containers, and whether to migrate a VM, a container, or an activity inside a container to increase energy efficiency.

Some believe that the combination of the two sand-boxing technologies (VMs and Containers) is more effective (Fig. 3.1-a). Others find the use of containers on bare metal is more efficient in terms of energy consumption (Fig. 3.1b).

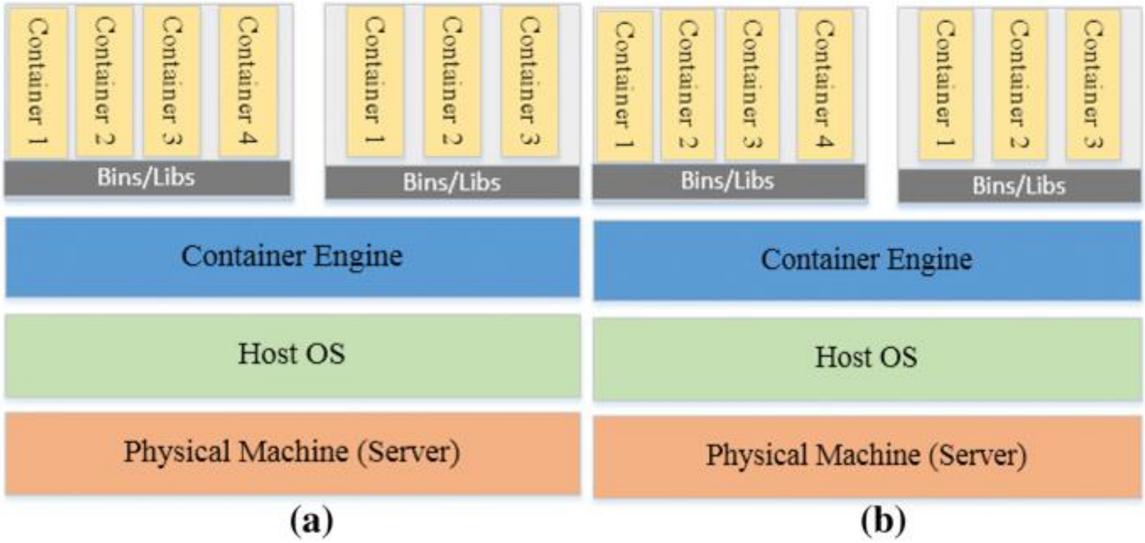


Fig. 3.1 (a) Containers on the top of VMs, (b) Containers on bare metal

Starting with combination of VM & containers, two frameworks are discussed in [1] below

3.1 Host Status

The host status module starts with a component of host underload / overload detector which compares utilization with two static thresholds. A container selector is then based either on an analysis of the correlation between container load and host workload, or on the container with the highest utilization.

3.2 Consolidation

This module selects destination hosts for these containers using two different algorithms based on the least complete host selection strategy and Correlation Host Selection (CorHS) strategy.

Results from several experiments show that "CorHS" is the most energy-efficient algorithm for selecting host with 7.45 % lower energy consumption and less than 5 % SLA violations as compared to other algorithms (random, FFD, least CPU utilization). Nonetheless, these findings indicate the same number of container migrations, as decisions for migration were focused on host load rather than VM load. [1]

Some other studies see containerization as a viable alternative to conventional hypervisor-based virtualization, using Docker container technology in particular.

3.3 Containerization

Because the energy consumption and efficiency of the container is primarily influenced by the number of cores allocated to it and their frequency, this author [12] focuses on the energy efficiency of the problem of core allocation for containers in a DC and determines the number of core, core frequencies and core deployment per container

This analysis considers 2 scenarios:

- Single server
- Multiple server

Both scenarios take DVFS per server and DVFS per container into account

As the problem was split into two sub-problem scenarios, dynamic programming was first proposed in order to achieve the optimal allocation of cores, taking into consideration the energy consumption of the previous allocation of cores, using recurrence [1]. However, in order to produce a more scalable, rapid and effective result, greedy heuristics and other consecutive allocation heuristics have been proposed.

Study on container resource management [1], as described above, is scarce and still causes significant confusion. For most cases, the design of containers on bar metal may be more energy efficient and appear to replace virtual machines. However, for several security reasons, it is important to incorporate both VMs and containers in order to benefit from the strengths of both sandboxing technologies. Further valuable studies need to be conducted in this direction.

4. Dynamic and Greedy Scheduling Algorithm

Conventional scheduling algorithms contrast with real-time task execution patterns in unpredictable cloud environments, since these schemes presume cloud computing environments are deterministic and pre-defined task schedules are available. These algorithms also do not consider pre-empting tasks from a VM and taking due account of temperature effects on VM.

To solve this problem, the authors in [2] have proposed a new algorithm that takes into account the priority of the user and the deadline for their requested tasks. They propose an energy-conscious, pre-emptive job scheduling algorithm that assigns tasks to correct VM which is capable of assigning CPU, memory, storage, SLA requirements to task, able to tolerate the increase in temperature due to the execution of the job and left with the max energy residue. Fig 4.1 shows the flow of this proposed algorithm

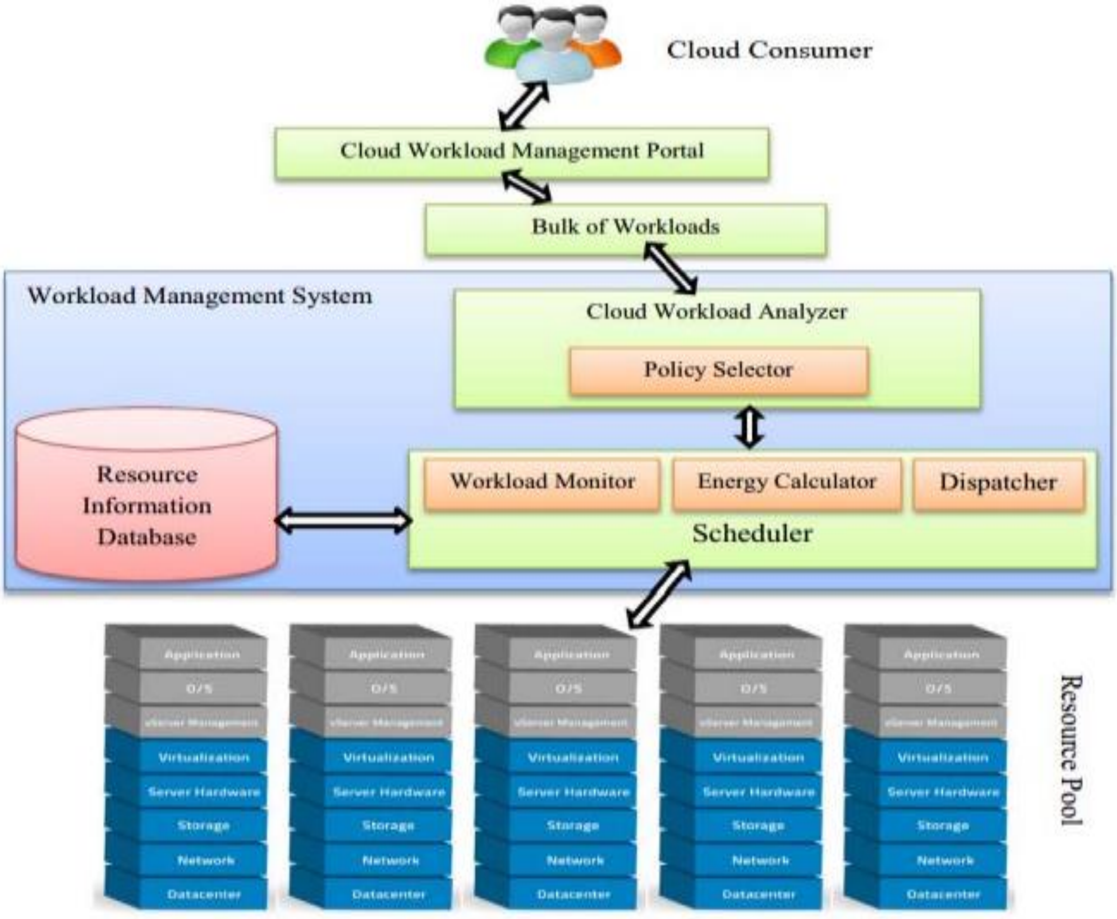


Fig. 4.1 Dynamic and Greedy resource scheduling architecture

4.1 Proposed Algorithm

In this proposed algorithm, on obtaining the task to be scheduled, the task scheduler will search for an active VM to allocate the task to. The selection will be performed based on these 3 criteria: [2]

- Must be capable of allocating CPU, memory, storage requirements of the task and be able to tolerate temperature changes due to task execution and on task allocation.
- The task (waiting time + processing time) must be completed within the required SLA.
- If more than one VM meets the above conditions, the scheduler opts for the VM having the maximum remaining CPU, memory, storage and temperature capacities after the execution of task.

If no active VM is able to assign the necessary resources to complete the task within the specified SLA, the scheduler will generate a new VM with the needed resources and add it to the active VM queue. If the VM has reached its threshold capacity, to optimize energy output, the algorithm is pre-empting the task and assigning it to another VM that is currently active.

They [2] are viewing the energy consumption of a VM as an integer programming problem with the goal of reducing the consumption of energy by VM and maximizing its residual energy capacity.

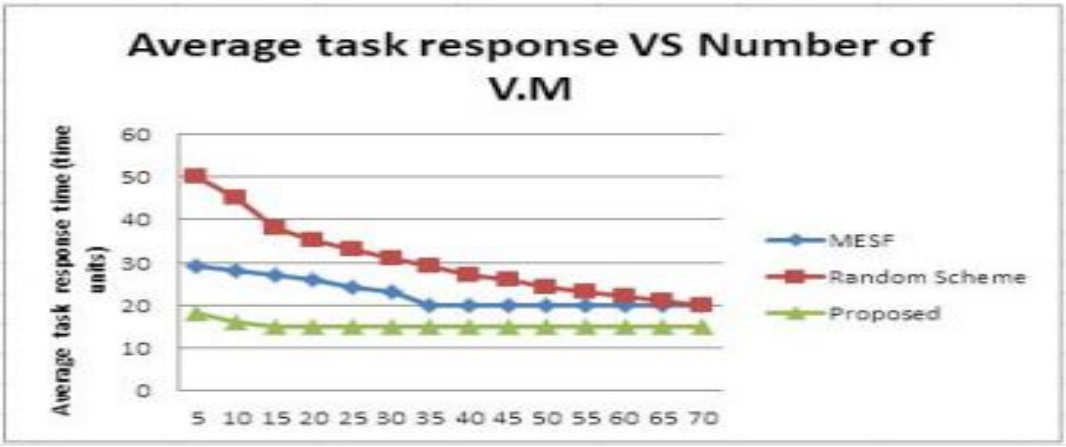
Residual Energy = Max Energy Capacity - Processing Energy Requirements

Where the maximum and current CPU, storage, memory, temperature and processing capacity of each VM is taken in to consideration along with the CPU, storage, memory and processing requirements of the current task that is to be scheduled by the algorithm.

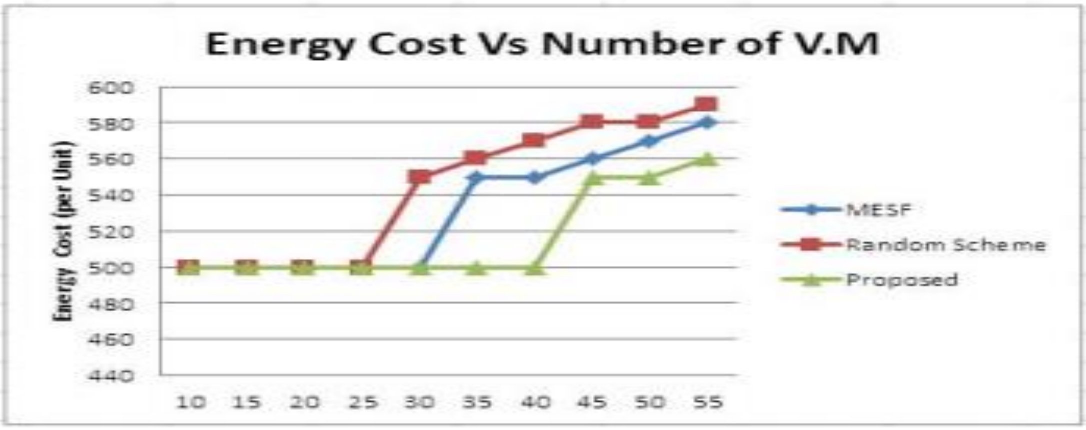
The complete algorithm and it’s step by step explanation have been elaborately provided in [2]

4.2 Results and Analysis

The authors [2] have evaluated the proposed scheme on a commonly used simulation framework called CloudSim. Rigorous simulations were performed with a number of distinctive workloads based on real-life data center usage. In order to check the effectiveness of our algorithms, we tested many key aspects of the algorithm such as AVERAGE TASK RESPONSE, ENERGY COST and compared them with other proposed MESF, Random Scheme algorithm. The following figure Fig 4.2 depict this comparison accurately



(a)



(b)

Fig. 4.2 (a) Average task response time (b) Total energy cost vs number of V.M

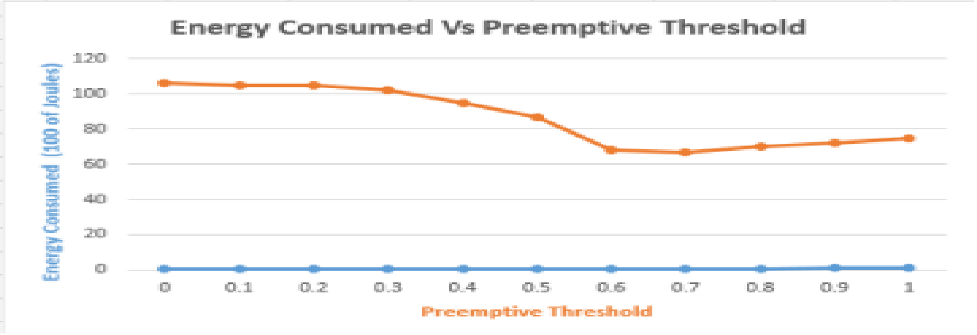


Fig. 4.3 Energy consumption vs Pre-emptive threshold

As seen in Fig. 4.3 above, as the proposed threshold increases, energy consumption decreases as a result of task pre-emptions. [2]

Thus, in this section we have summarized the proposed dynamic and greed scheduling algorithm presented in [2] which dynamically allocates the task to the most energy-efficient VM, based on an integer programming optimization problem. In the future, this algorithm can be elevated to evaluate its performance based on some more critical parameters.

5. Temporal Load-aware Resource allocation

Energy optimization in a data center is a challenging task due to server resource limitations, network topology and bandwidth constraints, migration costs for VMs, heterogeneity of workloads and servers. The incoming new jobs and the outgoing completed jobs also create heterogeneity in terms of the workload in time [4]. As a result, much of the previous research was focused on partial optimization of power consumption, optimizing either server and/or network power consumption through VM placement. Temporal Load-aware optimization has been researched on extensively.

This VM placement [4] technique focuses on energy consumption as a function of time, which also is a dynamic scheduling approach. It is assumed that the time-axis is slotted and VMs are allocated to tasks in unit slot times. The authors [4] presume that the arrival of workers in the system is in compliance with the Poisson method, although the analysis is applicable to other incoming processes. The newly arrived tasks in the current slot and residual jobs from the current slot will be scheduled for operation in the next slot.

5.1 Approach

They [4] consider 2 types of operational disciplines that the jobs can follow while releasing its VMs at the end of every slot, in accordance to the Bernoulli trials:

- The job releases all its assigned VMs simultaneously. As a result, the residual job will need full complement of its VMs
- The job releases all its assigned VMs individually. Then the residual job needs only a subset of the total VMs it currently holds

Now at the beginning of the next time slot, the algorithm will schedule the new incoming jobs and the residual unfinished jobs from the previous session, such that there is minimum power consumption in this new time slot.

For scheduling the residual jobs from the previous session, the system [4] will follow one of the 2 approaches mentioned bellow, based on whether migration of VM is permitted.

- If migration of VMs is permitted, the residual jobs are treated as new jobs
- If migration of VMs in not permitted, then the new freshly arrived jobs may have to wait for some VM to get free, as they might still be occupied by the unfinished jobs

As a consequence of migration, the system may end up in a lower power consuming state. But migration itself has its own communication and processing overheads that need to be taken care of by optimization.

Suppose G_r denotes the normalized cost of power consumption during migration VMs of type r. Then the equation denoting G_r is as follows

$$G_r = (VM\ MigrationpowerConsumption\,(Source + Destination)) \times \frac{MigrationDuration}{TimeSlotDuration}$$

5.2 Experimental results

This section presents some numerical results of experiments done by the authors [4] regarding the analysis of the algorithm presented above. The performance of the system is plotted as a function of discrete time and newly arrived jobs are distributed at the data center in compliance with the Poisson method. VMs are released according to independent Bernoulli tests of the jobs in the system.

Fig. 5.1 shows the optimal power consumption of the system plotted against the number of time slots for both, simultaneous and individual VM releases for job with migration cost G_r = 0.3

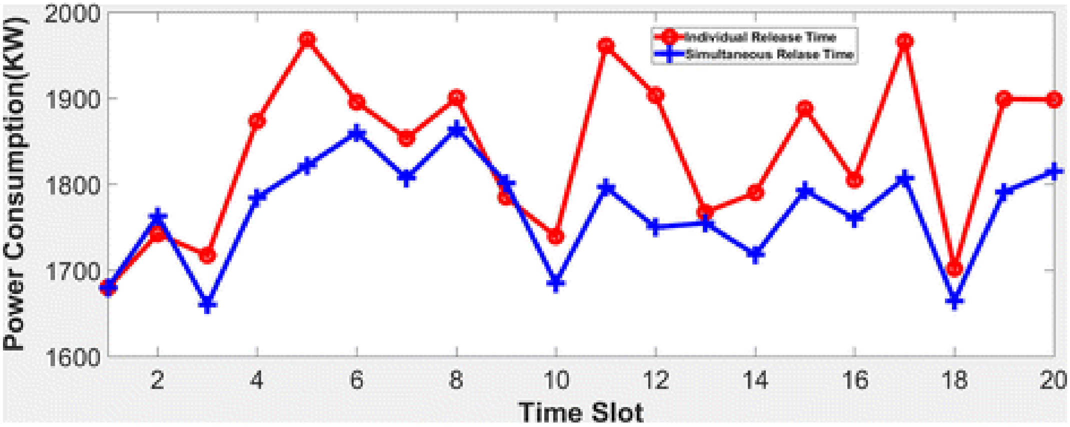


Fig. 5.1 Optimal power consumption vs Number of time slots for simultaneous and individual VM release

Fig. 5.2 shows optimal power consumption of proposed placement algorithm along with deterministic and random heuristics placement algorithms as a function of time with migration cost G_r = 0.3

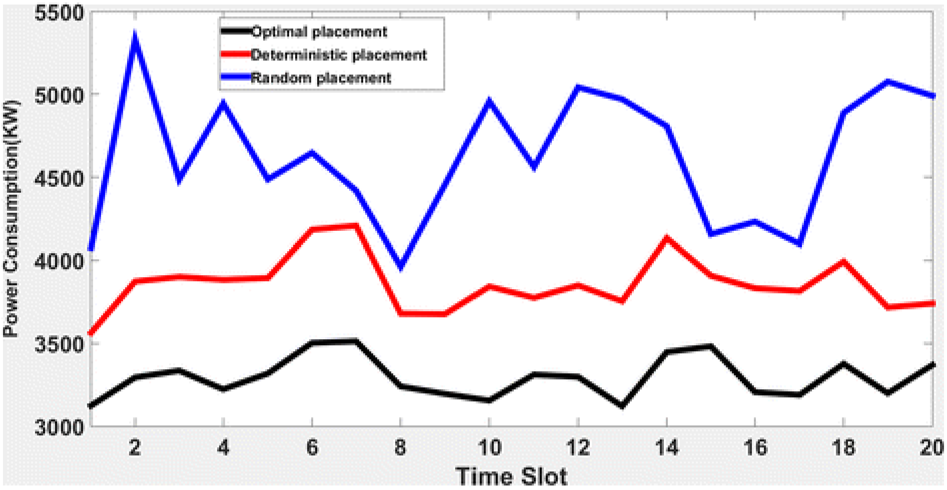


Fig. 5.2 Optimal power consumption as function of time

The proposed method of optimization could also be used for container-based systems instead of VMs. This proposed optimization will help cloud service providers to achieve energy savings.

6. Secure VM Allocation in Untrusted Environment

Data stored on the cloud belongs to various organizations and individual users, and it must be protected from malicious individuals. Several security attacks have been reported that threaten the fundamental principles of security. Data is at risk from not only outsiders, but even malicious insiders who might want steal valuable information of their competitors using these cloud services. Since cloud users have little power inside the cloud, it is therefore necessary for them to trust and understand the security policies of cloud service providers.

The solution proposed [8] in this section involves performing VM allocation in the presence of an untrusted cloud computing environment, keeping energy conservation in mind. This algorithm involves

- associating a trust factor with every host machine in the data center
- Resource allocation on untrustworthy host machines in the data center

6.1 Trust Based VM Allocation

The following fig. 6.1 represents the flow of this VM allocation strategy [8], taking Trust into consideration as an additional parameter.

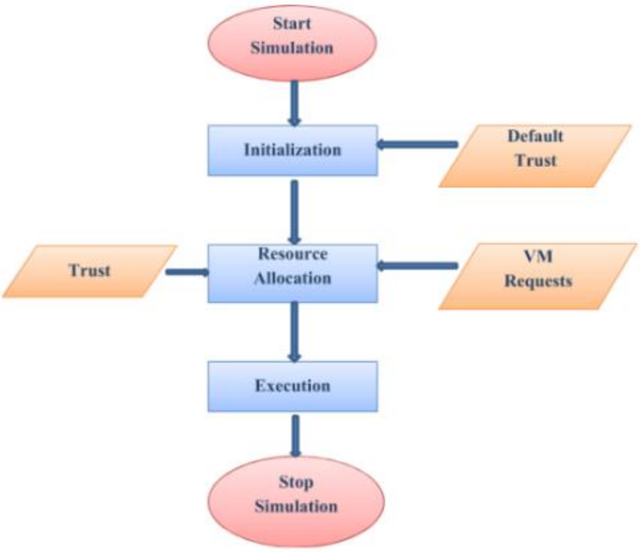


Fig. 6.1 Flowchart for VM allocation taking Trust into consideration

- All the host machines are initialized with the default trust value=1 after creation.
- In the data center, the VM is allocated to the host machine only if it trustworthy. And the host machine has enough capacity (i.e. availability of resources) to accommodate the VM
- After the allocation, execution of the tasks can begin and at the end, the created VMs are destroyed

6.2 Trust computation

Trust between 2 entities can be formed based on 2 methods:

- Trust (Direct) formed between the entities based on past knowledge or previous experiences. Initially a default value, trust keeps on changing periodically based on the entity's info
- Trust (Indirect) formed between the entities based on recommendations provided by a trusted 3rd-party entity.

If a party, say A, wants to interact with another party, say B, the party A will ask another 3rd-party entity, say C, to obtain the trust value for B.

Similarly, before assigning a host machine to any VM or before migrating VM to another host machine, the trust value of the host machine is calculated. The VM is allocated to the Host machine if and only if the host machine is found to be trustworthy. Else another host machine is selected.

Suppose a VM on host machine A is to be migrated to host machine B, A will compute the trust value of B by asking for recommendation from the data center (here, the trusted 3rd-party) as follows

$$T\ (\text{HostA}, \text{HostB}) = T\ (\text{HostB}) \times T\ (\text{HostA}, \text{DataCenter})$$

Where,

T (HostA, HostB) = Trust value of host machine B calculated by host machine A.

T (HostB) = Trust value of host machine as recommended by data center.

T (HostA, DataCenter) = Trust value of data center with respect to host machine A (assumed to be 1).

The Trust of every host in a data center is computed periodically on the basis of two factors:

- SLA violation: SLA breaches usually occur when CPU output demands exceed the available capacity, i.e. when resource needs are not fulfilled. This is to be minimized.

- Fault Tolerance: The rate at which faults occur at any given host determines the trust value of that host system/.

Thus, the proposed approach [8] minimizes the energy consumption in the datacenter present in an untrustworthy, unreliable clould computing environment by taking into consideration the trust value of the host along the available capacity of the host to handle the migration and allocation of a VM

The authors [8] have also provided a detailed algorithm that can be implemented easily.

6.3 Results and analysis

The trust value of any host regularly increases or decreases based on the percentage of SLA breaches and the frequency of faults on that host.

- If Trust value < 1.0 (ignorance value), VMs will not be assigned to that particular host
- If Trust value > 1.0, VMs can be assigned to the host if it has the available resources

The default (without Trust concept) and the Trust based concept can be compared using the following charts made when simulating the approach using the CloudSim tool.

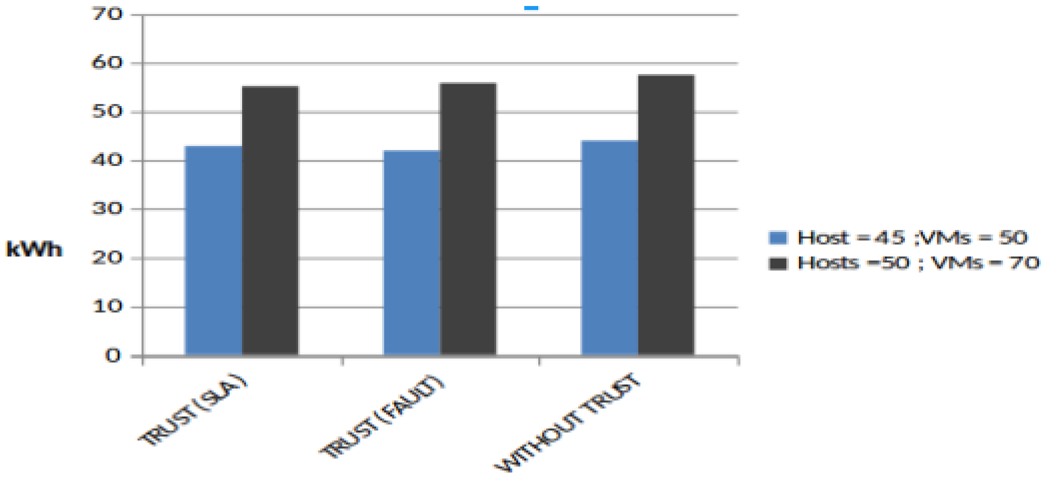


Fig. 6.2 Energy consumption in the absence and presence of Trust based model

This decrease in energy consumption as shown in Fig. 6.2 can be attributed to the marginal decrease in the number of migrations needed as migrations to host systems with trust value lower than the ignorance value were aborted.

Fig. 6.3 depicts the comparison of VM migrations required by both, Trust based and without Trust based models.

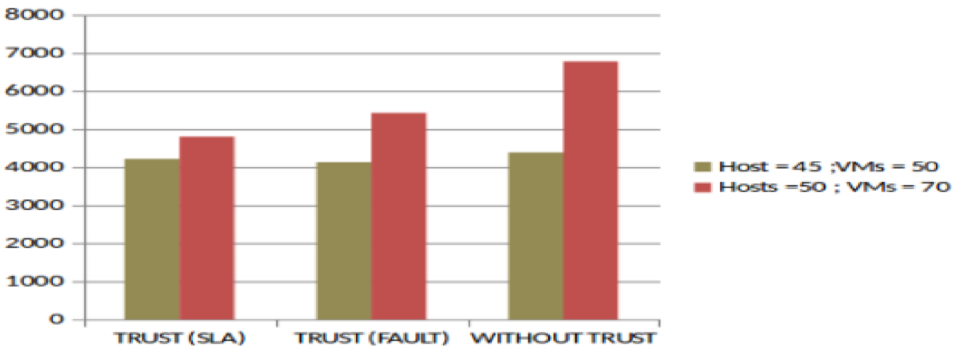


Fig. 6.3 Number of migrations in the absence and presence of Trust based model

7. Best-Fit Virtual Machine Allocation

As seen in the previous methods, VM allocation is a pretty fundamental resource in the Infrastructure as a Service (IaaS) sector. The Bin-Packing method had already been explained in Virtual Resource management (Pg. 3, topic 2.1). This section [5] also uses the Bin Packing method in its model, but also optimizes the Bin Packing algorithm by using the Best-fit (BF) scheme. This paper [5] consider the VM's memory and CPU demands while allocating it to any physical machine. The proposed VM placement algorithm would maximize the usage of physical servers and provide a speedier response mapping of the VM to the data center.

7.1 System Model

The system model for the proposed algorithm [5] is as shown in Fig. 7.1

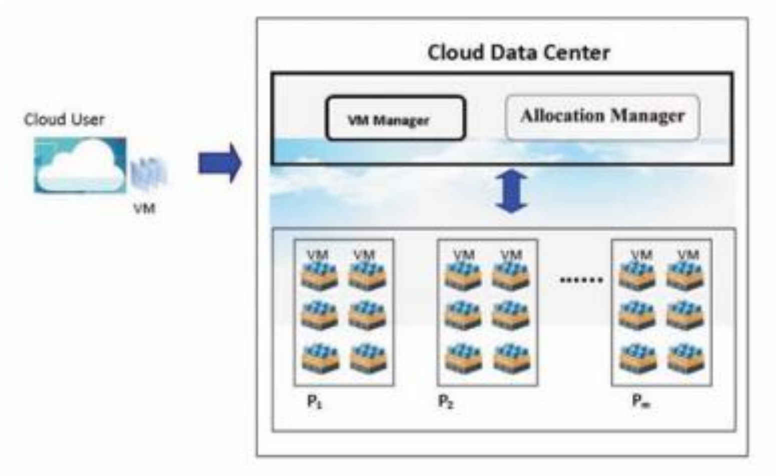


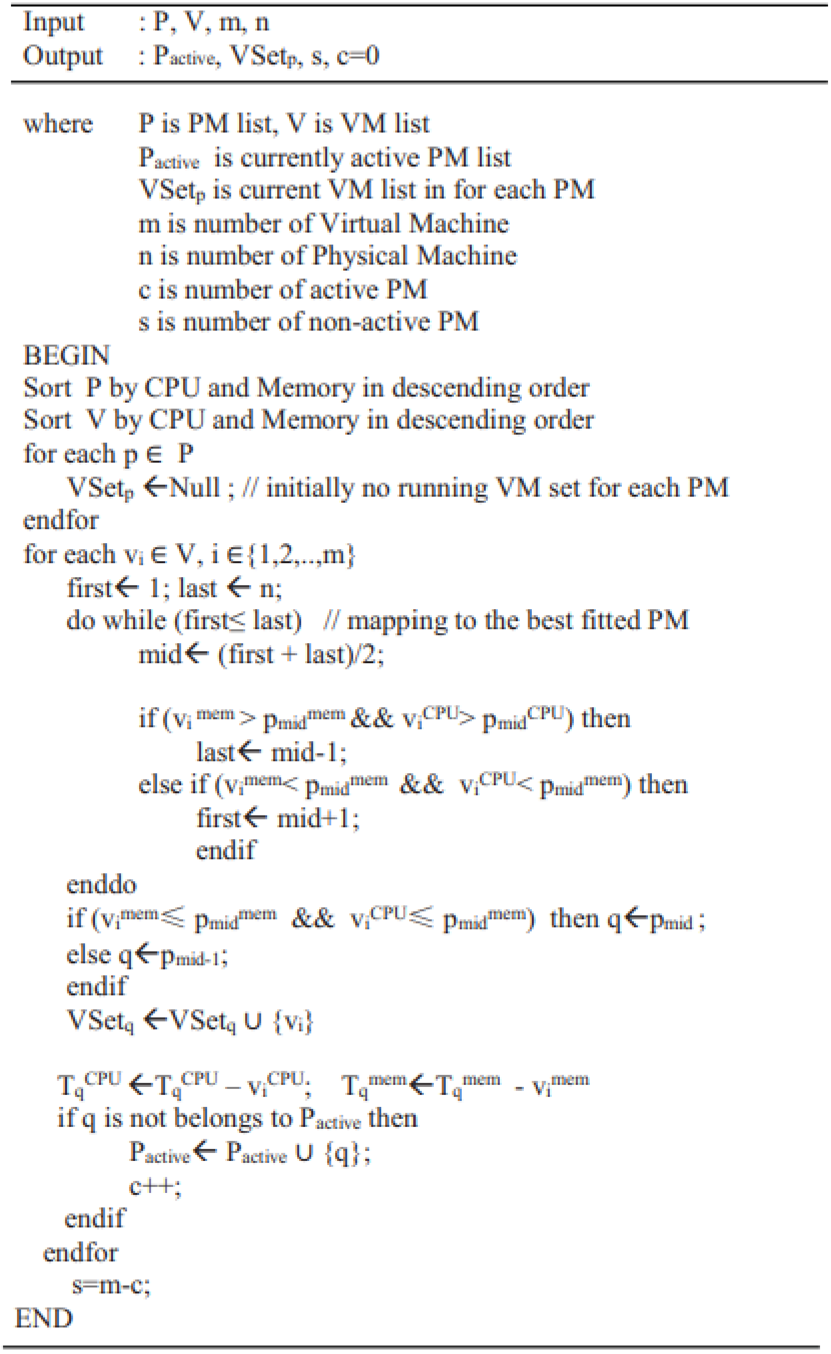
Fig. 7.1 Best-fit Bin Packing system model

Given below is the flow of the process followed by this approach:

- VM Manager collects and stores incoming VMs in a VM list at a regular interval of time, waiting for allocation in the list.
- The VM Manager then sorts the VMs and the physical machines in the list in decreasing order for the Allocation Manager, taking into consideration the memory and CPU resource demands of each VM.
- The Allocation manager takes the VM at the top of the list provided by the VM manager and assigns the most appropriate physical machine using the proposed algorithm.

7.2 Proposed Algorithm

The following figure Fig. 7.2 shows the algorithm that will be used to choose the most suitable Physical Machine. [5]



```
Input     : P, V, m, n
Output    : $P_{active}$, $VSet_p$, s, c=0

where     P is PM list, V is VM list
          $P_{active}$ is currently active PM list
          $VSet_p$ is current VM list in for each PM
          m is number of Virtual Machine
          n is number of Physical Machine
          c is number of active PM
          s is number of non-active PM
BEGIN
Sort  P by CPU and Memory in descending order
Sort  V by CPU and Memory in descending order
for each p ∈ P
    $VSet_p$ ←Null ; // initially no running VM set for each PM
endfor
for each $v_i$ ∈ V, i ∈{1,2,..,m}
    first← 1; last ← n;
    do while (first≤ last)   // mapping to the best fitted PM
          mid← (first + last)/2;

          if ($v_i^{mem} > p_{mid}^{mem}$ && $v_i^{CPU} > p_{mid}^{CPU}$) then
               last← mid-1;
          else if ($v_i^{mem} < p_{mid}^{mem}$  &&  $v_i^{CPU} < p_{mid}^{mem}$) then
               first← mid+1;
               endif
    enddo
    if ($v_i^{mem} \leqslant p_{mid}^{mem}$  &&  $v_i^{CPU} \leqslant p_{mid}^{mem}$)  then q←$p_{mid}$ ;
    else q←$p_{mid-1}$;
    endif
    $VSet_q$ ←$VSet_q$ ∪ {$v_i$}

   $T_q^{CPU}$ ←$T_q^{CPU}$ – $v_i^{CPU}$;    $T_q^{mem}$←$T_q^{mem}$  - $v_i^{mem}$
   if q is not belongs to $P_{active}$ then
         $P_{active}$← $P_{active}$ ∪ {q};
         c++;
   endif
  endfor
    s=m-c;
END
```

Fig. 7.2 VM Allocation using Best-fit Bin Packing algorithm

The complexity of this efficient Binary search algorithm is *O (log n)* which is much faster than the traditional sequential search algorithm. Since this algorithm directly provides the best suitable physical machine for the VM, lesser number of migrations are need and hence the energy consumption also experiences a significant drop.

8. More Energy Efficient Cloud Computing techniques

Here I share some more energy-efficient cloud computing techniques in brief, that should be taken look at, before deciding which approach is best suitable for your purpose

8.1 Location-Aware Virtual Network Embedding [3]

Several two-stage VNE algorithms with topology-aware node ranking methods dependent on topological attributes (node CPUs and link bandwidths) have been suggested to improve the acceptance ratio of incoming Virtual network requests and the total Substrate Network revenue.

This paper [3] present two very energy-efficient algorithms to reduce power consumption in software-defined Optical DC networks.

- A two-stage VNE algorithm focused on ranking of Global Topology Resource, GTR nodes (GTR-VNE)
- A coordinated VNE algorithm known as ACO-VNE based on the artificial intelligence Ant Colony Optimization (ACO) technique, taking into consideration the GTR nodes

VNE is one of the key problems in the virtualization of networks. Compared with GTR-VNE, overall power consumption decreased by 18.8% and 28.7% respectively, while the acceptance ratio increased by about 5%. This paper [3] will help you to reduce your power costs if you want to optimize you DCs by virtualizing its network using these two techniques.

8.2 QoS-constrained workflow scheduling [6]

This form of scheduling is more widely used in actual real-time applications. There is always a trade-off; if one attempts to minimize some variables, then another increases automatically. This approach is used to handle this trade-off.

The goal is to optimize one parameter, while constraining the other. This method generates a schedule according to the defined QoS constraints met by the preferred parameter.

8.3 Min-Max Scheduling [6]

In this approach, just one is to be pursued without consideration of other goals such as QoS factors, etc. For example, considering conservation of energy / cost as the only constraint without focusing on SLA violations or QoS requirements.

8.4 Dynamic Performance Scaling [6]

DVFS decreases power consumption by scaling frequencies (up / down) as and when required. If a CPU runs at a lower frequency, it would require less voltage and hence lesser power consumption. Voltage and frequency are therefore dynamically balanced, resulting in reduced power consumption.

It can take longer to finish a job, if the processor is running at lower frequencies. The processor usually begins at lower frequencies and increases gradually with workloads, and so does the energy consumption. A major challenge in reducing power consumption through DVFS is handling constraints with deadlines.

8.5 Dynamic Component Deactivation [6]

It involves the deactivation or shutdown of components in idle (not used) condition. This is one of the simplest methods out there to implement. Such transitions, however, can in some cases cause performance degradation and delays that draw additional power.

A transition should therefore only be performed if the idle timing is sufficient to pay for overhead during transitions.

9. Conclusion

This paper includes a survey of some of the energy-aware algorithms for reducing power consumption in Cloud data centers, proposed by some brilliant researches. This paper is just a summarization of their extensive efforts and suggest reading their papers mentioned below for a more detailed insight into their valuable research.

The use of large volumes of Cloud DCs results in an immense amount of energy usage which creates huge quantities of high carbon emissions, which have become the biggest problem of the 21st century. It is therefore necessary to manage both energy and QoS together in order to obtain sustainable and energy-efficient cloud services. [7]

Existing energy aware resource management methods and strategies focus primarily on consolidating VMs to reduce server-side power consumption only. However, other services including power, processing, cooling, and networks, they too consume a massive amount of energy. SLAs, QoS and power consumption must be handled simultaneously. [7]

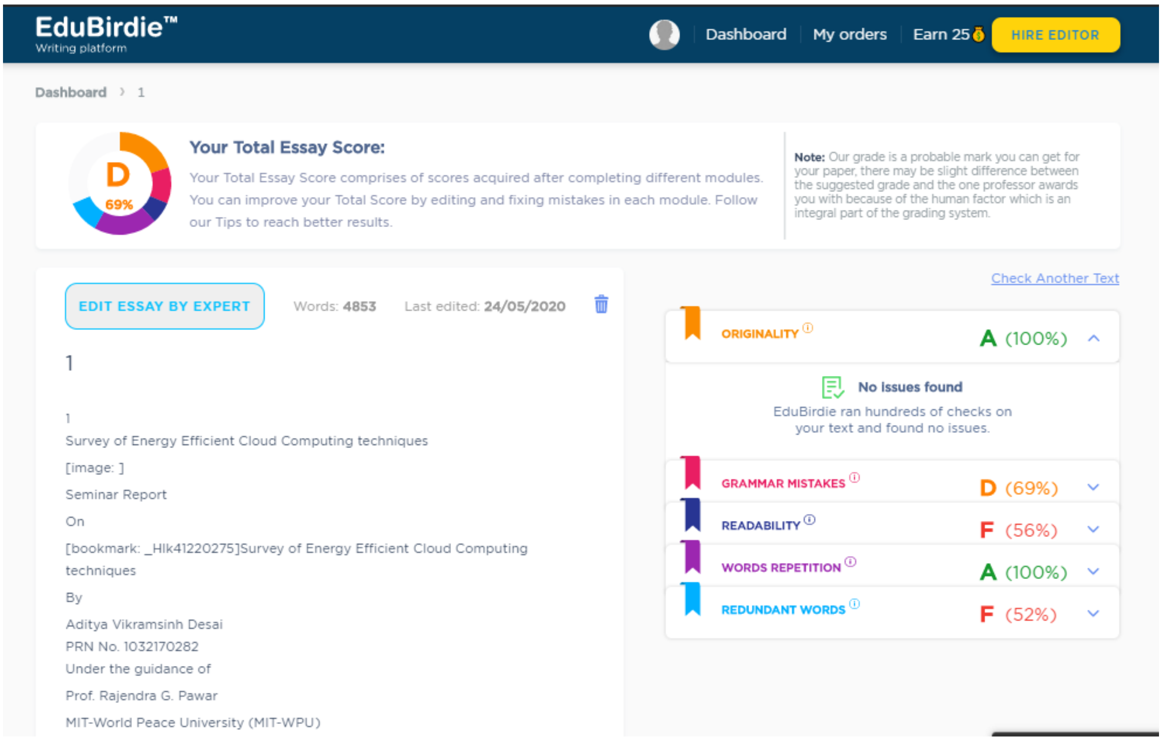
None of them can fix all issues, because each of them performs better in a particular requirements and specifications environment. Nonetheless, it has been concluded many times that a clever mix of approaches could resolve over-fitting and local convergence problems. Moreover, regarding the challenges faced by the cloud, there are many more challenges out there, to be solved.

REFERENCES

1. Hamzaoui, I., Duthil, B., Courboulay, V. et al. A Survey on the Current Challenges of Energy-Efficient Cloud Resources Management. SN COMPUT. SCI. 1, 73 (2020). https://doi.org/10.1007/s42979-020-0078-9
2. M. Sarvabhatla, S. Konda, C. S. Vorugunti and M. M. N. Babu, "A Dynamic and Energy Efficient Greedy Scheduling Algorithm for Cloud Data Centers," 2017 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM), Bangalore, 2017, pp. 47-52, doi: 10.1109/CCEM.2017.9
3. Y. Zong et al., "Location-aware energy efficient virtual network embedding in software-defined optical data center networks," in IEEE/OSA Journal of Optical Communications and Networking, vol. 10, no. 7, pp. 58-70, July 2018, doi: 10.1364/JOCN.10.000B58
4. Vakilinia, S. Energy efficient temporal load aware resource allocation in cloud computing datacenters. J Cloud Comp 7, 2 (2018)
5. K. M. Nwe, M. K. Oo and M. M. Htay, "Efficient Resource Management for Virtual Machine Allocation in Cloud Data Centers," 2018 IEEE 7th Global Conference on Consumer Electronics (GCCE), Nara, 2018, pp. 419-420, doi: 10.1109/GCCE.2018.8574804
6. Khattar, N., Sidhu, J. & Singh, J. Toward energy-efficient cloud computing: a survey of dynamic power management and heuristics-based optimization techniques. J Supercomput 75, 4750–4810 (2019)
7. Sukhpal Singh Gill and Rajkumar Buyya. 2018. A Taxonomy and Future Directions for Sustainable Cloud Computing: 360 Degree View. ACM Comput. Surv. 51, 5, Article 104 (December 2018), 33 pages
8. Swati Sharma and Rishabh Kaushal. 2017. Energy Conserving Secure VM Allocation in Untrusted Cloud Computing Environment. In Proceedings of the 10th Annual ACM COMPUTE Conference, India (ACM COMPUTE 2017). ACM, New York, NY, USA, 9 pages
9. Deafallah Alsadie, Zahir Tari, Eidah J. Alzahrani, and Albert Y. Zomaya. 2018. Dynamic resource allocation for an energy efficient VM architecture for cloud computing. In Proceedings of the Australasian Computer Science Week Multiconference (ACSW ’18). Association for Computing Machinery, New York, NY, USA, Article 16, 1–8

10. Halácsy G, Zoltán ÁM. Optimal energy-efficient placement of virtual machines with divisible sizes. Inf Process Lett. 2018; 138:51–6. https://doi.org/10.1016/j.ipl.2018.06.003.
11. H. Wang and H. Tianfield, "Energy-Aware Dynamic Virtual Machine Consolidation for Cloud Datacenters," in IEEE Access, vol. 6, pp. 15259-15273, 2018, doi: 10.1109/ACCESS.2018.2813541.
12. Ching-Chi L, Jian-Jia C, Pangfeng L, Jan-Jan W. Energy-efficient core allocation and deployment for container-based virtualization. In: IEEE 24th international conference on parallel and distributed systems (ICPADS). Singapore; 2018. pp. 93–101. https://doi.org/10.1109/padsw.2018.8644537.

Plagiarism Report



Your Total Essay Score:

Your Total Essay Score comprises of scores acquired after completing different modules. You can improve your Total Score by editing and fixing mistakes in each module. Follow our Tips to reach better results.

Note: Our grade is a probable mark you can get for your paper, there may be slight difference between the suggested grade and the one professor awards you with because of the human factor which is an integral part of the grading system.

Words: **4853** Last edited: **24/05/2020**

1

1
Survey of Energy Efficient Cloud Computing techniques
[image:]
Seminar Report
On
[bookmark: _Hlk41220275]Survey of Energy Efficient Cloud Computing techniques
By
Aditya Vikramsinh Desai
PRN No. 1032170282
Under the guidance of
Prof. Rajendra G. Pawar
MIT-World Peace University (MIT-WPU)

ORIGINALITY	A (100%)
No issues found EduBirdie ran hundreds of checks on your text and found no issues.	
GRAMMAR MISTAKES	D (69%)
READABILITY	F (56%)
WORDS REPETITION	A (100%)
REDUNDANT WORDS	F (52%)