I.J. Modern Education and Computer Science, **2018, 12, 28-35**
Published Online December 2018 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijmecs.2018.12.04

Energy Saving VM Placement in Cloud

Shreenath Acharya
Department of Computer Science, St Joseph Engineering College, Mangaluru, 575028, India
Email: shree.katapady@gmail.com

Demian Antony D'Mello
Department of Computer Science, Canara Engineering College, Mangaluru, 575028, India
Email: demian.antony@gmail.com

Received: 27 September 2018; Accepted: 17 November 2018; Published: 08 December 2018

Abstract**—**The tremendous gain owing to the ubiquitous acceptance of the cloud services across the globe results in more complexity for the cloud providers by way of resource maintenance. This has a direct effect on the cost economy for them if the resources are not efficiently utilized. Most of the allocation strategies follow mechanisms involving direct allotment of VMs onto the servers based on their capabilities. This paper presents a VM allocation strategy that looks at VM placement by allowing server capacity to be partitioned into different classes. The classes are mainly based on the RAM and processing abilities which would be matched with VMs need. When the match is found the servers from this category are provisioned for the task executions. Based on the experimentation for various datacenter scenarios, it has been found that the proposed mechanism results in significant energy savings with reduced response time compared to the traditional VM allocation policies.

Index Terms**—**Cloud, virtual machine, RAM, CPU, energy, response time

I. INTRODUCTION

Cloud is a modern technology that has evolved into all the main streams of computer science. A prime concern in the server virtualization is VM placement. It is the process of mapping the VMs to the servers to suit both the consumers and the providers' expectations. Most of the stated gains/benefits from the cloud comes from the resource multiplexing [1] using virtualization technology that allows to improve resource utilization and energy efficiency.

When there are limited resources in the datacenter, mapping can be done manually. But, when the resources are enormous manual mapping becomes complex and not feasible. This necessitates automated VM placement mechanisms carried out at either initiation time/run time.

In the cloud computing context, resource provisioning is the aspect which guarantees the satisfactory end user/consumer services. The IaaS providers provision the virtual machines as the resource for users job requests. The problem in these [2] type of provisioning is how and where to place the virtual machines owing to users request.

Cloud computing led to the setup of large-scale data centers with thousands of computing nodes consuming large amounts of electrical energy. These data centers incur high operational costs and emit carbon dioxide to the environment leading to the greenhouse effect. The reason may be high energy consumption due to the quantity of computing resources, inefficient hardware and resource usage. Green Cloud computing foresees to achieve efficient processing, infrastructure utilization and also to minimize energy consumption [3]. The virtualization technology allows Cloud providers to create multiple Virtual Machine (VM) requests on a single physical server thereby increasing resources utilization.

There can be incompatibility between user requests in cloud and specification of physical machine, which may lead to problems like poor load balancing, energy-performance trade-off and large power consumption. Reducing energy & better resource utilization could be handled by consolidating the VMs using dynamic migration facilitating idle node switch off to lower powering mode. It was envisaged that idle servers consume upto 70% of its peak power [4,14]. Shifting to lower power modes help to decrease the energy consumption. This improves the performance & provides better quality of service.

The proposed mechanism utilizes 3 classes namely, avid, confronted and intended state. This sets aside resources like RAM and CPU to be used in co-ordination with the users request by considering the factors like response time and energy consumption.

The rest of the paper is organized as follows. Section II presents an overview of the work done, section III depicts problem description, section IV describes system architecture, section V outlines the implementation, section VI shows the results and the conclusion in section VII.

II. RELATED WORKS

Resource allocation is one of the prime challenges in cloud computing. It requires many factors to be considered from the providers' side in order to sustain its market value. Among the factors, energy consumption is of highest priority from economical perspective and response time helps for being reliable.

Many authors have implemented various mechanisms to facilitate VM provisioning; some of them have been discussed.

Table I. Comparison Of Some Resource Allocation Policies

Author/Paper	Policy	Parameters considered	Experimentation tool	Draw back
Chao-Tung Yang et. al[5]	Dynamic Resource Allocation	Load balancing	High Performance Computing Challenge	Only load balancing problem is addressed
Nguyen Quang-Hung et. al[6]	Scheduling	Energy	Simulation with parallel workloads	Except energy no other factors addressed
Weiwei Lina et. al[7]	Threshold based DRA	Cost, peak load	CloudSim	Works best for only peak loads
SivadonChaisiri et. al[8]	Optimal resource provisioning	Minimizing Total cost	Simulator	Cost reduction was the only goal
Sharrukh Zaman et. al [9]	Combinatorial Auction	Revenue (demand based)	Real workloads through simulation	Only on-demand based revenue
Jyotiska Nath et. al[10]	Tier-centric	Resource utilization at tier level	Amazon EC2 public cloud	Except utilization no other factors addressed
Shabeera et. al[11]	Optimized VM Allocation	Resource utilization, Performance	Simulations, Ant Colony Optimization	Only for data intensive applications and only performance was the main concentration
Pooja et. al[12]	Hybrid lease model	Resource utilization, load balancing, no starvation/rejection	Adaptive Contracting With Neighbor (ACWN) algorithm	Only resource utilization and load balancing factors addressed

A dynamic resource allocation strategy [5] has been proposed which would adjust the resource sharing to be balanced across the servers. An open nebula core handles all the scheduling decisions in a ranked manner to control the VM migrations by maintaining memory and cpu capacity on the servers. This approach concentrated more on resource utilization and response time and no consideration of energy and cost reductions.

The energy consumption based on busy time minimization of the PMs [6] is implemented using EMinTRE-LET algorithm. This algorithm worked efficiently at homogeneous PMs with parallel workloads from the archives. Through simulations they have proved that their approach could reduce the energy consumption of about 51.5% compared to the modified Power Aware Best Fit Decreasing and other algorithms.

The threshold based allocation policy assigns VM based on the minimum requirements [7] by the applications. Once the need increases, dynamically the allocation will be moved onto higher VMs with sufficient capacity. This method proved to be effective during peak loads to minimize the cost of allocations but no other factors were given importance.

An optimal cloud resource provisioning has been implemented [8] which consists of 3 provisioning stages to minimize the total cost of allocation. This included resource reservation and on-demand plans to allocate the resources to the user tasks. This strategy mainly looking at the total cost and no other specific measures for energy, response time and other factors are mentioned.

A combinatorial auction mechanism [9] for improving the total revenue has been presented by the authors wherein the users will be bidding for the requisite number of resources. A minimum price is reserved to be paid and the actual payment is according to the bidding. This strategy enabled dynamic provisioning with best VM combination to be allocated for the tasks. This approach does not consider any other factor than the cost benefits.

Tier-centric approach by the authors [10] enables resource allocation tier-wise rather than traditional approach with least monitoring. The tier centric approach helps to dynamically allocate the resources as per the need and also helps to use the resource pool for a longer duration. This approach concentrated more on reducing the operational and resource costs for VM allocation and no other factors are taken into account.

A VM allocation policy for data intensive applications is implemented [11] which would select a subset of available PMs for placement. This selection using Ant Colony Optimization would be such that the VMs demand must be fulfilled based on applications request. This also considers the data transfer performed between the nodes to be minimal for the executions to improve the performance.

A hybrid lease model [12] for improving dynamic load balancing and resource utilization followed a lease policy to accept input requests and utilized adaptive contracting with neighbors algorithm for dynamic load balancing. Using this approach the authors are guaranteed that resource starvation would improve with better load balancing. necessary to develop complex digital systems.

The comparison of the approaches by way of their methodologies, benefits and the drawbacks has been represented in table I.

III. Problem Scenario

Consider a cloud datacenter from the leading provider consisting of large number of heterogeneous servers. Whenever an application/job is submitted for the service/execution, the virtualized infrastructure creates & allocates a specific virtual machine for the request requirements. The problem here is, where to place the VM. i.e., on which server this should be placed. This placing is a complex task as it requires plenty of factors for consideration.

The factors to be taken care are RAM, CPU, Storage, Bandwidth as there source requirements. Apart from this some other external parameters relating to economies from the providers are energy consumption, resource utilization and QoS. Among these energy consumption and the QoS are the prime factors for providers' gain and consumers' reliance on them. In this paper, reduction in energy consumption and the faster response time factors are considered for the VM placement towards profit gain to the stake holders.

Consider a datacenter with n number of servers,

$$S = \{s_1, s_2, s_3, ..., s_n\}$$

Let m be the number of virtual machines,

$$V = \{v_1, v_2, v_3, ..., v_m\}$$

If t is the number of applications/tasks for execution,

$$C = \{c_1, c_2, c_3, ..., c_t\}$$

The main aim is to assign, all the tasks to the specific VMs,

$$\text{i.e., assign } \sum_{i=1}^{t} Ci \text{ to } \sum_{j=1}^{m} Vi \quad \forall\ V_i \in V \qquad (1)$$

Mapping the specific VMs to the servers.

$$\text{Map } \sum_{j=1}^{m} Vi \text{ to } \sum_{k=1}^{n} Sk \quad \forall\ S_i \in S \qquad (2)$$

The above assignment and mapping is based on the total resources available in the datacenter and the resource requests from the user. During runtime, if more resources are requested by the VM and the mapped server does not have enough resources, usually dynamic migration to other servers with more resource availability is carried out. Avoiding frequent migrations to improve the response time is defined using some resource-cap (50-80)% as an upper bound on resource usage of servers of data center by following vertical scaling techniques.

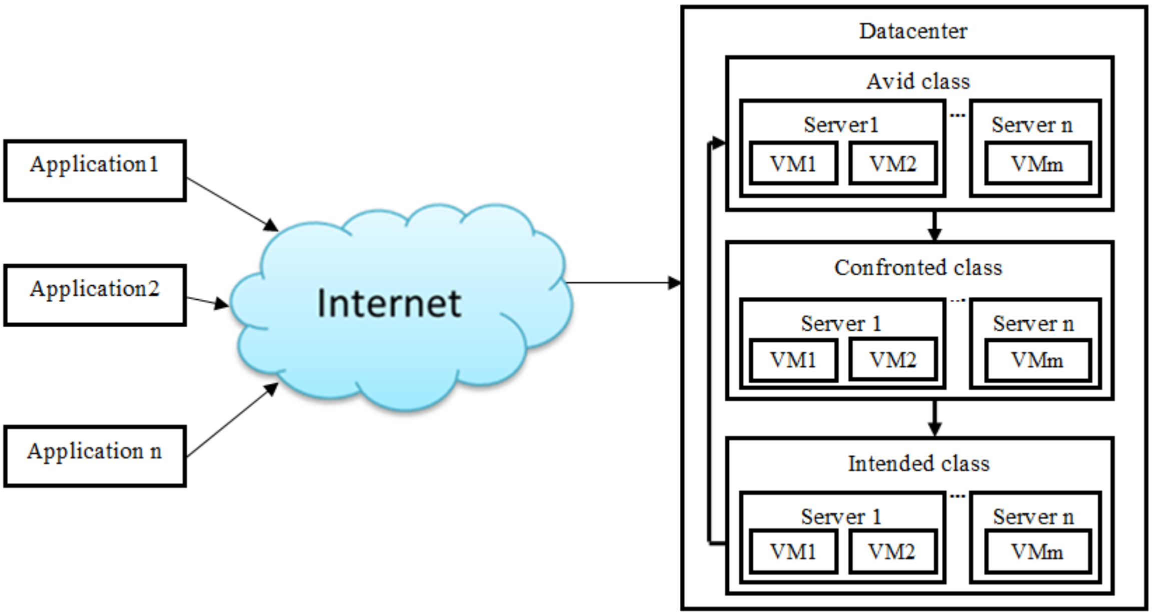

Fig.1. Basic System architecture

IV. System Architecture

The system contains a datacenter with a request to process multiple applications from the customers. The requests are processed at different levels based on the VMs requirements. The VMs are mapped to the servers for their needs using the levels as specified in fig. 1.

The levels are set mainly based on the availability of the RAM and the Processing capabilities in the servers. They are initially set at different levels as 50% in Avid class and 70% in the confronted class for mapping between servers and the VMs. Intended state is the exact match between the server and VMs capability. The resource capabilities of RAM and processer would be varied to identify the best suitable combination for different dataset scenarios and the varied no. of applications.

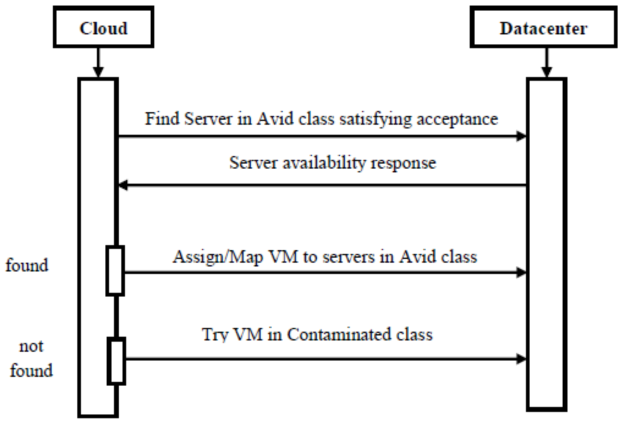

Fig.2. VM placement in servers from Avid class

Fig. 2 depicts the sequence diagram for verifying the servers availability to fulfill the needs of VMs to execute the given tasks in Avid class. If a match is found, servers are mapped to the corresponding VMs else it will be moved onto checking its suitability in the confronted class as shown in fig. 3.

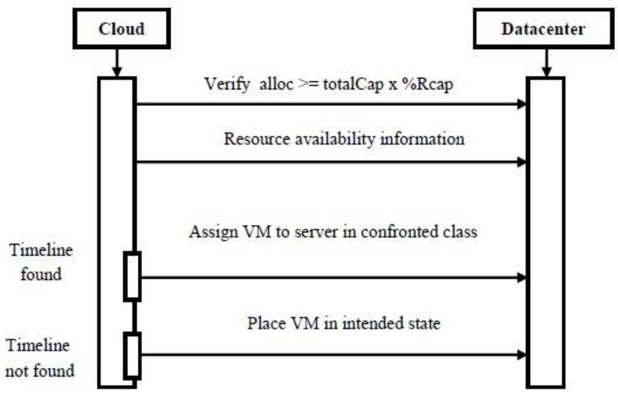

Fig.3. VM placement in servers from Confronted class

After verification and unable to identify the servers from the confronted class, the VMs will be checking in intended state wherein they would be selecting the servers exactly matching their requirements. If found, VM will be assigned, else, next VM in the queue will picked up for mapping and the same process is repeated as shown in fig. 4.

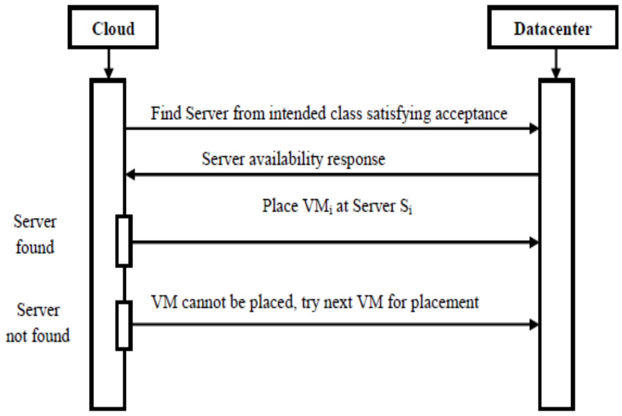

Fig.4. VM placement in servers from intended class

V. Implementation

Input: No. of Servers, VMs, RAM size, No. of Cores, Bandwidth, No. of tasks/applications

Output: Mapped VM's on different Servers

Determine the RAM and the CPU availability of the Servers:

$$A_{Vc} = Total_c - A_{lc} \quad (3)$$

$$A_{Vr} = Total_r - A_{lr} \quad (4)$$

Verify acceptability [13] for the Sever

$$(A_{lc}+Req_c)<=Total_c*Rcap_c \quad (5)$$

$$(A_{lr}+Req_r)<=Total_r*Rcap_r \quad (6)$$

Av_c, A_{lc}, Req_c are the available, allocated and the requested cores. A_{Vr}, A_{lr}, Req_r are the available, allocated and the requested memory respectively.

Place the VMs using either Avid or Confronted class of servers.

A. Procedure for VM Placement in Avid class

```
While (!VMplaced)
{
  Verify acceptability
  Find Server
  If (!Serverfound)
  {
    Try placing VM in Confronted
  }
  Else
  Allocate VM to the Server
}
```

B. Procedure for VM placement in confronted class

While (! Serverfound from Avid)
{
Try placing VM in confronted class servers
Perform acceptability test
If (server found)
{
Allocate the VM to Server
}
Else
Place the VM in intended class

If (No server satisfying VM req)
Select next VM
Repeat the process
}

C. Execution Process

While (task in hand)
{
Allocate suitable VM
Check for power consumption, response time
Reduce/Avoid No. of migrations
}
Power consumption by a server is proportional to the CPU utilization.

$$P(s) = P_{idle} + U*(P_{max} - P_{idle}) \quad (7)$$

Total power consumed:

$$P_T = \sum_{k=1}^{m} P(s) \quad (8)$$

Energy consumption is an integral of total power consumed [3] over a period of time, t.

$$E_T = \int_0^t P(s)\, dt \quad (9)$$

Total system maintenance cost also gets reduced once the power consumption gets reduced.

Server and VM configurations considered for the experimentations are:

Server configurations:
No. of Cores: 1 to 16
MIPS : 1000 - 10000
RAM: 4GB – 16GB
Storage : 16GB – 1 TB

The VM configurations:
No. of Cores: 1 to 16
MIPS : 1000 – 10000
RAM: 512MB – 4GB
Storage : 1GB – 8GB

Sample output of the system for no. of servers as 5, VMs as 10 and the tasks to be executed as 20 is shown in table II.

Table II. Sample Output Scenario

Cloudlet ID	STATUS	Datacenter ID	VM ID	Time (in sec)	Power (in Watts)	Start Time (in sec)	Finish Time (in sec)
6	SUCCESS	2	9	3.25	32.48	0.1	3.35
0	SUCCESS	2	4	4.83	24.17	0.1	4.93
3	SUCCESS	2	1	5.36	80.4	0.1	5.46
5	SUCCESS	2	4	8.11	40.55	0.1	8.21
15	SUCCESS	2	9	9.25	92.49	0.1	9.35
8	SUCCESS	2	1	9.36	140.39	0.1	9.46
1	SUCCESS	2	4	9.59	47.96	0.1	9.69
11	SUCCESS	2	4	9.92	49.62	0.1	10.02
7	SUCCESS	2	1	10.69	160.29	0.1	10.79
2	SUCCESS	2	1	11.02	165.29	0.1	11.12
18	SUCCESS	2	9	16.75	167.49	0.1	16.85
19	SUCCESS	2	9	20.75	20.5	0.1	20.85
17	SUCCESS	2	7	44	659.93	0.1	44.1
4	SUCCESS	2	0	56	112	0.1	56.1
10	SUCCESS	2	7	79.99	1199.88	0.1	80.09
14	SUCCESS	2	0	92	184	0.1	92.1
12	SUCCESS	2	0	108	216	0.1	108.1
9	SUCCESS	2	0	112	224	0.1	112.1
13	SUCCESS	2	7	114	1709.98	0.1	114.1
16	SUCCESS	2	7	116	1739.98	0.1	116.1

VI. Result and Analysis

The proposed strategy has been tested for varied number of hosts in the datacenter. The numbers of servers considered are 10, 50, 100, 200 and 500 respectively. For Avid class with 50% reserved RAM and 70% for confronted class.

The results shown in the fig. 5 depicts that the proposed approach has reduced power consumption compared to the first fit policy.

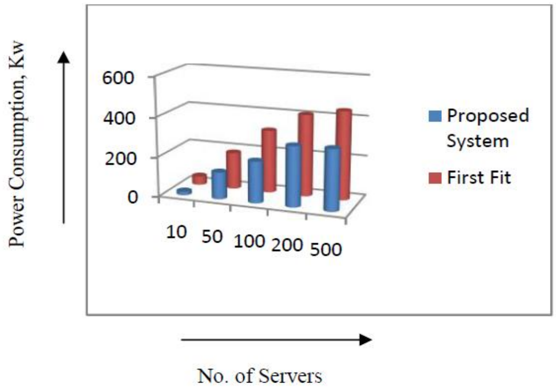

Fig.5. Power consumption Vs. No. of Servers

The response times for different scenarios of input and the datacenter configurations are as shown in fig. 6.

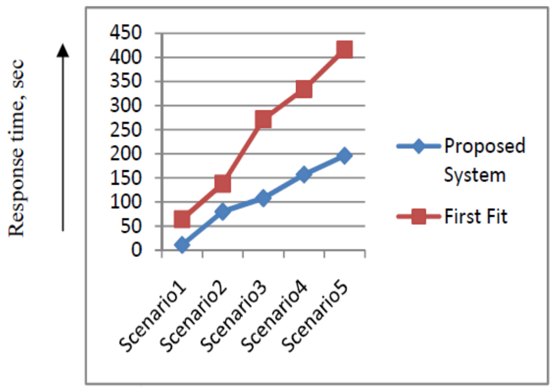

Fig.6. Response time Vs. Datacenter Scenarios

It is clear from the fig. 6 that the response time is significantly better in the proposed approach compared to the traditional first fit approach of initial VM placement.

For resources with 55% and 75% reserved levels the results are depicted in fig. 7 and fig. 8 respectively. It can be seen that power consumption is less compared to the first fit policy in fig. 7. The response time is also found to be lower compared to the existing method thereby displaying the benefits of the proposed method.

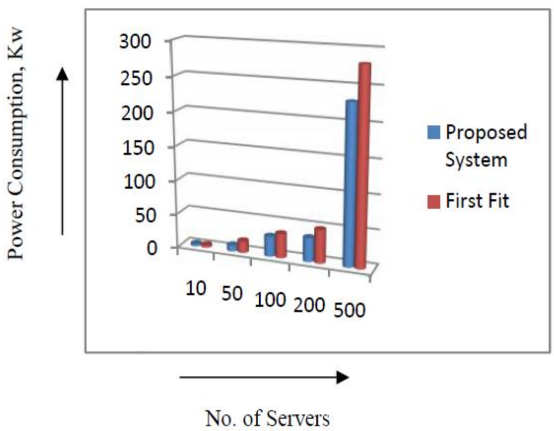

Fig.7. Power consumption Vs. No. of Servers

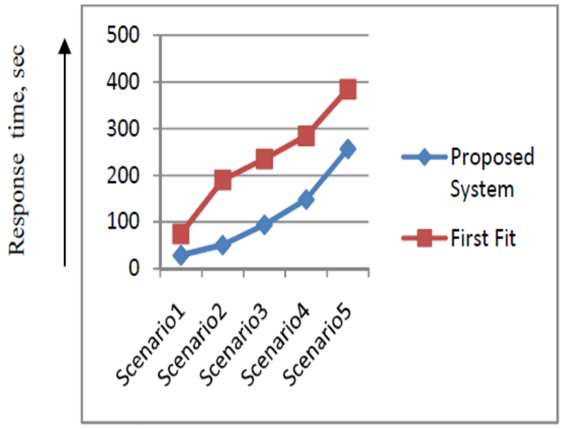

Fig.8. Response time Vs. Datacenter Scenarios

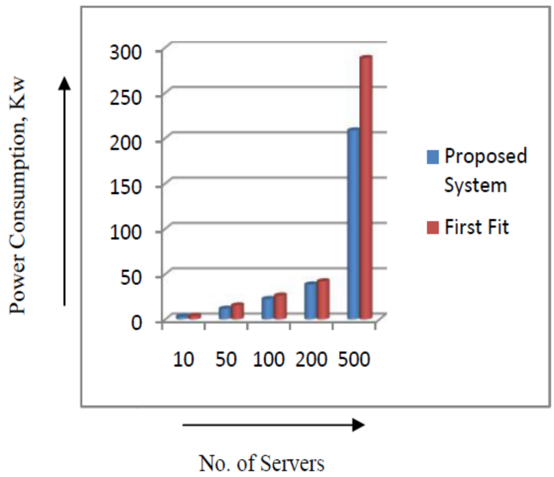

Fig.9. Power consumption Vs. No. of Servers

The power consumption with resource levels at 60% and 80% for the Avid and Confronted classes is shown in fig. 9. It can be analyzed that this approach also reduces energy consumption to a lower value.

Similarly fig. 10 shows the response times under different scenarios of input and datacenter clearly indicating that the proposed system is able to achieve better performance.

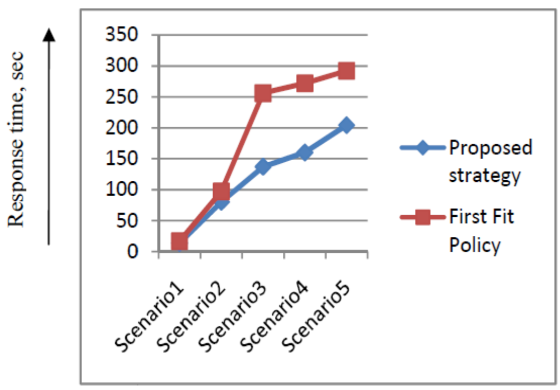

Fig.10. Response time Vs. Datacenter Scenarios

The average reduction in power consumption of the 3 categories considered with respect to the traditional first fit policy is shown in the fig. 11.

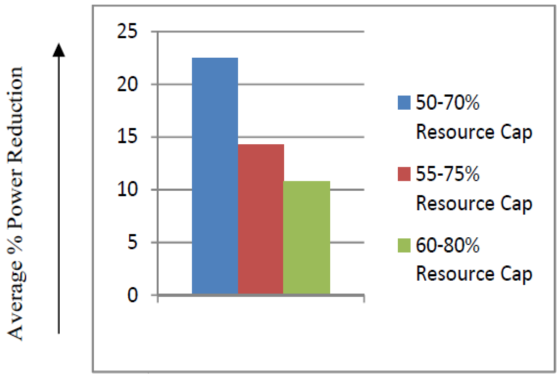

Fig.11. Average % Power Reduction vs. Resource Cap

The improvements in the response time i.e, reduction in time taken to respond & execute the applications of 3 categories of resource cap with respect to the first fit policy is shown in fig. 12.

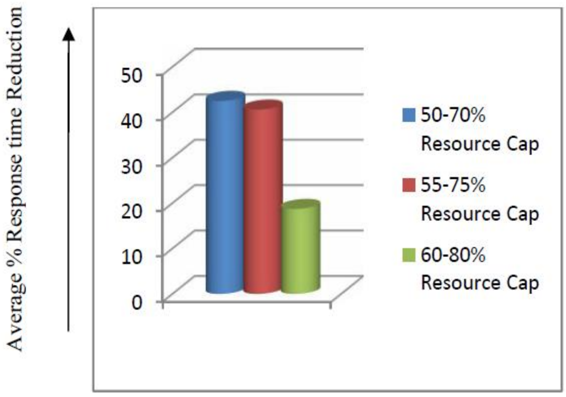

Fig.12 Average % Response time Reduction Vs. Resource Cap

Based on the results of the experimentations carried out for different ranges of RAM and processors at various classes of servers, it has been verified that all the scenarios of measurements were able to provide better results than the traditional first fit policy. But, the resource capacity of 50%- 70%provided best results. It provided an overall average power reduction of 22.53% and 42.5% faster response time while executing the given applications.

VII. Conclusion and Future Scope

Among the factors of consideration in cloud computing for better benefits & efficiency, energy consumption and the response time are of higher priority for the providers. This paper proposed a mechanism for VMs initial placement to the server that would consider minimizing no. of VM migrations to reduce both the power consumption as well as the response time. Based on the experimentation for heterogeneous number of servers and the tasks it has been clear that this technique of vertical scaling of servers capacity to map the VMs to the servers will result in better performance and reduced energy consumption. Experimentations have been conducted with varied resource capacities from 50 to 80% as thresholds to identify the better combination. Although all the variations provided better results than the existing policy, the resource capacity of 50% and 70% reserved capacities for Avid class and confronted class provided slightly better benefits. It has been proved that this initial VM placement method provides significant better performance as well as consumes less power. Since the energy consumption is proportional to the amount of power consumed over a period of time, energy consumed also becomes minimal.

The future scope could be using other parameters like bandwidth and disk capacity to set up the levels for VM mapping to the servers for initial placement. It could also be tested with real cloud set up to better understand the efficiency.

References

[1] Ankita Chaudary, Shilpa Ranab and K.J. Matahai**,** "A Critical Analysis of Energy Efficient Virtual Machine Placement Techniques and its Optimization in a Cloud Computing Environment", *Procedia Computer Science*, Volume 78, pp.132-138, 2015.

[2] Jiang Tao Zhang, Hejiao Huang and Xuan Wang, " Resource provision algorithms in cloud computing: A survey", *Journal of network and computer applications*, Volume 64, pp.23-42, 2016.

[3] Anton Beloglazov, Jemal Abawajy and Rajkumar Buyya, "Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing", *Future Generation Computer Systems*, 2011.

[4] Jasnil Bodele, Anil Sarje, "Dynamic Load Balancing With Cost & Energy Optimization in Cloud Computing", *IJERT* Vol. 2 Issue 4, ISSN: 2278-0181, 2013.

[5] Chao-Tung Yang, Hsiang-Yao Cheng, and Kuan-Lung Huang, "A dynamic resource allocation model for virtual machine management on cloud", *Springer-Verlag Berlin Heidelberg*, *CCIS* 261, pp. 581–590, 2011

[6] Nguyen Quang-Hung and Nam Thoai, "Minimizing Total Busy Time with Application to Energy-efficient Scheduling of Virtual Machines," *International Conference on Advanced Computing and Applications* pp. 141-148, 2016

[7] Weiwei Lina, James Z. Wangb, Chen Liangc, Deyu Qi, "A threshold-based dynamic resource allocation scheme for cloud computing", *Elsevier Procedia Engineering*, 23, pp. 695 – 703, 2011

[8] SivadonChaisiri, Bu-Sung Lee and DusitNiyato, "Optimization of resource provisioning cost in cloud computing", *IEEE transactions on services computing*, Vol. 5, No. 2, 2012

[9] Sharrukh Zaman and Daniel Grosu, "A combinatorial auction based mechanism for dynamic VM provisioning and allocation in clouds", *IEEE Transactions on Cloud Computing*, Vol. 1, No. 2, 2013

[10] Jyotiska Nath Khasnabish, Mohammad Firoj Mithaniand, Shrisha Rao, "Tier-Centric resource allocation in multi-tier cloud systems", *IEEE Transactions on Cloud Computing, in press*, DOI: 10.1109/TCC.2015.2424888, 2015

[11] T.P. Shabeera, S.D. Madhu Kumar, Sameera M. Salam and K. Murali Krishnan, “Optimizing VM allocation and data placement for data-intensive applications in cloud using ACO meta –heuristic algorithm,” *Engineering Science and Technology, an International Journal Elsevier*, Vol. 20, pp.616–628, 2017.

[12] Pooja S Kshirasagar and Anita M Pujar, “Resource Allocation Strategy with Lease Policy and Dynamic Load Balancing”, *International Journal of Modern Education and Computer Science, MECS Publishers*, 2, pp. 27-33, 2017. DOI: 10.5815/ijmecs.2017.02.03

[13] Madhukar Shelar, Shirish Sane, Vilas Kharat and Rushikesh Jadhav, “Efficient Virtual machine Placement with Energy Savings in Cloud datacenter”, *International Journal of Cloud-Computing and Super-Computing* Vol.1, No.1, pp.15-26, 2014 http://dx.doi.org/10.14257/ijcs.2014.1.1.02

[14] Shreenath Acharya and Demian Antony D’Mello, “Energy and Cost Efficient Dynamic Load Balancing Algorithm for Resource Provisioning in Cloud”, *International Journal of Applied Engineering Research (IJAER)*, Vol. 12, No. 24, 2017.

Authors’ Profiles

Shreenath Acharya received B.E from Mysore University and M.Tech from VTU. Currently serving in Computer Science & Engineering Department at St Joseph Engineering College, Mangaluru. He has over 19 years of experience in education sector and more than 25 publications in international conferences/journals. His areas of interest are cloud computing, big data, computer communication networks and security.

Dr. Demian Antony D’Mello is currently serving as the Professor & Head of Computer Science & Engineering Department at Canara Engg. College, Mangaluru. He received B.E in Computer Engineering from Mangalore University and M.Tech & Ph.D from NITK Surathkal. He has over 18 years of experience in education sector and over 45 publications in reputed international conferences/journals. He is a regular reviewer for many reputed journals and conferences. He also serves as an editorial board member for reputed journals. His areas of interest are web services, cloud computing, internet technologies and digital image processing.

How to cite this paper: Shreenath Acharya, Demian Antony D’Mello, "Energy Saving VM Placement in Cloud", International Journal of Modern Education and Computer Science(IJMECS), Vol.10, No.12, pp. 28-35, 2018.DOI: 10.5815/ijmecs.2018.12.04

