International Journal of Engineering &Technology, 7 (1.7) (2018) 21-26

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

RHEA: resource hypervisor and efficient allocator in cloud

G Soniya Priyatharsini [1]*, N Malarvizhi [2]

[1] Research Scholar, Department of Computer Science and Engineering, School of Computing, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Avadi, Chennai-62, TamilNadu, India
[2] Professor, Department of Computer Science and Engineering, School of Computing, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Avadi, Chennai-62, TamilNadu, India
**Corresponding author E-mail: sonigeorge85@gmail.com*

Abstract

In this modern world, people are not ready to waste their time in waiting for long duration. That's why cloud computing is such an enormous number of fans that it can be rented and also pay per use. The cloud service provider is concern about the data owner's satisfaction in cloud usage. The main area they concentrate will be the security of the owner's data and the resource allocation as per the request. This paper explains how the resources are efficiently allocated and scheduled to the clients. It follows four steps; firstly it identifies the active PMs. Next it defragments the identified machines. Then it balances the load along with the threshold feature to enhance the usage of the resource utilization. Finally it allocates the efficient Virtual Machines (VM) to the data owner as per the request. This is done using cloudsim along with java.

Keywords: *Cloud Resource Management; Virtualization Hypervisor; Allocation of Resources in Cloud.*

1. Introduction

Cloud is the widely used technology in the Information Technology world. It reduces the human working power. It also reduces the storage capacities of the computer hardware. The cloud computing environment can be explained by a figure 1.

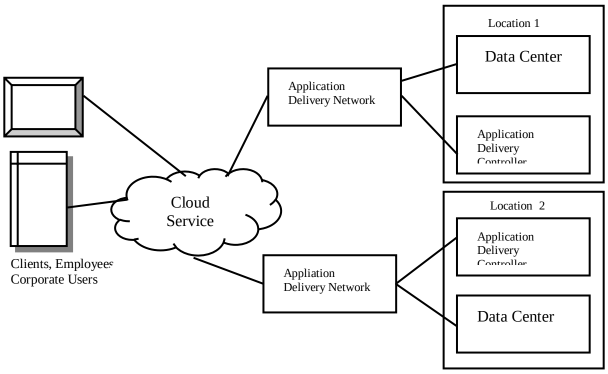
Location 1
Data Center
Application Delivery Network
Application Delivery Controller
Cloud Service
Clients, Employees Corporate Users
Location 2
Application Delivery Controller
Appliation Delivery Network
Data Center

Fig. 1: Cloud Computing Environment.

Inspite of the tremendous growth in the cloud computing, the clients are increasing in the cloud [1].

There are many features in the cloud. Some of them are:

1) Pay as we go- The client can pay for what he/she used.
2) Interface- The cloud computing environment is the user friendly interface.
3) Workspace- The working environment in cloud is the client requested and customized servicing area.
4) Virtualization- This techniques reduces the cooling consumption, usage of hardware etc.

The growth poses a huge threat of security and the resource provisioning problems to the service provider. So to use the resources efficiently, a resource manager should be there for the proper management of the resources. The allocation of the resources is done with the respect of some parameters such as availability, performance, response time, throughput, security etc.

1.1. Resource management

Research management involves provisioning of the resources [2], resource allocation and resource monitoring.

i) Resource provisioning: - It gives the search result for the adequate resources types which are available and are match the application requirements.
ii) Resource scheduling: - The procedure of assigning of available resources to the requested clients.
iii) Resource Monitoring: - It controls and also manages the software and the hardware infrastructure of the cloud.

1.2. Virtualization

It is the process where one system which has the ability of appearing likes the many systems. [3] It is used to improve the performance parameters by using the physical resources as a pool. From here the virtual resources can be allocated. Multiple virtual systems can run on single Physical Machine (PM).

The Virtual Machine architecture is shown in figure 2:

Applications	Applications	Applications
VM	VM	VM
Guest OS	Guest OS	Guest OS
Virtualization Platform (VMware, Xen…)		
Physical Hardware		

Fig. 2: Virtual Machine Architecture.

A hypervisor or a virtualization manager is a program that allows multiple operating systems to share single hardware host. Every

Copyright © 2018 G Soniya Priyatharsini, N Malarvizhi. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

guest operating system appears to have the host's processor, memory, and other resources all to itself. However the hypervisor is actually controlling the host processor and resources, allocating what is needed to each operating system in turn and making sure that the guest operating systems cannot disrupt each other.

Virtualization concept is used to share the resources. It helps in the cost reduction. This machines are isolated if this machines are physically separated. These also encapsulate a complete computing environment. They run independently of underlying hardware. [4] They can be migrated between hosts.

The main components of the resource management systems are as follows:

a) Inputs: - The policies of admission control and the offered workload, the load balancing, the energy optimization and the QoS.
b) Components of the system: - To estimate the relevant measures of the performance and controller, the sensors are used.
c) Outputs-: The resource allocation to the individual applications.

The rest of the paper is organized as follows: Section 2 explains the previous paper related to our work. In Section 3, proposed system of RHEA is explained in detail and describes how the resources are segregated for the scheduling process. In Section 4 the process of the allocation and scheduling of the resources are explained. Section 5 explains and analyzes the simulation results of the proposed method RHEA. Finally the conclusion of the work is explained in Section 6

2. Literature review

Nowadays the requirements of the cloud computing users are high and emerging widely. Thus to meet their requirements the cloud providers are trying to do their work efficient with quality of service. Some of the papers are discussed here on the basic of the efficient resource management.

Mohammed et al. [5] discussed that the multi tier cloud system's resource allocation. It addresses the problem of over provisioning in the enterprise class application. It proposed a multi tier cloud provisioning for reducing resources over provisioning. T-BICA is used to progress the process of the cloud managed resource allocation process .Also it prevents the over provisioning of the process. It is cost analysis and the Tier centric business impact. The leveraging of the existing layer is introduced.

Hong Xu et al.[6] has proposed an framework for a versatile and also an efficient resource management, Anchor. Anchor is architecture of the general resource management. It separates policies from mechanism using the stable matching framework. This is done while mapping VMs and physical servers. In this Anchor, a variety of different resource management policies are able to express by the operators and the clients. The important factor about the Anchor is the many to one. This efficiently matches the heterogeneous resource needs with the VMs using online and offline algorithms.

Sheng Di et al. [7] proposed the payment minimization and the error tolerant allocation. The cloud systems which are in the cloud can be clubbed. Then it can be allocated on demand. The contributions of this paper are: - It validates the effectiveness of the VM facilitated cluster environment. The polynomial time with VM technology is proposed based on the cloud environment. It can be reduced the cost of allocation.

Ping et al. [8] describes about the model where the resources are managed in hierarchical. This paper resolves the long waiting time and the heavy load in the centralized resource management. Resource tables used to clear this. It gives a resource allocation process with a model of resource management system.

Mayank Mishra et al. [9] proposed resource management which is dynamic along with the VM migrations. The efficiency of the resource management is increased by the VM relates features such as isolation, flexibility, migration of the machine state. The work explains about the usage and the process of the relocation of the VMs. This is towards resource management dynamically in the environment which is virtualized.

Abirami et al. [10] suggests that the response time and the waiting time is forced in the allocation of resources. This work concentrates on the new method called linear scheduling of resources and tasks. With the help of the available VMs and tasks, the scheduling algorithm is designed. By using this resource utilization is increased.

Clark et al. [11] explains the VMs which is lively migrated. It also explains the software and the hardware separation which provides the fault management. The running devices may have different traffic. This can be controlled by the algorithm which is mentioned here is rate adaptive algorithm. Migration overhead, page dirtied rate and the bandwidth are the factors affecting the total migration.

Shikharesh [12] describes the match making in his paper. This paper says that match making is the important scenario than then was the scheduling. It is because the resources are arranged first according to the user's request. It is then designed from the obtained resource pool. It also explains that matchmaking before scheduling makes some uncertainties.

Nilabja et al. [13] explains the pay per use scenario. The optimization of the system is identified here. The system behavior is concerned on the basis of the performance and usage of the resources. The cost of these can be varied basis on different factors like cost of configuration, cost of leasing resources etc.

Soramichi et al [14] illustrates the Miyakodori algorithm for the memory reusing technique. This technique decreases the data amount transferring in relocation lively. When it executes the VMs may migrates. This leads the memory image to be reused. Thus by using this the migration time can be reduced simultaneously with greater optimizations algorithms.

3. Proposed system

In cloud computing there are enormous area for the scope of improvement. The cloud provider and the cloud user's relation must be like a win win policy. No one should lose from their perspective. While supplying the resources the cloud provider should not have lack of resources also cloud user should not found that he/she is spending more. So this can be done only by using the resource management techniques. There are many resource management policies are available in cloud. Here this work concentrates on the power consumption and cost. This can be achieved by using the following objectives.

1) Identifying the active PMs.
2) Defragmenting the identified VMs in PMs.
3) Resource Management using threshold value.
4) Efficient allocation of the cloud resources

3.1. Identifying the active PMs

The PMs which are active are recognized with the help of the binary cuckoo search algorithm. It gives the accurate results whether the PMs are active or not. This algorithm is already compared with the other optimized algorithms such as ant colony algorithm [15], bee colony algorithm [16], wolf theory [17], and particle swan optimization [18]. This yields the result as that this cuckoo search algorithm is comparatively give more adequate results. The pseudo code for finding the active PMs using binary cuckoo search algorithm is given here:

```
Input:Entirehostlist,Output: hostlist
leftOverVMs ← 0
for host ∈hostlist do
    for virtualmachine ∈host.VMList do
        if possible to reallocate virtualmachine in host then
            Reallocate resources for virtualmachine in host
        else add virtualmachine to leftOverVMs list
```

```
end if
end for
end for
for virtualmachine ∈leftOverVMs do
Place virtualmachine in a PM in first fit manner
end for
```

From the above pseudocode it is clearly known that the active PMs can be identified whether the PM is active or not. This is because the maximum utilized servers can be arranged in the decreasing order. This method gives the user to view the underutilized server in the last. So with respect to that of any conditions the underutilized server is removed.

3.2. Defragmenting the identified VMs in PMs

The defragmentation of the identified PMs can be carried out with the help of following Figures. Figure 3(a) shows the resources without the residual fragmentation. Figure 3(b) shows the PMs with the residual resource fragmentation.

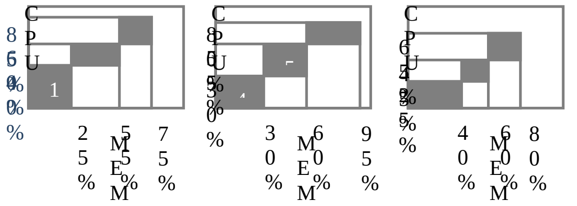

Fig. 3 (A): Resource Allocation with More Residual Resource Fragmentation.

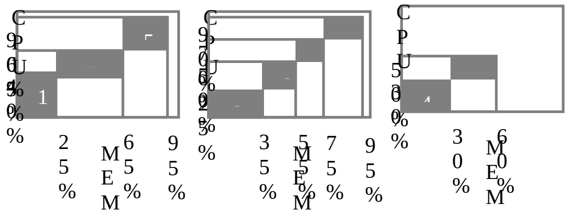

Fig. 3 (B): Resource Allocation with More Residual Resource Fragmentation.

The values can be derived with the help of the following formula.

$$\frac{\sum_{i=1}^{m}(T_j - C_{ij})2}{\sum_{i=1}^{m}(T_j - C_{ij})}$$

Defragment=

Here,
m – Total number of active PMs
T_j – Threshold of the j-th source
C_{ij}– Consumption of the jth resource in ith PM.
Thus the lesser is the resource fragmentation, higher the value of the defragment. From this defragmented PMs the inactive machines can be set to sleep mode and then the active PMs can be rearranges for the next process.

3.3. Balancing the load in the VMs

A system should have the upper and the lower threshold limit. It can be said as overloaded, if the limit is beyond the upper limit and can said to be underloaded, if it is said to be below the lower limit. So by using the threshold limits, it is important for a system to use the proper allocation techniques. This paper concentrates on the upper and the lower threshold limits of the efficient allocation. It is also important to minimize the migration issues. Minimized number of migration gives more efficient results and that can be explained with the following steps:

a) Calculate the load on the PMs and VMs
b) Calculate the lower and the upper threshold values
c) Select the host to be migrated and the PMs to accept the migrated VM.

A) Calculation of Load

The load on the PMs can be found by using the summation of the all VMs load.

$$VM_{load}^{m} = \frac{\sum_{i=1}^{n} VM_{load}^{i}}{n}$$

Here the number of VM in mth host.
In this paper the parameters such as CPU utilization, Bandwidth utilization and the memory utilization are considered for each VMs. For the ith VM, the above said parameters can be calculated as follows:

$$VM_{t}^{cpu} = \frac{\text{Total capacity used by the host}}{\text{Total capacity of the host}}$$

$$VM_{t}^{bw} = \frac{\text{Total bandwidth used by the host}}{\text{Total bw capacity of the host}}$$

$$VM_{t}^{ram} = \frac{\text{Total ram used by the host}}{\text{Total ram capacity of the host}}$$

Then the total load on the ith vm can be calculated as follows:

$$VM_{load}^{i} = VM_{t}^{cpu}$$

B) Calculation of Upper and Lower threshold

It is observed that when the threshold value increases it reduces the system performance. Thus it is concerned that the value of the threshold is not much increased.

$$Thres=\sum_{i=1}^{n} 1 \frac{VM_{i}^{cpu}+VM_{i}^{ram}+VM_{i}^{bw}}{n}$$

T_{upper}=1-x*Thres

T_{lower}=0.3

There are many of the VMs are running in the PMs. So it will be very difficult to choose the appropriate VMs for migration. If the inadequate VM is chosen, then it raises many migration problems. So it is important to choose the correct VM. To do the work, this paper takes the help of the Enhanced Cloud Resource Consolidating (ECRC) algorithm for the better performance.
Pseudocode for the ECRC Algorithm is given below:

```
candHosts ← Ø, dispValues ← Ø
for host ∈ PMList do
dispValues[host] ←getDispValue(host)
end for
candHosts ← getCandHosts(PMList, dispValues)
meanUtil ← findMeanUtil(candHosts)
Sort candHosts based on meanUtil
Assign a category to each candHost
for candHost ∈candHosts do
Assign a category to each vm in candHost
Order vms in candHost in decreasing order of suitability of migrating out
for orderedVm ∈ OrderedHostVMs do
Find set of vms in candHost belong to a category in the opposite order of suitability
if orderedVm and the set of vms can be exchanged with decrease in resource fragmentation then
Schedule allocations of orderedVm and the set of vms
end if
end for
```

end for

Input: PMList, VMList, AlloSched Output: AlloSched
This gives the PMs ready for the migration list. Next the selected VMs have to be allocated efficiently.

4. Resource hypervising and efficient allocation

The workflow of the proposed VM allocation and scheduling method is given below:

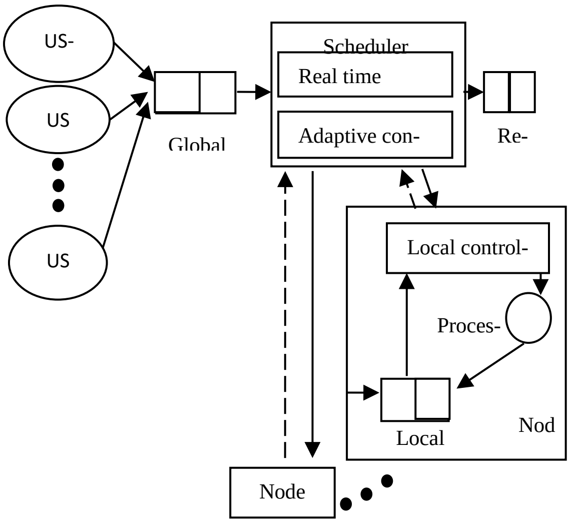

Fig. 4: Resource Hypervisor and Efficient Allocator.

Here the filtered VMs are given as the input. Thus the scheduling takes place with the help of the following instructions.

1) Selected VMs will be arranged in the queue on the basis of the fitness value
2) The arrival of the VMs can be explicit on the basis of the length of the queue.
3) Every VM is independent. It executes each one at a time.
4) When any VMs fail, then the VM waiting in the second independently do on one another.

The tasks are stored in the TQ (Task Queue). For a proper scheduling technique it is believed that the waiting time and the processing time will matters. Thus this work concentrates on this area for the better result.

Waiting time $W_i = \frac{R}{1-\rho}$, i=1, 2, 3…n

Here R is the Residual service time and ρ is the system utilization. Where,

$R = \sum_{i=1}^{n} \frac{\rho i}{\mu i}$, i= 1, 2, 3,…n

$\rho = \frac{\gamma i}{s * \mu i}$, i= 1, 2, 3…n

s- The network node's number.
This shows the waiting process of the task. The next vision is to process the waiting tasks with the efficient manner. i.e total time consumed for the process of j^th task. It also can be done with the following steps.

$T_i = P_{ij} + W_i$

P_{ij} gives the time consumed by the node j to process task i.

Makespan = maxT_i

Flowtime = $\sum_{i=1}^{n}$ Ti

Here both the makespan and the flow time should be minimized. Thus it can be expressed as:

F=min [a x makespan + b x flowtime]

Here both a,b are weight coefficients. It should be greater than or equal to zero and also a along with b gives 1 as output. Both are equally important that's why 0.5 is given for both a and b.
Procedure for the resource allocator.
Initialize the position of all the nodes and the parameters.
While iteration t is less than the maximum iterations do:
Calculate the next position of the node
Calculate the fitness value of the every candidate.
End while
Select the optimal solution and return
After completing the process work it gives the better result for the VM placement. Thus by comparing with the existing works it gives the better result in scheduling the VMs.

5. Experimental results

In this section, the results are expressed for various parameters. By using the method RHEA, the performance of the resources changed. The following Figure 5 illustrates the chart comparison of energy consumption among the proposed system with the existing two algorithms. Those are Optimized Cloud Resource Provisioning and Optimal Virtual Machine Placement. The workloads are given in GB as inputs. The input datas are given in the Table 1. Figure illustrates that the output received is slightly (0.009% to 0.05%) differing from the existing techniques. It is reducing much power consumption. So it can be concluded that this technique is useful in the energy consumption process.

i) Energy consumption

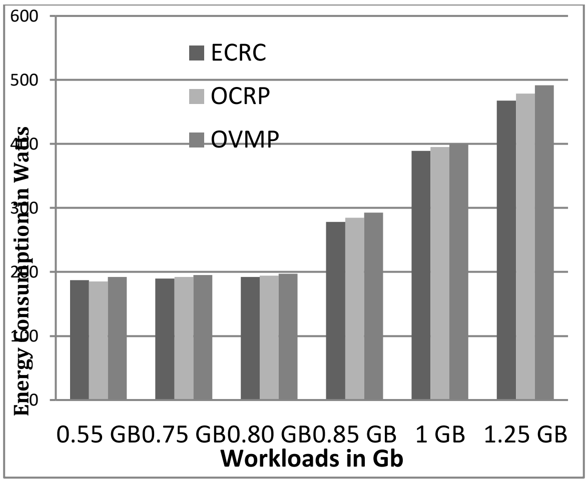

Fig. 5: Consumption of Energy Using Various Algorithms.

ii) Defragmentation of data

The following Figure 6 explains the data fragmentation of the identified physical servers. It compares with the already existing algorithms like OVMP and OCRP for server consolidation. The below listed Table 2 explains the values for the data fragmentation. In each consolidation interval, the defragmentation of the residual resources is comparatively high for ECRC technique. So from the experimental study it can be concluded that the proposed ECRC technique provides better result while comparing with the existing algorithms. The retrieved result can be gained only because of the defragment value. Whenever the value of the defragment is higher, then the defragment is lesser.

Table 1: Energy Consumption Datasets

Workload in size	ECRC	OCRP	OVMP	ECRC % on OCRP	ECRC % on OVMP
0.55 GB	187.2	185.4	192.3	0.009615	0.02724
0.75 GB	189.7	192.3	195	0.01371	0.02794
0.80 GB	192.1	194.3	197.4	0.01145	0.02759
0.85 GB	278.2	284.7	292.5	0.02336	0.0514
1 GB	389.4	395.2	409.4	0.01489	0.05136
1.25 GB	467.9	478.8	491.9	0.0233	0.05129

Table 2: Defragmentation Datasets

Consolidated Interval	OCRP	OVMP	ECRC
1800	64	71	65
5400	31	28	32
9000	30	24	30
12600	21	20	22
16200	26	25	28
19800	17	15	18
23400	15	10	16
27000	11	12	15
30600	13	14	15
34200	14	10	15
41400	16	12	16
45000	17	14	17
48600	17	13	18
52200	18	18	18
55800	11	10	11
59400	20	20	21
63000	30	30	32
66600	15	12	19
70200	21	18	23
73800	14	10	15
77400	14	11	15
81000	14	10	15
84600	14	11	14

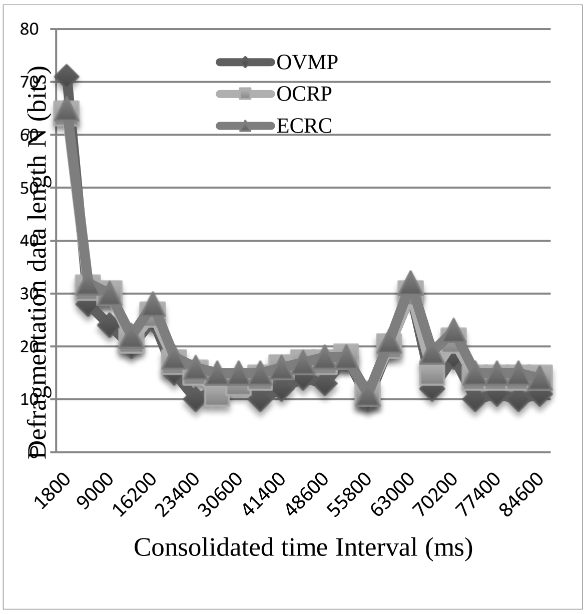

Fig. 6: At Different Consolidation Intervals Residual Resource Defragmentation.

iii) Cost

By using the ECRC technique the cost can be fairly reduced. It is because only the active machines are in the working stage and all the others are in the sleep mode. Thus by reducing the power consumption the cost for the expenditure of power can be reduced automatically. It also reduces the man power too. The below shown table listed the comparison between the FOG and OCRP algorithm with the proposed system. The numerals given here are in dollars.

Table 3: Performance Comparision with Respect to Cost

Cost/no of tasks	FOG	ORCP	ECRC
10	3.6348	3.672	3.5229
20	6.2052	6.411	5.8303
30	11.4159	12.0167	10.6139
40	19.9727	20.522	18.7145
50	25.5633	26.7745	24.04

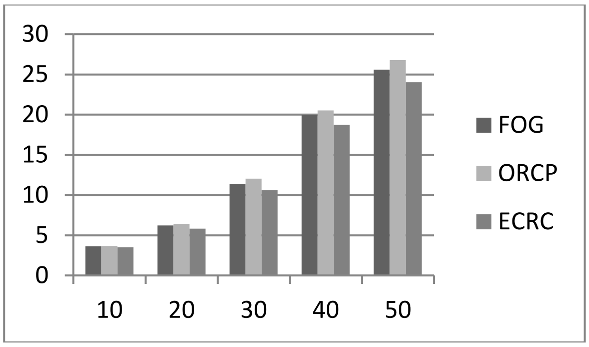

Fig. 4: Comparison of Existing Algorithms with ECRC.

The cost expenditure is minimized by using this technique. When comparing with existing systems the proposed system gives 0.15 % to 0.43% of reduction of the cost expenditure. So it can be used in the process for minimizing the cost expenditure.

iv) Data Allocation and scheduling

To reach the convergence state, the virtual machines are allocated with respect to the different algorithms. Initially 10 virtual machines are allocated using the FOG technique and the ECRC technique. It proves that it will be change for different techniques. Likewise here the experiment is done for various allocations of virtual machines. That is from 10 to 100 virtual machines. This allocation explains the convergence state for the virtual machines.

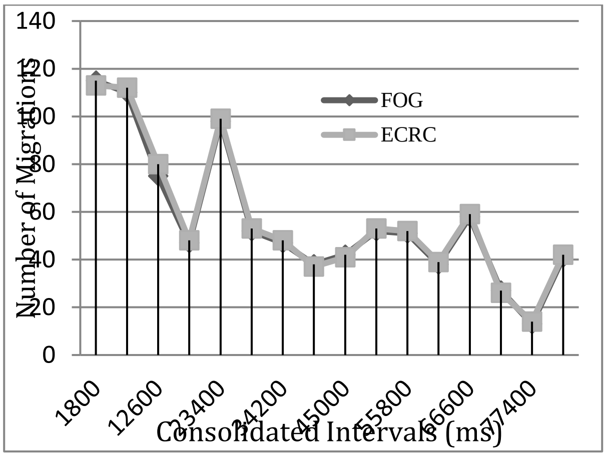

Fig. 5: Comparison with FOG for Allocation with ECRC.

Figure 5 illustrates the allocation process for the proposed work. This gives better result while compared with the existing works.

6. Conclusion

This paper explains a model RHEA. A Resource Hypervisor and Efficient Allocator of the resource. This model efficiently identifies the active PMs and from those machines the load is balanced with the help of the threshold and cost as parameters. These parameters give a result in the load balancing of the resources in the PM and also in the VM. Finally by using these parameters the scheduling is calculated by weighing time and the processing time. Here the flow time is minimized to achieve good results. Thus the resources are allocated and scheduled efficiently.

References

[1] Puneet Himthani,” Efficient Technique for Allocation of Processing Elements to Virtual Machines in Cloud Environment”, International Journal of Computer Science and Network Security (IJCSNS), VOL.16 No.8, August 2017.

[2] R.Madhumathi and R.Radha Krishnan,” Priority queue scheduling approach for resource allocation in cloud”, Asian journal of Information technology, 15(3):472-480, ISSN1682, 2017.

[3] Nitishchandra Vyas, Prof. Amit Chauhan,” A survey on virtual machine migration techniques in cloud computing”, Innovation in Engineering & Management (IJAIEM) or International journal of application, Issue 5, Volume 5, May 2016.

[4] G. Naga Srikanth, Dr. G. Naga Satish, R. Krishnam Raju Indukuri, Dr. P. Suresh Varma,” A novel Scheduling Model for resource allocation in cloud computing”, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 6, Issue 7,July 2016.

[5] Mohammad firoj mithani, Shrisha rao,” Improving resource allocation in multi tier cloud systems”, IEEE International conference on systems conference, 19-22 March 2012, pp 1-6.

[6] Hong Xu, Baochun Li, “Anchor: An versatile and efficient framework for resource management in the clouds”, IEEE Transcations on parallel and distributed systems, Vol 24,No 6, June 2013, pp 1066-1076.

[7] Sheng Di, Cho-Li Wang, “Error –Tolerant resource allocation and payment minimization for cloud system”, IEEE Transactions on parallel and distributed systems, Vol .24, No 6, June 2013, pp 1097-1106.

[8] Ping Guo, Ling-ling Bu,” The Hierarchical resource management modal based on cloud computing”, IEEE symposium on Electrical and Electronics Engineering(EEESYM),24-27, June 2012, pp 471-474.

[9] Mayank Mishra, Anwesha Das, Purushottam Kulkarni, Anirudha Sahoo, “Dynamic resource management using virtual machine migrations”, IEEE Communications Magazine, Vol 50, Issue 9, September 2012. https://doi.org/10.1109/MCOM.2012.6295709.

[10] Abirami S.P, Shalini Ramanathan” Linear scheduling strategy for resource allocation in cloud environment”, International journal on cloud computing and architecture, vol 2, No 1, February.

[11] Christopher Clark, Keir Fraser, Steven hand, Jacob Gorm Hanseny, Eric July, Christian Limpach, Ian Pratt, Andrew warfield,” Live migration on virtual machines”, 2nd Symposium on Networked systems design and implementation (NSDI), May 2005.

[12] Shikharesh Mujumdar, “Resource management on cloud: Handling uncertainities in parameters and policies”, CSI Communications, edi pp 16-19.2011.

[13] Nilabja Roy, Abhisheik Dubey and Aniruddha Gokhale, “Efficient autoscaling in the cloud using predictive models for workload forecasting”, Volume 3, January 2012,

[14] Soramichi Akiyama, Takahiro Hirofuchi, Ryoushi Takano, Shinichi Honiden, ‘Miyakodori: A memory reusing mechanism for dynamic VM consolidation” Fifth International conference on cloud computing, IEEE 2012.

[15] Visu, P., S. Koteeswaran and J. Janet, Artificial bee colony based energy aware and energy efficient routing protocol. J. Comput. Sci., 8(2): 227-231.2012.

[16] D.T. Pham, A. Ghanbarzadeh, E. Koç, S. Otri , S. Rahim , M. Zaidi,” The Bees Algorithm – A Novel Tool for Complex Optimisation Problems” Manufacturing Engineering Centre, Cardiff University, UK, 2005.

[17] Shahla Shoghian, Maryam Kouzehgar,” A Comparison among Wolf Pack Search and Four other Optimization Algorithms”, International Scholarly and Scientific Research & Innovation, World Academy of Science, Engineering and Technology, Vol: 6, 2012.

[18] Zhang Yu1 and Xiaomei Yang,” Full Glowworm Swarm Optimization Algorithm for Whole-Set Orders Scheduling in Single Machine”, The Scientific World Journal, Volume 2013, Article ID 652061.