IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-6, Issue-9, Sept. 2018 E-ISSN: 2347-2693

Improving Overall usage of Servers by Measuring Uneven Utiliztion of a Server and allocating the Applications in the Face of Multidimensional Resource Constraints

S.K. Sonkar[1*], M. U. Kharat [2]

[1]Dept. of computer engineering, KKWagh IEE&R Nashik, SP Pune University, India
[2]Dept. of computer engineering, METIOE Nashik, SP Pune University, India

Corresponding Author: Sonkar83@gmail.com, Tel.: +919970061302

Available online at: www.ijcseonline.org

Accepted: 20/Sept/2018, Published: 30/Sept/2018

***Abstract*—**Major objective of cloud provider is to maximize the resource utilization of cloud servers as well as to reduce the energy consumption and operative cost of the datacenter. However, the servers in many existing datacenters are underutilized in practice due to over-provisioning of peak demand. Many times, the datacenter come across situations wherein large number of application requests simultaneously demand multidimensional resources such as CPU, memory, bandwidth. In such situations it is highly impractical for the cloud service provider to satisfy the application requests of all the users within stipulated time, especially when sufficient resources are not available with them. In order to address this problem, we designed a system which measures the resource utilization of all servers before allocating the application requests to server and then dynamically allocate the application requests to the server which is underutilized. This yields in improving the overall utilization of servers. Our system initially checks the server utilization in terms of CPU, Memory and Bandwidth resource utilization against predefined threshold value. If resource of any server goes beyond its threshold value, then application request will not be allocated to that server to avoid the server overloading. That means our system redirect the application request to the underutilized server so as to improve the server resource utilization in the face of multidimensional resource constraints. The experimental results demonstrate that our system improves the overall server resource utilization by 10%.

***Keywords*—** Cloud Service Provider, User Request, Resource utilization, Resource constraints.

I. Introduction

The cloud access information globally, it is available anytime and anywhere on demand [1]. Cloud server is created to manage and store data and runs different applications & deliver the services

The cloud provides mainly three types of services. (SaaS) Software as a service, in this application runs on cloud Infrastructure and accessible from different client devices through a program interface or web browser. (PaaS) Platform as a service, in this get a core OS and other building block services which allow to run third-party or your own application. Application created using programming languages, services, libraries and tools supported by the provider. (Iaas) Infrastructure as a Service, this provide different services such a Storage, virtualization, Load balancers and Networking etc.

There are four types of cloud deployment models on the basis of size and access. Public Cloud Infrastructure is open to use, flexible, low Cost and Reliable. Private Cloud is owned by particular organization, enterprise or institution. Private cloud provides better security, better control than the public cloud. Community cloud is a cloud shared manually by specific community from different organization or firms and use for ventures, joint businesses, and research organization. Hybrid cloud is a combination between two or more clouds (public, private or community) [2].

Virtualization is the technology is to create virtual version of actual machines such as desktop, server, operating system, and network or storage device. Due to the virtualization multiple applications and OS can run simultaneously on the same machine. This minimizes the cost and increases flexibility, utilization and efficiency of exiting computer hardware [3].

Resource Management is a main issue in cloud computing. Providing all requested resource is the challenging task to cloud provider. Resource management and allocation is very complex task in cloud computing thus allocation of cloud resources must be efficient and in a fair manner [4-8].

Load balancing is distributing the amount of work which gives high availability of resources. It also improves the reliability & performance of databases, applications, websites and also different services by dividing the work load across multiple cloud servers. There are some of the cloud computing issues such as Virtual Machine Migration, Energy Management, Resource overloading in server etc. [9-14].

Rest of the paper is organized as follows: section II describes related work. Our contributions are presented in section III. Section IV represents proposed system architecture along with proposed algorithm to handle the system efficiently. Section V describes performance analysis & Conclusion is discussed in section VI.

II. Related Work

This section presents different resource allocation and scheduling techniques which helps to present our proposed work.

Jue Nie, et.al has proposed a scheduling algorithm using hybrid approach i.e. combination of genetic as well as ant colony algorithm which can be used in cloud computing for task scheduling [15].

An approach proposed by Hsin-Yu Shih has analyzed the policies which focuses on logical resources which are free and does not focus on the physical resources, so this work improves the resource utilization [16].

The approach discussed by abrol et.al is about how the service of cloud like PAAS and IAAS controls the management and placement of the resources for the tasks provided to the cloud [17].

Gema Ramadhan et.al has provided an algorithm which provides efficient as well as good mechanism which assigns multiple client requests to multiple cloud nodes using their algorithm which helps in achieving the efficiency [18].

Rohit nagar et.al proposed an algorithm to reducing the execution time for scheduling purpose in cloud environment called as PEFT genetic algorithm. As the proposed approach have some advantages like reduces the completion time and increases the resource utilization and also performs better at scheduling of tasks [19].

A new approach is proposed by Neha Sethi, et.al named as variance honey bee behavior with multi-objective optimization method because of the bad convergence speed of previous approaches like genetic algorithm, non-dominated sorting genetic algorithm and techniques like ant colony optimization where all are NP-complete in nature [20].

Using dragonfly optimization algorithm because of its speed and preciseness in scheduling responsibilities Zahra Amini et.al. Proposed an approach that gives a procedure of allocating resources to digital machines in cloud computing. By using this technique there was improvement in the resource allocation and load balancing between virtual machines as compared to other approaches when considered factors like response and execution times [21].

The main focus given in a work of Ojasvee Kaneria et.al is to speed up the access speed by modifying the load balancing algorithms as the effective use of cloud depends upon the factors like security, privacy and most importantly speed [22]. Zhang jiadong et.al proposed an approach which uses threshold window strategy and an advanced AR prediction model to reduce migration of VMs [23]. The framework also improves reliability of datacenter but the framework also has its issues like fast response time, error handling and distributed response.

P Geetha, et.al summarizes different load balancing algorithm in cloud computing along with its advantages and disadvantages. How all type of load balancing algorithms can be used in hybrid environment like green cloud computing vs mobile cloud computing or any other computing environment are also discussed [24].

Sajeeb Saha et.al developed an approach for cloud migration decision-making algorithm to compute-extensive tasks. It is also used to decide the feasibility of execution on a cloud server instead of a cell device. It can be seen as entirety time may be reduced by aspect of 6 to eight of cellular carrier when a cloud server is utilized [25].

M. Kumar et.al discussed different vm scheduling algorithm in terms of various performance parameters such as Scalability, reliability and QoS and observed that there is enough space available to improve the performance of existing algorithms by using more hybrid optimization techniques [26- 27].

III. Contribution

Our contribution to this work is as follows:

Based on the research gap, proposed system is designed to improve overall usage of servers by measuring uneven utilization of a server and allocate application request in the face of multidimensional (CPU, Memory, Bandwidth) resource constraints.

IV. SYSTEM MODEL

In this section, part A describes the System Workflow whereas section B discusses the Proposed Algorithm.

A. System Workflow

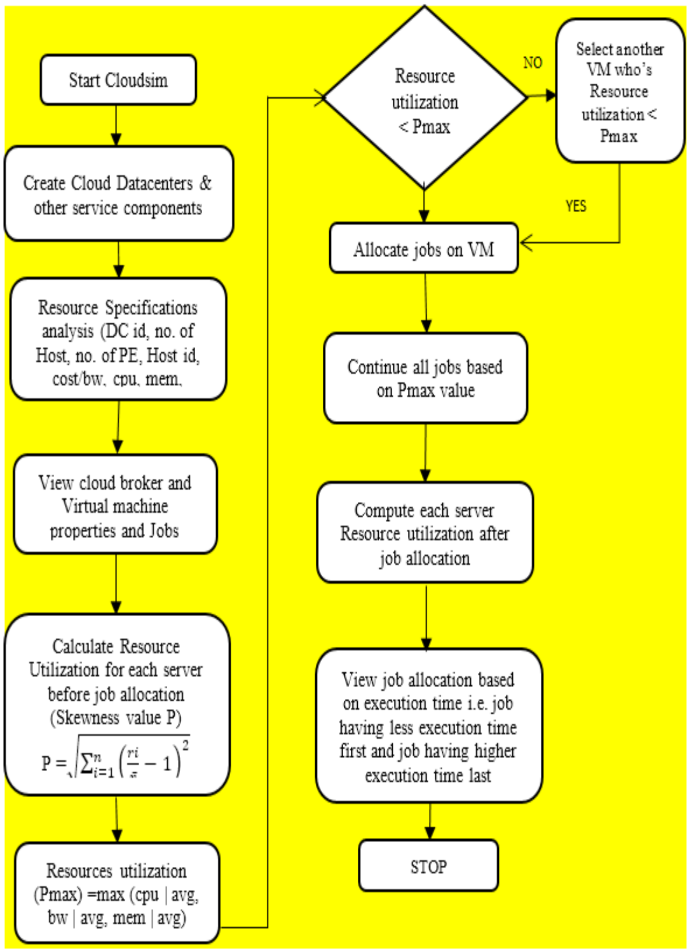


Figure 1.System Workflow

The system workflow is explained as follows:

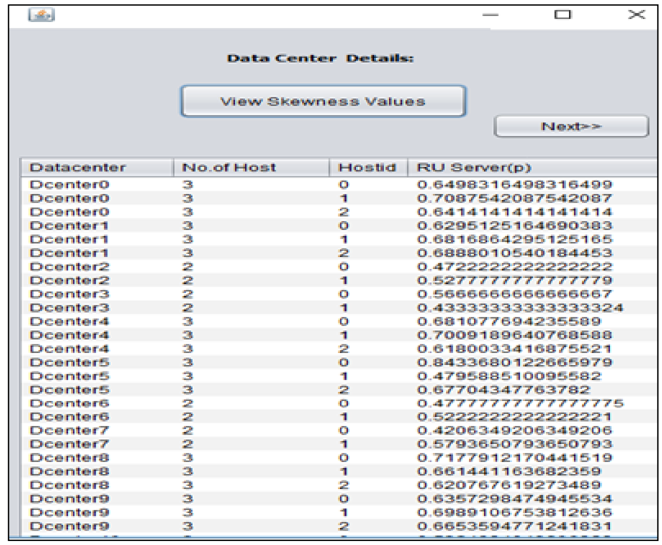
To implement the proposed work, first of all system will get input from cloudsim simulation tool. Following are the major steps for implementation purpose:

First of all, create number of datacenters and other resource components such as number of Virtual machines (VM's), the number of Brokers and the number of Cloudlets or Jobs. In next step we can see Datacentre's properties such as Datacenter ID's, Number of Hosts, Number of Processing Elements (PE), Host ID, Cost/BW, Cost/Mem, and Cost/Million Instruction per Second, Cost/sec, Cost/Storage, MIPS, & Mips/PE. In next step we can view the cloud broker details in terms of Broker name and broker ID. Next we can view VM Properties such as VM ID, Broker ID, the number of Million Instruction per Second (MIPS) executed by that VM, the amount of RAM allocated, and amount of Bandwidth allocated for that VM. Next to this, the Resource utilization (Skewness Value) for each server before job allocation is calculated by using the following formula

$$\text{Skewness Value (P)} = \sqrt{\sum_{i=1}^{n}\left(\frac{ri}{\bar{r}} - 1\right)^2} \quad (1)$$

Let n be the number of resources to be considered and r_i be the utilization of the i^{th} resource.

Let $\bar{r}$ be the average utilization of all resources for the server. The same is shown in fig.2:



Datacenter	No.of Host	Hostid	RU Server(p)
Dcenter0	3	0	0.6498316498316499
Dcenter0	3	1	0.7087542087542087
Dcenter0	3	2	0.6414141414141414
Dcenter1	3	0	0.6295125164690383
Dcenter1	3	1	0.6816864295125165
Dcenter1	3	2	0.6888010540184453
Dcenter2	2	0	0.4722222222222222
Dcenter2	2	1	0.5277777777777779
Dcenter3	2	0	0.5666666666666667
Dcenter3	2	1	0.43333333333333324
Dcenter4	3	0	0.681077694235589
Dcenter4	3	1	0.7009189640768588
Dcenter4	3	2	0.6180033416875521
Dcenter5	3	0	0.8433680122665979
Dcenter5	3	1	0.479588510095582
Dcenter5	3	2	0.67704347763782
Dcenter6	2	0	0.47777777777777775
Dcenter6	2	1	0.5222222222222221
Dcenter7	2	0	0.4206349206349206
Dcenter7	2	1	0.5793650793650793
Dcenter8	3	0	0.7177912170441519
Dcenter8	3	1	0.661441163682359
Dcenter8	3	2	0.620767619273489
Dcenter9	3	0	0.6357298474945534
Dcenter9	3	1	0.6989106753812636
Dcenter9	3	2	0.6653594771241831

Fig.2: Initial Resource utilization of servers.

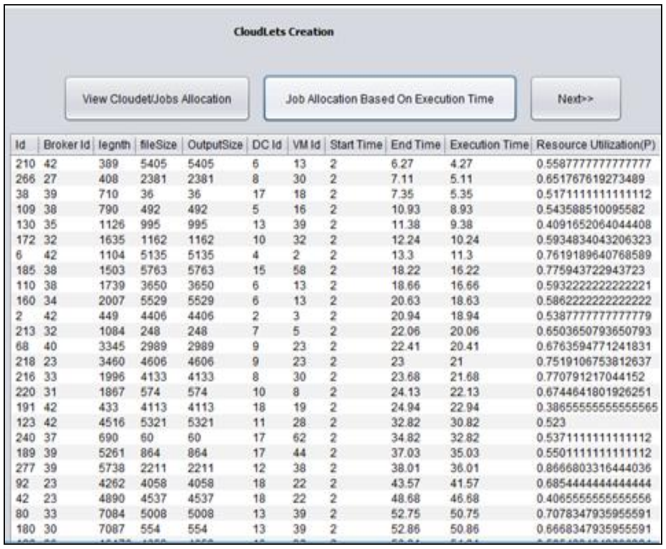
Fig.2 shows initial resource utilization of servers. By observing this resource utilization system can know that which server is highly utilized and which server is low utilized. Then the average resource utilization in terms of CPU, Memory and Bandwidth is calculated. Whichever may be higher resource (CPU, memory, bandwidth) utilization, considering that value is Pmax value. Which is calculated by following equations.

$$Resource\ Utilization\ (p) = \max(CPU\ avg, BW\ avg, Mem\ avg) \quad (2)$$

$$Pmax\ Value \leftarrow Highest\ Resource\ utilization\ Value\ or\ skewness\ value \quad (3)$$

Then jobs are allocated to only that vm which are having Resource Utilization below the Pmax to avoid the overloading of the server. In other words, jobs are allocated to the less utilized server and avoid the high utilized server resulting that improve the overall server utilization.

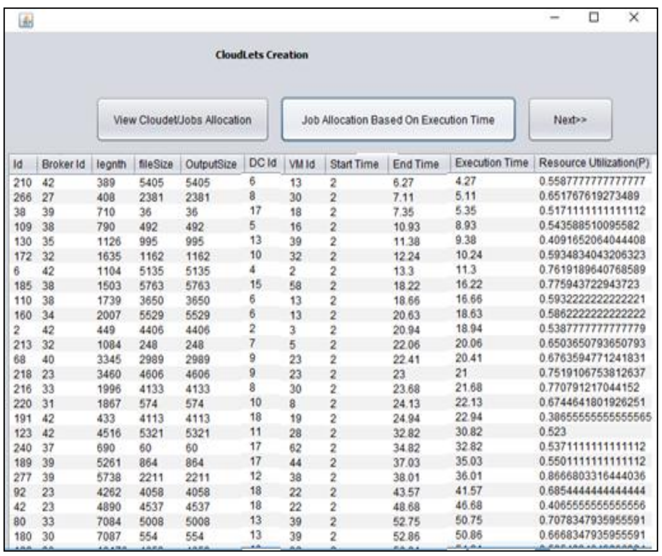
Before cloudlet allocation resource utilization of datacenter 6 –host id-0 is 0.47777(As shown in fig.2) and after the cloudlet allocation the resource utilization is 0.5587 shown in fig.3. Same things are for all the servers, which indicates the overall server resource utilization improves.



Id	Broker Id	legnth	fileSize	OutputSize	DC Id	VM Id	Start Time	End Time	Execution Time	Resource Utilization(P)
210	42	389	5405	5405	6	13	2	6.27	4.27	0.5587777777777777
266	27	408	2381	2381	8	30	2	7.11	5.11	0.651767619273489
38	39	710	36	36	17	18	2	7.35	5.35	0.5171111111111112
109	38	790	492	492	5	16	2	10.93	8.93	0.543588510095582
130	35	1126	995	995	13	39	2	11.38	9.38	0.4091652064044408
172	32	1635	1162	1162	10	32	2	12.24	10.24	0.5934834043206323
6	42	1104	5135	5135	4	2	2	13.3	11.3	0.7619189640768589
185	38	1503	5763	5763	15	58	2	18.22	16.22	0.775943722943723
110	38	1739	3650	3650	6	13	2	18.66	16.66	0.5932222222222221
160	34	2007	5529	5529	6	13	2	20.63	18.63	0.5862222222222222
2	42	449	4406	4406	2	3	2	20.94	18.94	0.5387777777777779
213	32	1084	248	248	7	5	2	22.06	20.06	0.6503650793650793
68	40	3345	2989	2989	9	23	2	22.41	20.41	0.6763594771241831
218	23	3460	4606	4606	9	23	2	23	21	0.7519106753812637
216	33	1996	4133	4133	8	30	2	23.68	21.68	0.770791217044152
220	31	1867	574	574	10	8	2	24.13	22.13	0.6744641801926251
191	42	433	4113	4113	18	19	2	24.94	22.94	0.3865555555555565
123	42	4516	5321	5321	11	28	2	32.82	30.82	0.523
240	37	690	60	60	17	62	2	34.82	32.82	0.5371111111111112
189	39	5261	864	864	17	44	2	37.03	35.03	0.5501111111111112
277	39	5738	2211	2211	12	38	2	38.01	36.01	0.8666803316444036
92	23	4262	4058	4058	18	22	2	43.57	41.57	0.6854444444444444
42	23	4890	4537	4537	18	22	2	48.68	46.68	0.4065555555555556
80	33	7084	5008	5008	13	39	2	52.75	50.75	0.7078347935955591
180	30	7087	554	554	13	39	2	52.86	50.86	0.6668347935955591

Fig.3: Cloudlets Allocation

Once the job is allocated to different VM's, proposed system applies the job priority algorithm based on execution time, in which the job with less execution time will get first priority for the execution.



Id	Broker Id	legnth	fileSize	OutputSize	DC Id	VM Id	Start Time	End Time	Execution Time	Resource Utilization(P)
210	42	389	5405	5405	6	13	2	6.27	4.27	0.5587777777777777
266	27	408	2381	2381	8	30	2	7.11	5.11	0.651767619273489
38	39	710	36	36	17	18	2	7.35	5.35	0.5171111111111112
109	38	790	492	492	5	16	2	10.93	8.93	0.543588510095582
130	35	1126	995	995	13	39	2	11.38	9.38	0.4091652064044408
172	32	1635	1162	1162	10	32	2	12.24	10.24	0.5934834043206323
6	42	1104	5135	5135	4	2	2	13.3	11.3	0.7619189640768589
185	38	1503	5763	5763	15	58	2	18.22	16.22	0.775943722943723
110	38	1739	3650	3650	6	13	2	18.66	16.66	0.5932222222222221
160	34	2007	5529	5529	6	13	2	20.63	18.63	0.5862222222222222
2	42	449	4406	4406	2	3	2	20.94	18.94	0.5387777777777779
213	32	1084	248	248	7	5	2	22.06	20.06	0.6503650793650793
68	40	3345	2989	2989	9	23	2	22.41	20.41	0.6763594771241831
218	23	3460	4606	4606	9	23	2	23	21	0.7519106753812637
216	33	1996	4133	4133	8	30	2	23.68	21.68	0.770791217044152
220	31	1867	574	574	10	8	2	24.13	22.13	0.6744641801926251
191	42	433	4113	4113	18	19	2	24.94	22.94	0.3865555555555565
123	42	4516	5321	5321	11	28	2	32.82	30.82	0.523
240	37	690	60	60	17	62	2	34.82	32.82	0.5371111111111112
189	39	5261	864	864	17	44	2	37.03	35.03	0.5501111111111112
277	39	5738	2211	2211	12	38	2	38.01	36.01	0.8666803316444036
92	23	4262	4058	4058	18	22	2	43.57	41.57	0.6854444444444444
42	23	4890	4537	4537	18	22	2	48.68	46.68	0.4065555555555556
80	33	7084	5008	5008	13	39	2	52.75	50.75	0.7078347935955591
180	30	7087	554	554	13	39	2	52.86	50.86	0.6668347935955591

Fig.4: job allocation based on execution time.

As shown in fig.4 cloudlet id 210-Datacenter id-6-vm id-13 having execution time 4.27ms whereas cloudlet id 110-Datacenter id-6-vm id-13 having execution time 16.66 ms, also cloudlet id 160-Datacenter id-6-vm id-13 having execution time 18.63 Millisecond. Therefore, our job priority algorithm will give the first priority to cloudlet id 210 then cloudlet 110 and finally to cloudlet id 160 to execute.

B Algorithm:

1. Take the input parameters form cloudsim i.e. Number of Datacenters, Number of VM's, Number of Cloud Brokers and Number of Cloudlets.
2. View Datacenters characteristics
3. View Cloud Broker details
4. View Virtual Machine Properties
5. Calculate Server Resource Utilization (Skewness Value)

   Skewness Value (P) = $\sqrt{\sum_{i=1}^{n}\left(\frac{ri}{\bar{r}}-1\right)^{2}}$

   Let n be the number of resources to be consider and r_i be the utilization of the i^{th} resource.

   Let $\bar{r}$ be the avg. utilization of all resources for the server.
6. We know that,
   Let DC_n be set of datacenters
   Where n=1 to N. N=No of Resources
   Find Ri ←{CPU, Memory, Network}
7. Compute each server resource utilization ←(before job allocation)
   For each Job_i in JList
   Apply Parallel Scheduling
   Allocate Job_i to VM_i in Hostlist
   for i=0 : (DC_i) **do**

   for j=0: (PM_j) **do**

   for k=0: (VM_k) **do**

   Mem+= Men;

   CPU += CPU;

   Bw+= BW;

CPU avg utilization = total CPU utilization| no_of_vm;
BW avg utilization = total BW utilization | no_of_vm;
Mem avg utilization = total Mem. utilization| no_of_vm;
Resource Utilization (p) =max (CPU avg, BW avg, Mem. avg)
Pmax Value ← Highest Resource utilization Value or skewness value

8. Every job allocate based on resource utilization or skewness value.
9. Continue all jobs allocation based on skewness (P)Value
10. Compute each server resource utilization (After job allocation)
11. For job=0; n do
    View job allocation based on execution time i.e. the job having less execution time first then job having higher execution time respectively.
12. Which results improving overall server resource Utilization.

V. PERFORMANCE ANALYSIS

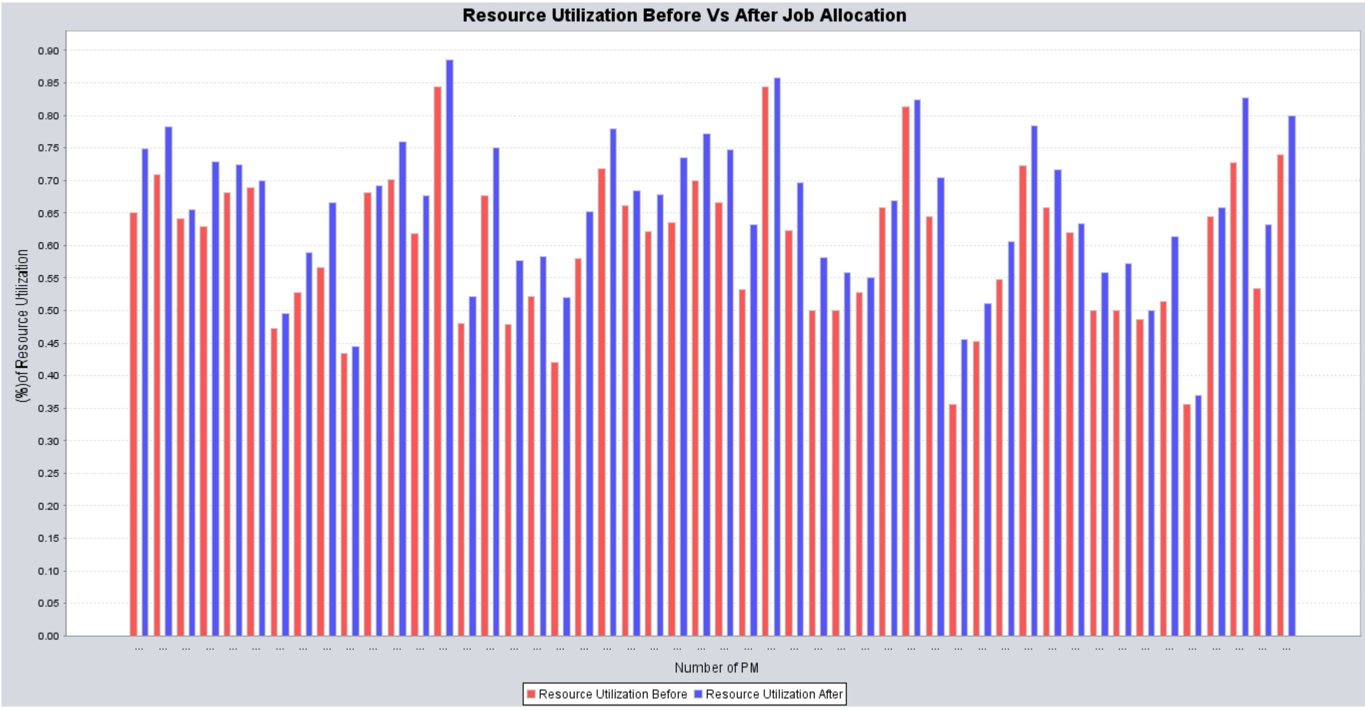


Fig.5: Graph for Resource Utilization before Vs after Job Allocation

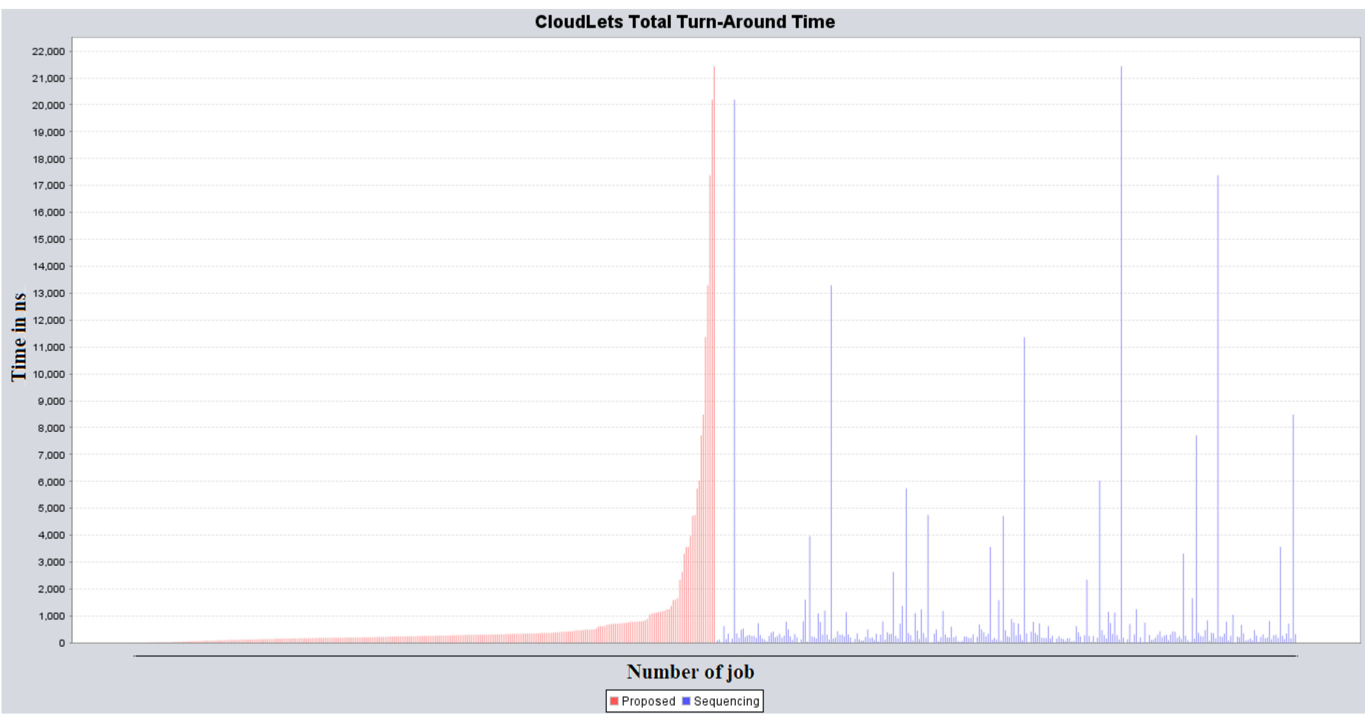


Fig.6: Graph for Cloudlets Turnaround time.

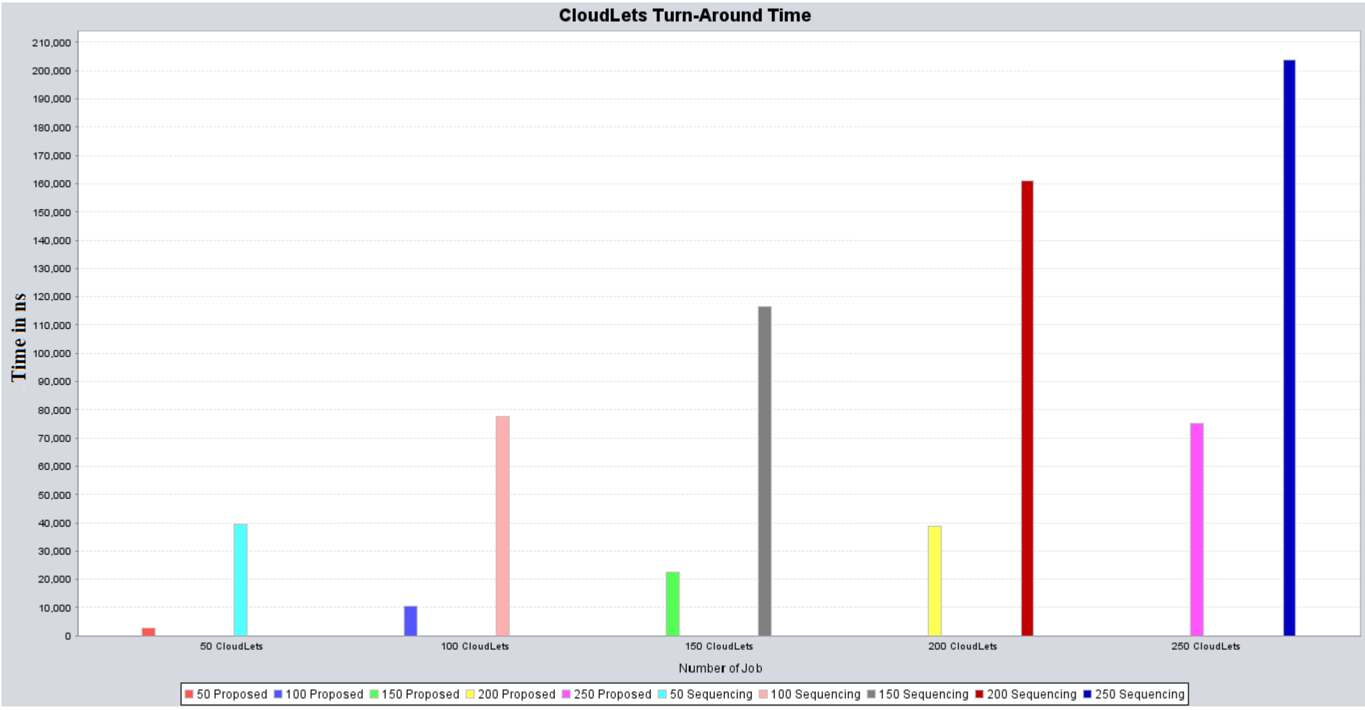


Fig 7: Graph for Cloudlets total turnaround time

The System is implemented using cloudsim tool having physical configuration, i3 processor, 4GB RAM and 1TB HDD. Fig.5 shows graph for Server Resource utilization for all the servers in the system before and after allocation of application requests to them. X-axis indicate the number of servers (Physical Machines), whereas Y-axis indicates Server Resource Utilization. The Red bar line indicates Resource utilization before the allocation of application requests, whereas the blue line indicates Resource Utilization after allocation of application requests. For instance, for Physical machine 0 (PM0) the resource utilization before allocation is 0.65, whereas after the allocation it is 0.75. It is observed that like PM0, the overall utilization of all the servers is improved. Fig.6. shows that Graph for cloudlets total turnaround time. The results in this figure represent how much time is required to complete the jobs (user requests). As shown in fig.6, the time required to complete the user requests by our system is gradually increased step by step. Whereas by using sequencing algorithm the time required to complete the jobs is variable. Fig7 shows the graph for cloudlets (job) total turnaround time. This figure reveals that the time required to complete 50 jobs using our system is 2681.32 nanosecond, whereas time required to complete the 50 jobs using sequencing method is 22304 nanosecond. Like that time required to complete the 250 jobs using proposed System is 61955.57 nanosecond, whereas by using sequencing algorithm the time required to complete the jobs is 102953.52 nanosecond. Therefore, it is observed that Cloudlet total turnaround for 250 jobs is reduced by 6.833e-7 minutes.

VI. CONCLUSION

To improve job performance and resource utilization in a virtual machine-based cloud environment several Multidimensional resource constraint such as CPU, memory and bandwidth are used. Novel resource utilization model proposed here can handle the resource requirements and also improves the overall utilization of a servers in the face of multidimensional resource constraints.

Based on server resource utilization observation, requests are mapped to low utilized server so that to avoid server overloading as well as to improve the overall resource utilization of servers.

The experimental results shows that in proposed technique Server Resource Utilization is improved by 10%. Also, the time required to complete the jobs in the proposed system is gradually increasing, whereas by using sequencing algorithm the time required to complete the jobs was varying significantly. Further it is also observed that Cloudlet total turnaround for 250 jobs is reduced by 6.833e-7 minutes.

ACKNOWLEDGMENT

FA would like to thanks KKWagh IEE&R, Nashik for providing facilities to make successful work of this paper.

REFERENCES

[1] Abhinandan S. Prasad, and Shrisha Rao,"A Mechanism Design Approach to Resource Procurement in Cloud Computing", IEEE Transactions on Computers, Vol. 63, No. 1, Pp.17-30, January 2014.

[2] Jens-Matthias Bohli, Nils Gruschka, Meiko Jensen, Member, IEEE ,Luigi Lo Iacono, and Ninja Marnau," Security and Privacy-Enhancing Multicloud Architectures" IEEE Transactions On Dependable & Secure Computing, Vol. 10, No. 4, July/August 2013.

[3] En-Hao Chang, Chen-Chieh Wang, Chien-Te Liu, Kuan-Chung Chen, Student Member, Ieee, And Chung-Ho Chen, Member, IEEE," Virtualization Technology For Tcp/Ip Offload Engine", IEEE Transactions On Cloud Computing, Vol. 2, No. 2, April-June 2014.

[4] Zhen Xiao, Senior Member, Ieee, Qi Chen, And Haipeng Luo," Automatic Scaling Of Internet Applications For Cloud Computing Services", IEEE Transactions On Computers, Vol. 63, No. 5, May 2014.

[5] Hui Zhang, Guofei Jiang, Kenji Yoshihira, And Haifeng Chen,"Proactive Workload Management In Hybrid Cloud Computing IEEE Transactions On Network And Service Management", Vol. 11, No. 1, March 2014

[6] S.K.Sonkar, .M.U.Kharat," A Review on Resource Allocation and VM Scheduling Techniques and a Model for Efficient Resource Management in Cloud Computing Environment", IEEE International conference on ICTBIG), ISBN: 978-1-5090-5519-9. Nov. 2016, DOI: 10.1109/ICTBIG.2016.7892646.

[7] Shi J.Y., Taifi M., Khreishah A., "Resource Planning for Parallel Processing in the Cloud," in IEEE 13th International Conference on High Performance and Computing, Nov. 2011, pp. 828-833.

[8] S.K.Sonkar, Dr.M.U.Kharat," A Survey on Resource Management in Cloud Computing Environment", International Journal of Advanced Trends in Computer Science and Engineering, vol4 (issue4), Pages: 48 – 51, July - August 2015, ISSN: 2278-3091.

[9] Zhen Xiao, Senior Member, IEEE, Weijia Song, and Qi Chen, "Dynamic resource allocation using virtual machine in cloud computing environment," IEEE Transaction on Parallel and Distributed Systems, vol.24, no.6, pp.1107-1117, June 2013.

[10] J. Hu, J. GU, G. Sun, and T. Zhao, "A scheduling strategy on load balancing of virtual machine resources in cloud computing environment," in 3rd IEEE Int. Symp. On Parallel Architectures, Algorithms and Programming (PAAP), 2010, 18-20 Dec. 2010, pp.89-96.

[11] Xin Li; Zhuzhong Qian; Ruiqing Chi; Bolei Zhang; Sanglu Lu," Balancing Resource Utilization for Continuous Virtual machine Requests in Clouds", Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS),IEEE Conference Publication 2012 ,Pages: 266 - 273, DOI: 10.1109/IMIS.2012.72

[12] Sonali L. Vidhate, M. U. Kharat "Resource Aware Monitoring in Distributed System using Tabu Search Algorithm", International journal of Computer Applications, Vol no.96, issue 23, June 2014, ISSN:0975-8887

[13] Daochao Huang, Peng Du, Chunge Zhu, Hong Zhang, Xinran Liu"Multi-resource Packing for Job Scheduling in Virtual Machine Based Cloud Environment", IEEE symposium on Service Oriented System Engineering, Catalogue number: 978-1-4799-8365, DOI: 10.1109/SOSE.2015.30.

[14] Ryan Jansen, Paul R. Brenner, "Energy Efficient Virtual Machine Allocation in the Cloud an Analysis of Cloud Allocation Policies", 2nd IEEE conference on Green computing, Catalogue number: 978-1-4577-1221-0/11.

[15] Jun Nie, "Research on Task Scheduling Strategy Based on Cloud Computing Environment," Journal of Applied Science and Engineering Innovation, Vol.5 No.1, 2018, pp. 9-12,ISSN:2331-9070

[16] Hsin-Yu Shih, Jenq-Shiou Leu, "Improving Resource Utilization in a Heterogeneous Cloud Environment," IEEE 18th Asia-Pacific Conference on Communications (APCC) 2012, pp. 185-189, 2012, DOI: 10.1109/APCC.2012.6388127.

[17] Preeti Abrol, Dr. Savita Gupta, Karanpreet Kaur, "Analysis of Resource Management and Placement Policies using a new Nature Inspired Meta Heuristic SSCWA avoiding Premature Convergence in Cloud", IEEE International Conference on Computational Techniques in Information and Communication Technologies(ICCTICT),2016,DOI: 10.1109/ICCTICT.2016.7514659

[18] Gema Ramadhan, Tito Waluyo Purboyo, Roswan Latuconsina, "Experimental Model for Load Balancing in Cloud Computing Using Throttled Algorithm", International Journal of Applied Engineering Research, Volume 13, Number 2 (2018) pp. 1139-1143, ISSN 0973-4562

[19] Rohit Nagar, Deepak K. Gupta and Raj M. Singh , "Time Effective Workflow Scheduling using Genetic Algorithm in Cloud Computing," I.J. Information Technology and Computer Science, 2018, 1, 68-75 Published Online January 2018 in MECS (http://www.mecs-press.org/) DOI: 10.5815/ijitcs.2018.01.08

[20] Neha Sethi, Dr.Surjit Singh, Dr.Gurvinder Singh, " Multiobjective Artificial Bee Colony based Job Scheduling for Cloud Computing Environment", International Journal of Mathematical Sciences and Computing, 2018, 1, 41-55 Published Online January 2018 in MECS (http://www.mecs-press.net/ijmsc) DOI: 10.5815/ijmsc.2018.01.03

[21] Zahra Amini, Mehrdad Maeen, Mohammad Reza Jahangir, "Providing a load balancing method based on dragonfly optimization algorithm for resource allocation in cloud computing," International Journal of Networked and Distributed Computing, Vol. 6, No. 1, Jan 2018 pp.35-42.

[22] Ojasvee Kaneria, R K Banyal , "Analysis and Improvement of Load Balancing in Cloud Computing," IEEE International Conference on ICT in Business Industry & Government (ICTBIG), 2016, DOI: 10.1109/ICTBIG.2016.7892711

[23] Zhang Jiadong, Liu Qiongxin, Chen Jiayu, "An Advanced Load Balancing Strategy For Cloud Environment," 17th IEEE International Conference on Parallel and Distributed Computing, Applications and Technologies, 2016, DOI 10.1109/PDCAT.2016.58

[24] P.Geetha, Dr.C.R.Rene Robin, "A Comparative-Study of Load-Cloud Balancing Algorithms in Cloud Environments", IEEE International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS-2017), DOI: 10.1109/ICECDS.2017.8389549.

[25] Sajeeb Saha, Mohammad S. Hasan, "Effective Task Migration to Reduce Execution Time in Mobile Cloud Computing", Proceedings of the 23rd International Conference on Automation & Computing, University of Huddersfield, Huddersfield, UK, 7-8 September 2017.

[26] M. Kumar, Sumar, S. Singh, "A survery on Virtual Machine Scheduling Algorithms in Cloud Computing", IJCSE, Vol.6,issue3,pp.485-490, March 2018,.E-ISSN-2347-2693.

[27] Bhupesh Kumar Devgan, Amit Agarwal, Venkatadri M, Ashutosh Pasricha, "Resource Scheduling in Cloud: A Comparative Study", IJCSE, Vol.6, issue8, pp.168-173, Aug 2018. E-ISSN-2347-2693

Authors Profile

Mr. S. K. Sonkar- Completed the Bachelor and Master degree in Computer science and Engineering from SRTMU Nanded. He is presently Ph.D Research Scholar of KKWagh IEE&R Nashik, SP Pune University & working in AVCOE Sangamner (MH). His area of interests include Computer Network and Cloud Computing.

Dr. M. U. Kharat- Completed Ph.D (Computer Science & Engg.) From Devi Ahilya University, Indore, India. Presently he is working at MET's IOE, Nashik, Maharashtra, India, as Professor & Head, Computer Engineering Department. He has presented good number of papers at National and International conferences and also published papers in National and International Journals on various aspects of Computer Engineering and Networks. He has worked in various capacities in academic institutions at the level of Professor, Head of Computer Engineering Department, and Principal. His areas of interest include, Computer Networks, Distributed System and the Cloud Computing.

