International Journal on Recent and Innovation Trends in Computing and Communication
Volume: 5 Issue: 7

ISSN: 2321-8169
156 – 161

Migration Control in Cloud Computing to Reduce the SLA Violation

Deeksha Bundela
Department of computer science
Sant Longowal Institute of Engineering and Technology
Longowal, India
dikshabundela.1994@gmail.com

Major Singh Goraya
Department of computer science
Sant Longowal Institute of Engineering and Technology
Longowal, India
mjrsingh@yahoo.com

Abstract*:*The requisition of cloud based services are more eminent because of the enormous benefits of cloud such as pay-as-you-use flexibility,scalability and low upfront cost. Day-by-day due to growing number of cloud consumers the load on the datacenters is also increasing. Various load distribution and dynamic load balancing approaches are being followed in the datacenters to optimize the resource utilization so that the performance may be maintained during the increased load. Virtual machine (VM) migration is primarily used to implement dynamic load balancing in the datacenters. But, the poorly designed dynamic VM migration policies may negate its benefits. The VM migration overheads result in the violations of service level agreement (SLA) in the cloud environment.In this paper,an extended VM migration control model is proposedto minimize the SLA violations while controlling the energy consumption of the datacenter during VM migration. The parameters of execution boundary threshold is used to extend an existing VM migration control model. The proposed model is tested through extensive simulations using CloudSim toolkit by executing real world workload. Results are obtained in terms of number of SLA violations while controlling the energy consumption in the datacenter. Results show that the proposed modelachieves better performance in comparison to the existing model.

Keywords:cloud computing; energy efficiency; SLA; VM migration; VM consolidation

____*****____

I. INTRODUCTION

Cloud computing provides a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with the minimal management effort [1]. Cloud resources are provided to the cloud usersthrough Software-as-a-Service (SaaS), Platform-as-a-Service (PaaS) and Infrastructure-as-a-Service (IaaS) service models[2,3].

In the cloud environment, the computing resources are placed in the geographically located datacenters which are the large scale store house for the resources and services[4],[5]. In the cloud environment, computing services are to be delivers as per some predefined service level agreements (SLAs). SLA is a contract between the service provider and the consumer to define the quality/conditions of service delivery. In the recent time, the acceptability of cloud services is increasing around the world which is increasing the load on the datacenters. The chances of violation of SLAs are also becoming high. SLA violations result in lost business revenue.Further the high capacity servers and other associated equipment in the datacenters consumehuge amount of energy to fulfill the consumers'computing needs.For example, inU.S.,datacenters consume about 70 billion kWh energy (approximately 1.8% of total U.S. consumption), worth $4.5 billion. Therefore, the operational expenditure of the datacenters is increasing. To ensure SLAs and minimize the operational expenditure is one of the primary objective of the datacenter deployment. Various techniques have been proposed in the literature to minimize the energy consumption and SLA violation in cloud datacenters. For energy efficiency, the techniques such as Dynamic Voltage and Frequency Scaling (DVFS), Dynamic Voltage Scaling (DVS), Dynamic Power management (DPM), Server Consolidation VM Migration (SCVM), etc. are primarily used. In SCVM, the workload from an underutilized server is moved (VM migration) to another server so that the underutilized server could be shut down in order to reduce the energy consumption as well as SLA violation.VM migration is a process of cancelling a VM from one server and creating the same on another server[6], [7]. Practically, two types of VM migration are used, viz. *live VM migration* and *non-liveVM migration*[8]. In Live migration, a virtual machine is transferred from one physical server to another physical server, while connected to the client or applications of that virtual machine. In Non-Live migration, application services are stopped during migration.Generally, it takes about 15 seconds to create a VM on a server [9].

In this paper, an extended migration controlmodel (based on SCVM) is proposed for reducing theService Level Agreement (SLA) violation in cloud computing environment with controlled energy consumption [10]-[12].The prime strategy used in the paper is to select a VM for migration in such a way that the chances of SLA violation are minimized.

The proposed model is evaluated through extensive simulations using CloudSimtoolkit in terms of SLA and energy consumption. The proposed modelis also compared with an existing model. Results show that the proposed model considerably reduces theSLA violations in cloud environment while controlling the energy consumption.

The remaining paper is outlined as follows: section II includes the related work. Section III describes the proposed model. The performance evaluation is given in Section IV. The paper is finally concluded in Section V.

II. RELATED WORK

In 2012, Beloglazov et al. [13] proposed an energy-aware allocation heuristics providing data center resources to client applications that improved the energy efficiency of the data center without negotiating the Quality of Services (QoS). Author primarily focused on reduction of energy consumption in data center as well as delivering QoS to the client by introducing energy-efficient resource allocation policies and scheduling algorithms. As a result, considerable reduction was obtained in energy consumption and service level agreement (SLA).In 2012, Beloglazov and Buyya [14] worked on dynamic consolidation of VMs, which improved both the energy consumption and SLA violation. Energy consumption can be reduced in datacenter by switching off the idle server. But,due to the variability of workloads experienced by applications, SLAwas being violated. To solve this problem, the VM placement was optimized continuously in an online manner. Results showed a great improvement in both energy consumption and SLA violation. In 2013, Cao and Dong [15]proposed a service level agreement(SLA) violation decision algorithm. By using host overloading detection and VM selection algorithms, energy consumption and SLA violation were reduced. In 2014, Huang et al.[16] proposed an energy consumption model. Energy Consumption was evaluated by combining the computing cost, switching cost, and migration cost. Authors focused on reducing both energy consumption and SLA violation.In 2015, Mohammad Alaul et al. [10], described the selection policies of VMs. Author modified the existing overload detection algorithm by using mean and standard deviation.Three VM selection algorithms were redesigned using migration control. In 2015, Mohammad Alaul et al.[11], proposedthree modified VM selectionalgorithms; viz. Minimum Migration Time with Migration Control (MMTMC), Maximum Correlation with Migration Control (MCMC), Random Selection with Migration Control (RSMC). VM selection algorithmswere combined with the migration control toreduce the energy consumption. In 2017, Hasan and Goraya [17] proposed a fault-tolerant service framework in cloud with better resource utilization. The authors customized the pre-existing technique of cooperative computing [18], [19].

III. PROPOSED WORK

Host overloading detection and VM selection methods are used to reduce the energy consumption and SLA violation.Basic VM selection policies used in papers [11], [14] are as following:

1. Minimum Migration Time (MMT)

Minimum migration time policy selects those VMs for migration which can be migrated within least time limit. *MMT* can be measured by *RAM* utilized by the *VM* to the bandwidth offered to the given host.

$$m \in V_k \mid \forall n \in V_k, \frac{RAM\,(m)}{NET_k} \leq \frac{RAM\,(n)}{NET_k}$$

where *RAM(m)*is the amount of *RAM* currently utilized by the VM *m*. Suppose, *k* is the host which have two VMs *m* and *n*, $V_k$be the set of VMs. $CPUu(m_t)$ is termed as CPU utilization of *VM* at time *t*.$CPUu(m_{t-1})$, $CPUu(m_{t-2})$ .... $CPUu(m_{t-p})$ *are CPU* utilization of up to previous *p* number time frames where overload detection algorithm was activated. Two thresholds are applied on VMs. One of them is $CPU_{Threshold}$ whichis used to determine whether the VM is steady or not, *p* is the window size and *NET* is the bandwidth.

At any moment *t*, the Minimum Migration Time find VM *m* will be selected for migration for the below formulas:

only if ;

$$\frac{[\,CPU_u(m_t) + CPU_u(m_{t-1}) + CPU_u(m_{t-2}) \ + \cdots + CPU_u(m_{t-p})]}{p+1} \leq CPU_{threshold} \quad (1)$$

2. Random Selection (RS)

VM will be randomly selected [4,5,6] for migration by random selection policy. By using a uniformly distributed discrete random function, VM will be selected for migration from the overloaded host to another host. Equations of RS policy are described below:

$$v \overset{d}{\Leftarrow} R(0, |V_k|)$$

3. Maximum Correlation (MC)

Maximum correlation policy [10],[11]states that the probability of server overloading is based on the resource usage by applications running on an oversubscribed server which shows a positive correlation between both of them. Mostly advice to migrate a VM if a higher correlation of *CPU* usage is found between one or more *VM*.

To select the migratable VM we use maximum correlation with extended migration control. Multiple correlation coefficient is used to measure the correlation between *CPU* utilizations by VMs [15]. Correlation coefficient is termed as *R*.

Suppose, *p*is the total number of VM and $M_1, M_2 \ldots M_P$ is the *CPU* utilization of *p* VM. Let *N* represents the *VM* where we want to determine the maximum correlation with jth*VM*. Here *p-1* and *1* are independent and dependent random number of *N*. Main objective is to find the intensity of the correlation between *N* and *p-1* remaining random variables. *N* and *(p-1) m1* vectored of *N* is represented as *n*.

$$M = \begin{bmatrix} 1 & m_{1,1} & m_{1,p-1} \\ \vdots & \vdots & \vdots \\ 1 & m_{p-1,1} & m_{p-1,p-1} \end{bmatrix}$$

$$N = \begin{bmatrix} n_1 \\ \vdots \\ n_p \end{bmatrix} \quad (2)$$

By using the below equation, we can obtain the predicted value of dependent random variable. Here predicated value is expressed as n^.

$$b = (M^T M)^{-1} \quad M^T n \quad (3)$$

Correlation coefficient *(CC)* can be obtained from the below equation:

$$R^2_{N,M_1,\ldots M_{p-1}} = \frac{\sum_{j=1}^{p}(n_j - x_n)^2(ń - x_{ń})^2}{\sum_{j=1}^{p}(n_j - x_n)^2 \sum_{j=1}^{p}(ń - x_{ń})^2} \quad (4)$$

For finding the correlation coefficient mean value is required here $x_n$and $x_{\wedge n}$ are the sample mean of *N* and *Ń*.

$$R^2_{m_j,m_{1\cdots m_{p-j}},m_{p+j\ldots m_p}} \quad (5)$$

Maximum correlation policy selects a VM *v* which satisfies the conditions defined in below equation (6). This condition is checked with migration control described in equation (1)

$$v \in V_k | \forall u \in V_k, R^2_{m_v,m_{1\cdots m_{v-1}},m_{v+1\ldots m_p}} \geq R^2_{m_u,m_{1\cdots m_{u-1}},m_{u+1\ldots m_p}} \quad (6)$$

The main objective of the presented paper is to reduce the SLA violation and to control the energy consumption consumption. In this wok, we introduce three modified VM selection policies by extended the migration control parameter which are as follows:

- **Minimum Migration Time with Extended Migration Control (MMTEMC)**
- **Random Selection with Extended Migration Control (RSEMC)**
- **Maximum Correlation with Extended Migration Control (MCEMC)**

An additional migration control parameter is developed i.e. $\text{ExecutionBoundary}_{\text{Thrshold}}$. Extended migration control threshold parameters exclude the VM which have less than 70% of the execution. $\text{ExecutionBoundary}_{\text{Thrshold}}$ is defined in equation (7).

$$\frac{(CurT_m - AT_m)}{BT_m} \leq ExecutionBoundary_{Thershold} \quad (7)$$

Where $CurT_m$, AT_m and BT_m are the current time, arrival time and the burst time of the task of VM m respectively. VM will be selected for migration by using the above defined policies only if the VM will satisfies both the threshold i.e. $\text{CPU}_{\text{Threshold}}$ and $\text{ExecutionBoundary}_{\text{Thrshold}}$. These reduce the SLA violation as well as energy consumption. By using the equation (7)VM migration list is reduced due to the $\text{ExecutionBoundary}_{\text{Thrshold}}$. This threshold restricts those VMs to be migrated which is near to the task completion.

IV. Performance Evaluation

Performance Metrices

Energy consumption: Energy is defined as a capacity to do work and its unit for measure is Joule [17]. Total energy consumption is calculated by combining the energy of computation, migration and switching.Energy consumption while executing the tasks (computation energy) calculated as follows:

$$E_{comp} = \sum_{j=1}^{M} \int e_c\,(v_j(t))\,dt \quad (21)$$

where, $e_c$is the energy consumption ofj^{th} server.

VM migration cost is defined as energy consumed by network devices which provides a communication link and the energy consumed by the migrated VMs memory size [16]. Energy consumption while migrate the VMs (Migration energy) is calculated as:

$$E_{mig} = \sum_{j=1}^{M} e_m \frac{r_j}{bw_j} \quad (22)$$

where, *M* is the total number of *VM*s which are migrated. $r_j$is the memory size of j^{th} migrated server, unit of energy consumption for migration is termed as e_m.andbw_j is the bandwidth.

Energy consumption while servers changes its state from sleep to active state (Switching energy) is calculated as:

$$E_{swit} = \sum_{j=1}^{k} \frac{E_{sj} * T_{sj}}{2} \quad (23)$$

Here, E_{sj}energy consumption difference between sleep mode and active idle mode of the j^{th} server, $T_{sj}$is the time taken for thej^{th} server to switch from sleep mode to active mode, and *k* is the rebooted servers.

The total energy consumption is calculated as:

$$E_{total} = E_{comp} + E_{mig} + E_{swit} \quad (24)$$

SLA violation: SLA is an agreement between the customer and the service provider [16], [17], [22]. A quality of service (*QoS)* is negotiated between service provider and customer. If the provider cannot meet the required *QoS* then *SLA* is being violated. *SLA*violation is calculated as:

- *SLAV overloaded CPU*: This is the ratio of the total active time of the *SLAV* time and server,
- *SLAV unmet MIPS*: This is the ratio of total indirect *MIPS* during *VM* migration.

*SLAV Composite = SLAV overloaded CPU * SLAV unmute MIPS*

Experimental Setup

To evaluate the proposed algorithm, CloudSim 3.0.3 simulation toolkit is used[20], [21]. Following parameters are used to implement the proposed algorithm.

- *G4* and *G5* servers are considered in this work. Table 1 describe the configuration of these servers.
- 800 number of physical hosts are considered. Out of 800 physical hosts, 400 hosts are of type *HP* ProLiant *ML110 G4* and rest are of *HP ProLiants*. These servers are virtualized (deploy VMs) to executethe tasks. Table 2 describes the configurations of VMs.
- Real world traces are used to evaluate the performance of proposed algorithm. PlanetLab consist real world workload traces which are taken from March and April 2011. After every 5 minutes *CPU* utilization of all the deployed*VMs* are checked and updated.

In CloudSim toolkit, Static CPU Utilization Threshold(*THR*), Adaptive Median Absolute Deviation (*MAD*), Adaptive Interquartile Range (*IQR*), Local Regression(*LR*) and, Robust local Regression(*LRR*), are *VM* overload detection algorithm.

Minimum Migration time(*MMT*), Maximum correlation (*MC*) and, Random selection (*RS*) are *VM* selection algorithms. By combining the VM selection and overload detection algorithms, fifteen combinations are obtained to evaluate the result. Proposed algorithm is compared with built-in algorithms in CloudSim toolkit.

At the initial stage of simulation, VMs are deployed and randomlyassigned to the two datacenters. After that, overload detection algorithm check that any VM is overloaded or not. If any *VM* is overloaded then migrate it to the other underutilized server. Comparison parameters such as *ESLAV* (product of energy consumption and *SLA* violation) and, Energy consumption are considered to compare the proposed algorithm with the existing built-in CloudSim toolkit algorithms. For calculatedthe extended migration control $CPU_{Thershold}$and $Extendedexecutionboundary_{Theshold}$is used. $CPU_{Thershold}$ for *MMTEMC, RSEMC, and MCEMC* are .40, .85, and .85 respectively. $Extendedexecutionboundary_{Theshold}$is se

Results and Discussion

A migration control threshold parameter is included in the VM selection algorithms. The migration control threshold parameters exclude the VMs which have less than 70% of the execution[22].

- ***Minimum Migration Time with Extended Migration Control (MMTEMC)***

While computing the MMT policies (IQR, LR, LRR) with the proposed MMTEMC,thereduction in ESLAVand energy consumption is noticed as shown in Figs. 1 and 2.

- ***Random Selection with Extended Migration control (RSEMC)***

Table 1. configuration of server HP ProLiant ML110 G4 and HP ProLiant ML110 G5

Server's name	CPU	CORES	RAM	BW Gbit/s	STORAGE GB	MIPS
HP ProLiant ML110 G4	Intel Xenon 3040	2	4096	1	100	1860
HP ProLiant ML110 G5	Intel Xenon 3075	2	4096	1	100	2660

As shown in Fig. 3 and 4 we conclude that IQR_RSEMC and THR_RSEMC policy consume less energy as well as it saves from SLAV than the existing built-in policy.

- ***Maximum correlation with extended migration control (MCEMC)***

Comparison of ESLAV and energy consumption between MC and MCEMC are shown in Fig. 5 and 6.

It is clear from the Figs that the proposed policy of MC gives equal and better reduction in both the comparison parameters (i.e. ESLAV and energy consumption).

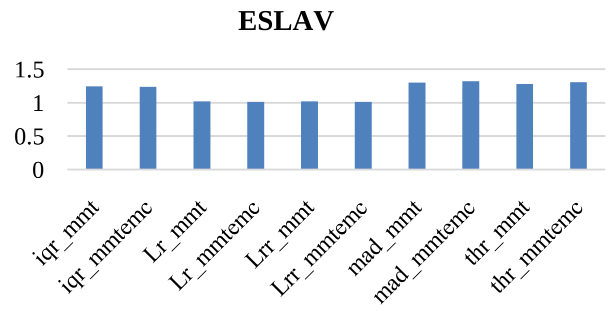

Figure 1. Comparison of ESLAV between MMT and MMTEMC

Table 2. Configuration of VMs

Type of VM	MIPS	RAM	BW	Cores
Type 1	2500	850 MB	1 GB/s	1
Type 2	2000	3750 MB	1 GB/s	1
Type 3	1000	1700 MB	1 GB/s	1
Type 4	500	613 MB	1 GB/s	1

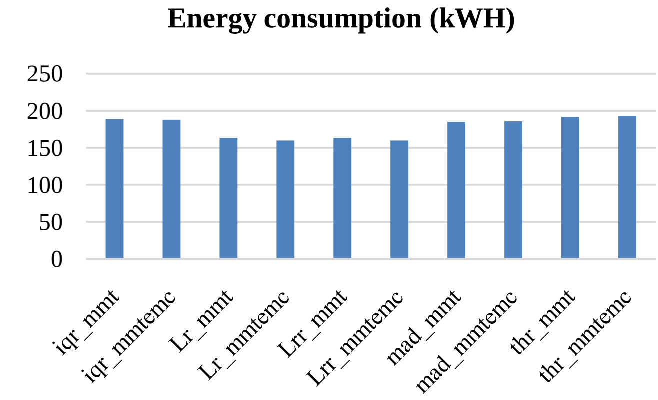

Figure 2. Comparison of energy consumption between MMT and MMTEMC

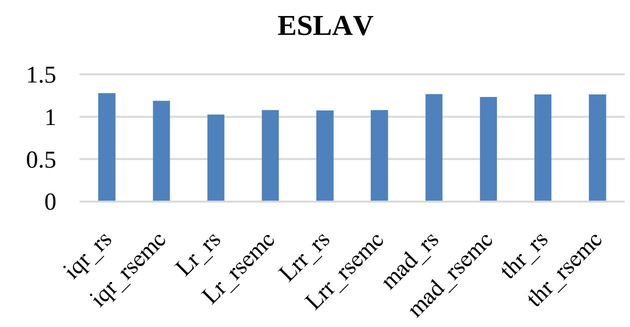

Figure 3. Comparison of ESLAV between RS and RSEMC

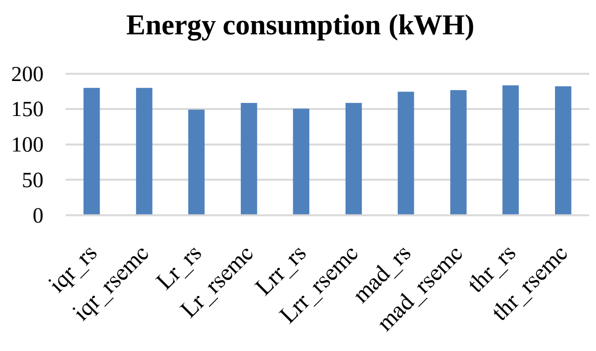

Figure 4. Comparison of Energy Consumption between RS and RSEMC

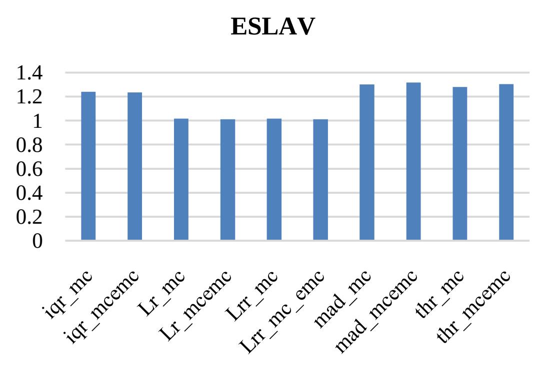

Figure 5. Comparison of ESLAV between MC and MCEMC

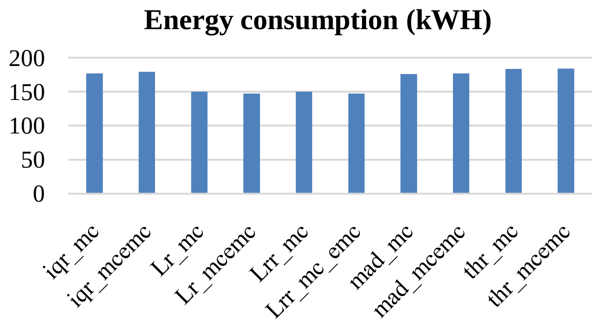

Figure 6. Comparison of Energy Consumption between MC and MCEMC

V. CONCLUSION AND FUTURE SCOPE

Due to increased load on the datacenters their energy consumption as well as chances of SLAV are increased. Datacenter consumes a huge amount of energy. Dynamic VM consolidation is one of the important technique for reducing the SLAV and energy. To reduce the energy consumption and SLAV extended migration control is applied on the VM selection policies by which better results are achieved. Basically, in this paper, VM selection policies are rebuilt using extended migration control. Two thresholds are introduced in migration control. After simulation, we have found that the proposed policies perform very well than the existing policies. A great reduction is noticed in each comparison parameter especially in energy consumption and ESLAV.

VI. REFERENCES

[1] P. Mell, T. Grance, and T. Grance, “The NIST Definition of Cloud Computing Recommendations of the NationalInstitute of Standards and Technology.

[2] Z. Mahmood, “Cloud Computing : Characteristics and Deployment Approaches,” pp. 121–126, 2011.

[3] Y. Ubaidullahbokhari, Mohammad, QahtanMakkiShallal, “Cloud Computing Service Models: A Comparative Study,” pp. 16–18, 2016.

[4] N. Garg and M. S. Goraya, “A Survey on Energy-Aware Scheduling Techniques in Cloud Computing Environment,” *Int. J. Comput. Sci. Inf. Secur.*, vol. 14, no. 10, pp. 523–528, 2016.

[5] J. Thaman and M. Singh, “Improving performance of cloud datacenters using heuristic driven VM migration,” *Proc. 2015 1st Int. Conf. Next Gener. Comput. Technol. NGCT 2015*, no. September, pp. 41–45, 2016.

[6] Y. Li, W. Li, and C. Jiang, “A Survey of Virtual Machine System : Current Technology and Future Trends,” *Int. Symp. Electron. Commer. Secur.*, pp. 332–336, 2010.

[7] W. Cerroni and F. Esposito, “Optimizing Live Migration of Multiple Virtual Machines,” *IEEE Trans. Cloud Comput.*, vol. 7161, no. c, pp. 1–1, 2016.

[8] R. W. Ahmad, A. Gani, S. H. Siti, M. Shiraz, F. Xia, and S. A. Madani, “Virtual machine migration in cloud data centers: a review, taxonomy, and open research issues,” *J. Supercomput.*, vol. 71, no. 7, pp. 2473–2515, 2015R.

[9] J. Wang, W. Bao, and X. Zhu, “Festal: Fault-Tolerant Elastic Scheduling Algorithm For Real-Time Tasks in Virtualized clouds”, IEE Transactions on Computers, vol. 64, no. 9, September 2015.

[10] M. Alaul, H. Monil, R. Qasim, and R. M. Rahman, “Incorporating Migration Control in VM Selection Strategies to Enhance Perfor- mance,” *Int. Strateg. to Enhanc. Perform.*, vol. 6, no. 4, pp. 135–151, 2014.

[11] M. Alaul and H. Monil, “Implementation of Modified Overload Detection Technique with VM Selection Strategies Based on Heuristics and Migration Control,” *Int. Conf. Comput. Inf. Sci.*, pp. 0–4, 2015.

[12] J. J. Prevost, K. Nagothu, B. Kelley, and M. Jamshidi, “Prediction of cloud data center networks loads using stochastic and neural models,” *2011 6th Int. Conf. Syst. Syst. Eng.*, pp. 276–281, 2011.

[13] A. Beloglazov, J. Abawajy, and R. Buyya, “Energy-aware resource allocation heuristics for efficient management of data centers for Cloud computing,” *Futur. Gener. Comput. Syst.*, vol. 28, no. 5, pp. 755–768, 2012.

[14] A. Beloglazov and R. Buyya, “Optimal Online Deterministic Algorithms and Adaptive Heuristics for Energy and Performance Efficient Dynamic Consolidation of Virtual Machines in Cloud Data Centers,” *Concurr. Comput. Pract. Exp.*, vol. 24, no. 13, pp. 1397–1420, 2012.

[15] Z. Cao, “Energy-aware framework for virtual machine consolidation in Cloud computing,” *Int. Conf. High Perform.*

Comput. Commun., p. 429, 2013.

[16] C. Paper, "Energy-aware VM consolidation approach using combination of heuristics and migration control Energy-aware VM Consolidation Approach Using Combination of Heuristics and Migration Control," *Internatioal Conf. Digit. Inf. Manag.*, no. December 2015, pp. 74–79, 2014.

[17] M. Hasan and M. S. Goraya, "Resource Efficient Fault-Tolerant Computing Service Framework in Cloud," *Int. J. Comput. Sci. Eng.*, vol. 9, no. 3, pp. 51–60, 2017.

[18] M. S. Goraya and L. Kaur, "Fault tolerance task execution through cooperative computing in grid," *Parallel Process. Lett.*, vol. 23, no. 1, pp. 1–20, 2013.

[19] M. Hasan and M. S. Goraya, "Priority Based Cooperative Computing in Cloud Using Task Backfilling," *Lect. Notes Softw. Eng.*, vol. 4, no. 3, pp. 229–233, 2016.

[20] K. Park and V. S. Pai, "CoMon : A Mostly-Scalable Monitoring System for PlanetLab," no. August, 2004.

[21] F. De Rose, "Cloudsim : A Toolkit For Modeling And Simulation Of Cloud Computing Environments And Evaluation Of Resource Provisioning Algorithms,"*Softw. Pract. Exp.*, No. August 2010, Pp. 23–50, 2011.

[22] K. Shahzad, A. I. Umer, And B. Nazir, "Reduce Vm Migration In Bandwidth Oversubscribed Cloud Data," Pp. 140–145, 2015.