Energy and SLA based Ant Colony Algorithm for Virtual Machine Consolidation

G.Sushmitha, G.M.Karthik, M. SayeeKumar

Abstract – Cloud data centers provide virtual resources as per user's demand. Virtual Machine Consolidation is the process of migrating virtual machines from heavily loaded and lightly loaded hosts to moderately loaded hosts to increase energy efficiency. But this can impact the SLA between the cloud service provider and customer because migration incurs additional energy consumption. Energy consumption and SLA violation are major concerns of a cloud service provider. To solve this problem, a combined metric which is the product of energy consumption and Performance degradation due to Migration (PDM) is used as heuristic information for Ant Colony Optimization. The virtual machine to be migrated is chosen based on Virtual Machine (VM) Selection policies which consider minimal migration time or minimal product of computing and memory resources. The performance evaluation shows the efficiency of the algorithm.

Keywords – (Data Centers, Cloud Computing , Virtual Machine Placement,Ant Colony Optimisation ,Energy consumption)

I. INTRODUCTION

Cloud Computing is the provisioning of virtual resources to the consumer. The virtual machine requests of the consumer are hosted on of the physical machines of the data centers. The initial placement technique is referred to as the static virtual machine placement. The consumption of energy in datacenters of United States was around 70 billion Kilowatt-hours in 2014.The world-wide data center energy consumption is rising every year due to more services offered by the data centers. The power usage effectiveness of Oregon's datacenter is nearly 1.08.This signifies that the energy consumption is mainly due to CPU utilization. However for the purpose of minimizing energy consumption and load balancing the dynamic virtual machine placement came into place. Unnecessary energy consumption can be minimized by efficient resource utilization thereby avoiding performance degradation, The count of active servers can be minimized when the VMs are migrated from overloaded physical machine to lightly loaded physical machine .The virtual machines in the physical machines with meager workload must be migrated. After which the physical machine can be put to power saving modes. There are millions of virtual machine requests coming in and they have to be placed in the physical machine and the virtual machines have to be placed in such a way it reduces energy consumption and SLA violations. Dynamic consolidation of VMs includes categorizing the hosts, into high, medium and lightly loaded hosts.

Revised Manuscript Received on April 30, 2019.

Sushmitha Govindaraju, SRM Institute of Science and Technology, Chennai (Tamil Nadu), India.

G.M.Karthik, SRM Institute of Science and Technology, Chennai (Tamil Nadu), India.

M.Sayeekumar, SRM Institute of Science and Technology, Chennai (Tamil Nadu), India.

Then the virtual machine that has to be migrated to a destination host has to be chosen. The destination host can be a new host or host that is neither overloaded nor under loaded. Migration also affects the energy efficiency. The negotiation between the cloud service provider and the customer regarding the service requirements and penalties that will be incurred if they are not met is referred to as the Service Level Agreement. The service unavailability could be due to 100 % CPU utilization or virtual machine state transfer during migration time. If a host server fails, and then applications hosted in the virtual machines would be down. Genetic Algorithms, Ant colony optimization algorithms, particle swarm optimization, bin packing and First Fit Decreasing (FFD)algorithms are employed. They differ by the objectives considered and models and metrics used to support the placement. Evolutionary algorithms aim to reduce the count of active Physical Machines (PM) and focus on network related parameters[5],[17]. Some Virtual Machine Placement (VMP)algorithms consider reducing energy consumption [25],[29],[12],[23] and SLA violation as objectives[6].Some VMP approaches concentrate on reducing SLA violation and power consumption[11].Network performance bottlenecks could cause SLA violation resulting in the failure to meet the Quality of Service (QoS) constraints. Some VMP techniques focus on network bandwidth for effective communication between physical servers during migration [21].There are techniques that consider the association between the virtual machines and performs migration based on topology [24].

II. RELATED WORKS

Several VMP algorithms consider objectives either from cloud service provider (CSP) perspective or the client's perspective. Some VMPs focus on both objectives from CSP and clients' perspective .From the CSP perspective, cloud resources such as energy, network, power, and cost are optimized. From the client's perspective, SLA performance degradation and QoS are considered. Our main focus is on minimizing energy consumption, SLA violations and reducing migration time. CompVM is a VMP scheme that has reducing the count of active PMs and migrations as objectives [13], [8].Wrasse is a static provisioning scheme and focuses on resource allocation[4], [11].Cloud Scale is a dynamic provisioning scheme that predicts resource demand and focus on saving energy [25].

The aim of GACOVMP is to cut down the number migrations and uses energy related metrics as heuristic information[15].ATEA is a dynamic provisioning scheme that categorizes hosts into four classes and migrates VMs from over loaded hosts to under loaded hosts [12], [26]. ECS and ECS + idle are heuristic based algorithms whose focus is on energy

consumption but do not consider SLA violation [24].DVFS•

Performance Degradation due to Migration

policy was proposed to minimize power consumption. The processor voltage is minimized when the work load is less and vice versa [9].Dynamic slack algorithm was proposed to minimize energy consumption but does not focus on SLA violation [10]. Single threshold algorithm focuses on minimizing energy consumption and SLA violations but the During migration the performance of VM is degraded because transferring the VM state to destination requires suspension of the VM.

M - Count of virtual machines. R - CPU requirement of the VM.

C - 10% utilization of the virtual machine during migration.

SLAs are not at lower levels [16].Double Threshold algorithm

proposes two thresholds and focuses on energy consumption•

PDM = 1/M ($\sum^{n}$ $E = \sum$

C_i/R_i *)*

and SLA violation but it does not effectively reduce them [2]. Adaptive threshold based algorithm is proposed to improve energy efficiency but the energy consumption is not effectively reduced[1]. MIMT is an energy aware algorithm that is based

The objective is to reduce both energy consumption and SLA violation metric hence the combined metric can be deduced as follows.

*EPDM = E * PDM*

on determined thresholds hence they do not perform well for

C. Host Load Detection Policy

dynamic workloads[27].Fault tolerance mechanisms have been have been proposed for Zen hypervisors[3]. Check pointing mechanisms have been proposed for kernel level [17].Failure detection and recovery methods are discussed[19].The proposed ant colony optimization (ACO)algorithm uses SLA and energy metrics as heuristic information. It also discusses two VM selection policies based on minimum migration time and minimum computing resources. The proposed algorithm produces novel results.

III. MODELING

The energy consumption of servers needs to be determined by the model explained below. A combined metric for tackling energy consumption and SLA violation used as heuristic information is detailed in this section .The host load detection The concept of Interquartile Range *(IQR)* can be used to set the CPU threshold .The high ,medium and low cpu thresholds are represented by HL, ML, LL. If the CPU usage crosses the HL threshold then the host is overloaded and the VMs have to be migrated either to under loaded host or new host. If the CPU usage is greater than or equal to ML and less than the HL threshold then it is a medium loaded host. If the CPU usage is greater than LL and less than ML then it is a lightly loaded host.

HL = 1– s*(IQR)

ML = (0.8)(1 – s*(IQR))

LL = (0.4)(1 – s*(IQR)) IQR = Q3 – Q1;

s is the consolidation parameter.

policy which determines the CPU usage of every host and two*D. VM Selection Policy*

VM selection policies which select VM for migration are also discussed.

A. Energy Consumption Model

Energy consumption is formulated as the aggregate of energy consumed by every physical machine. The model proposed in [7] is used this work.

VM Selection policies determine the VM that will be chosen for migration from a heavily loaded host to medium or lightly loaded host. Two VM selection policies are discussed here. They are as follows.

Least Migration Time Policy (LMT)

The VM to be selected for migration in case of server consolidation is based on the minimum migration time policy.

Energy and PDM Metric

$j=1$

$((Maxp_j - Minp_j) * Ucpu_j + Minp_j) * X_j(1)$

The migration time is estimated as the(1r)atio of the memory

$Maxp_j$ denotes the peak power consumption of the PM j.$M\ in\ p_j$ denotes the power consumption when the PM j is idle.$Ucpu_j$ denotes the CPU utilization of PM j.X_j denotes the state of PM. If it is active then its value is 1 and if it is in sleep or shutdown state its value is 0.

B. Metrics for Evaluating a PM

The cloud data centers have to meet the QoS requirements of the consumer. In worst cases when the host CPU is overloaded up to 100% or during the migration of VMs, performance is degraded leading to increase in response time and reduced throughput. Performance Degradation due to Migration (PDM) metric is discussed here.

utilization and the network bandwidth. During migration memory pages of the VM in the source PM are transferred to destination PM, with the available network bandwidth. The virtual machine which has the least migration time when compared with another virtual machine in the host is chosen for migration.

Minimum Migration time = M (s)/B (s) <= M (a)/B(a)(2)

s∈ S, a ∈ S,S is the virtual machines hosted in the PM.M,B denotes memory and bandwidth utilization respectively

Least memory size and CPU Utilization policy(LMC)

The virtual machine with the least product of memory size and CPU utilization is selected for virtual machine.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication

M,B denotes memory and bandwidth utilization respectively vm ∈ S, t∈ S, S is the virtual machines hosted in the PM.

LMC = M (vm)*C (vm) <= M(t)*C (t) (3)

IV. ENERGY AND SLA BASEDSERVER CONSOLIDATION

A. Ant Colony Optimization:

ACO is a widely used meta heuristic approach that was formulated by observing ant colonies. Ants live in colonies and they search for food sources. During the search the ants leave a substance on their path from their colony to the food source. This is called pheromone trail. There can be several paths from the colony to the food source. The pheromone trail depends on the quality and quantity of food. Gradually pheromone trail on the shortest path becomes thicker because ants were able to quickly navigate between source and colony. The shortest route can be considered as heuristic information.ACO is predominantly used for Travelling Salesman problem (TSP),static placement of VMs and in networks to find the best path for transmission of data.

B. Initialisation & State Transfer Probalility

The energy consumption E parameter is used as pheromone trails for finding a physical host. The energy consumption is used to initialize the pheromone values initially.

$\tau e = 1/E(S0)$ (4)

A random number q is generated and lies between 0 and 1.If q is less than q0 then we formulate the probability based on the below equation.

$Pb(i,j) = \tau(i,j)\alpha h(i,j)\beta$ (5)

If q is more than q0 then the following equation is used to formulate the probability With the below equation.

$Pr(i,j) = \tau(i,j)\alpha h(i,j)\beta / \sum s \in PM\ \tau(i,s)\alpha h(i,s)\beta$ (6)

Pheromone Updation & Heuristic Information

Local pheromone update is performed after every virtual machine of the request is placed on the host by the ant.

$\tau e(t) = (1-\mu l)\tau e(t-1) + (\mu l)\tau e$ (7) Global pheromone update is performed only when every ant has computed solution and is used to enhance the learning capability of ants. The global update is performed on each solution.

$\tau e(t) = (1-\mu g)\tau e(t-1) + (\mu g)/E(S)$ (8)

Heuristic information is used along with pheromone trail for better optimization. The performance variation when The EPDM metric can be used as heuristic information.

$h(i,j) = 1/EPDM$

TABLE- I: Description for Symbols

Notation	Description
E	Energy Consumption
$Maxpj$	Peak Power Consumption of PM j
$Minpj$	Minimum Power Consumption of PM j
$Ucpuj$	CPU Utilization of PM j
Xj	State of PM j
Ri	Requested amount of CPU by VM i
Ci	CPU utilization during migration
s	Consolidation parameter
M(i)	Memory Utilization of virtual machine i
B(i)	Bandwidth of virtual machine i
C(i)	CPU Utilization of i
$\tau e(t)$	Pheromone trail
q	Random Number
q0,	Predefined parameter
Pb	Probability if q is less than q0
Pr	Probability if q is great than q0
μl	Local Pheromone Update
μg	Global Pheromone Update
h(i,j)	Heuristic information for placing VM i on PM j
α	Parameter to control the impact of pheromone trail
β	Parameter to control the impact of heuristic information

C. Algorithms

The process flow of the proposed algorithm has been depicted in “Fig.1”.It consists of allocation of virtual machines by FFD to servers ,followed by load detection and finally dynamic allocation of virtual machines based on load on the servers. Static Virtual Machine Placement (SVMP):FFD heuristic is used for static placement. The virtual machines are sorted in decreasing order and placed in the physical machines. The initial solution is calculated for every request and the pheromone trails are initialized accordingly. Host Load Detection (HLD): The hosts are categorized based on thresholds values and the virtual machines from the heavily loaded hosts are migrated to hosts with less load.Dynamic Virtual Machine Placement (DVMP):The virtual machines will be placed in the hosts based on the resource utilization ensuring that the placement did not cause any resource contention. Each ant k constructs solution .Local pheromone update is performed. Only when every ant has constructed solution global pheromone update is performed. In the end the required solution is obtained. By the ant colony optimization algorithm, the global and local update would help in finding the optimal solution.

HOST LOAD DETECTION ALGORITHM (HLD) :

HLD categories the host based on the CPU workload as shown in “Fig.3”. The CPU workload of n values is represented in the form of quartiles. The difference between the first and third quartile is used for determining the three threshold values HL, ML and LL.

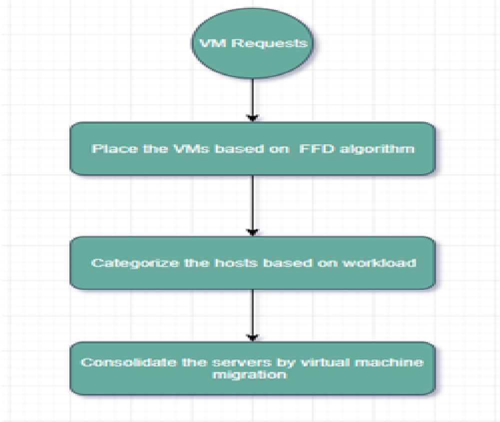

Fig. 1.System Workflow

SVMP ALGORITHM

FFD algorithm is used for static virtual placement. The FFD algorithm ranks the VMs in the decreasing order of their CPU utilizations and allocates them to the PMs as depicted in "Fig.2".

Input: VMlist, PMlist

Output: Allocation of VM to PMs

1: Rank the VMs in descending order based on their CPU utilizations.

2: for every VM v_i in VMlist

3: for every PM pm_j in PMlist

4: if $(mips_{pm} - mips_v >= 0)$

5: allocate host for VM and remove VM from VMlist

6: end if

7: end for

8: end for

Fig.2. SVMP Algorithm

Input: HL, ML, LL, N

Output: hlist, mlist, llist

Initialize three sets hlist, mlist, and llist

1: for each host j=1 to N

2: Let the CPU utilization of the host be c.

3: Calculate the Euclidean distance hc, mc, lc between

CPU utilization of the host and the three thresholds.

4: The least of the three distances signifies the list to which the host will be added.

5: If hc is minimum then host will be added to hlist.

6: Else if mc is minimum then host will be added to mlist.

7: Else host will be added to llist.

8: end for

Fig.3. Host Load Detection Algorithm

VM SELECTION ALGORITHM

The algorithm gives an option to choose one of the VM selection policy. Then based on the policy the virtual machine is selected by migration time or the product of CPU and memory utilization as shown in "Fig.4".

Input: VMs (v_c, v_m, v_n),PM,policy

Output: VM

1: Choose the VM selection policy

2: if (policy == 'LMT') then

3: for each VM j=1 to n do

4: Calculate the migration time by "(2)"

5: end for

6: Choose the VM with least migration time

7: end if

8: if (policy == 'LPMC') then

9: for each VM j=1 to n do

10: Deduce LMC by "(3)"

11: end for

12: Chose the VM with least value of PMC.

13: Return the VM selected by the selection policy

Fig.4. VM Selection Algorithm

DVMP ALGORITHM

DVMP The virtual machines from the heavily loaded hosts are migrated to the lightly loaded hosts for server consolidation. As shown in "Fig.5" three list of hosts differing by their workload are given as input to the algorithm. The resource capacity of the medium loaded hosts calculated. For each physical machine in lightly loaded and heavily loaded hosts, the virtual machines are selected for migration to another host based on VM selection algorithm and are added to list vmlist. For every virtual machine, the hosts that have the required resources to run the virtual machine are added to capable hosts. q is a random number which is compared with a fixed parameter

□0.There are two methods to compute probability based on the comparison. The probability for placing a virtual machine on a server is calculated with the help of pheromone and heuristic information of the algorithm. The heuristic information is determined by energy and PDM metric .The physical machine with the maximum probability is chosen .Once the new host has been chosen, release the virtual machine from the original host and allocate the resources in the new host to the virtual machine. Once the vm is allocated local pheromone is updated. After placing all the virtual machines in the physical machine global pheromone update is performed. This algorithm is investigated with two scenarios that is by creating a host failure and by requesting additional resources for the virtual machine. The algorithm also deals with varying workloads. For varying workloads interquartile range is chosen as host load detection method. For host failure and varying workloads choosing the virtual machine and migration of virtual machine is by VM selection policies and the DVMP algorithm.

DVMP Algorithm

Input: PMs, VMs

Output: Migration of virtual machines from overloaded hosts to under loaded and medium loaded hosts.

Input: hlist, mlist, llist

1: Initialize set of variables: anum, n, max, μ_l, μ_g, α, β.

2: Initialize optimal solution set S_p as empty and N;

3: for each PM in mlist

4: Calculate available resource capacity *based on* $(M_r^j - M_{r,m}^j)$ and add the hosts to hostlist.

5: end for

6: for each PM in hlist && llist do

7: Determine the virtual machines based on the VM selection algorithm and add them to list vmlist

8: end for

9: while n < max

10: for each ant k do

21: Release the VM from existing host and migrate the VM to PM with maximum probability.

22: else

23: Calculate the probability of placing the PM on the VM by "(6)" for every PM in capable hosts

24: Release the VM from existing host and migrate the VM to any of the PMs of the capable hosts randomly.

25: Local update is performed by "(7)"

26: end while

27: end for

28: for each solution do

29: Global update is performed by "(8)"

30: end for

31: n + +

32: end while

33: Return optimal solution

11: // ants generate solution

12: while vmlist contains VMs do

13: Get a VM v_i in vmlist;

//Let v_c, v_m, v_n be the requested resource by the virtual machine.

14: for each PM in hostlist do Estimate the available resource (p_c, p_m, p_n)

15: if ($v_c < p_c$ and $v_m < p_c$ and $v_n < p_n$)

16: add list to capable hosts

17: end for

18: Determine a random number q.

19: if ($q < q_0$)

20: Calculate the probability of placing the PM on the VM by "(5)" for every PM in capable hosts

Fig.5. DVMP Algorithm

V. PERFORMANCE EVALUATION

In this experiment we will be using cloudsim simulator to create an artificial cloud setup in which we create VMs and PMs and place according to VM allocation policy. We create a VMCreation.Java file in the default package and we modify the VMAllocationPolicy file in the org.cloudbus.cloudsim package. We add functions for calculating CPU utilization, energy consumption, probability ,probability ratio, detecting host load, selecting virtual machines for migration, ant colony based function for server consolidation, FFD based algorithm for static allocation. We get the input for the number of virtual machines and once the user enters the value, virtual machines are created. The resource requirements of the virtual machines are not specified by the user, we use a random number generator based function that generates the cpu and memory requests randomly. The FFD based algorithm sorts virtual machines in their decreasing order and allocates the virtual machines to their hosts. The host will be allotted virtual machines till their cpu utilization is 80 percent. Beyond that the host might get overloaded and result in SLA violations. We also let the servers be packed with virtual machines, and we detect the various loads of host by interquartile range method and migrate the virtual machines from heavily loaded and lightly loaded to mediumly loaded hosts. This would be suitable for hosts with varying workloads.

The energy consumption of the hosts is displayed in the console. Then the user can choose between two options. The first option is to generate a host failure and the second option is to place additional resources request on behalf of a virtual machine. The first option causes a host failure. Any host can be selected through random number generator based function and

the hosts status is set to failed. The host is removed from the hosts that are in the datacenter. The virtual machines in the host are migrated to other hosts. It is ensured that the destination hosts do not get overloaded through migration. And we migrate only when there is a failure. The second option is the virtual machine requesting for additional resources. The user is asked to input the virtual machine that needs additional resources and also the memory and cpu utilization required by the virtual machine. The cpu utilsation of the host is calculated. If the utilization does not go beyond eighty percent after calculation of cpu utilization by adding the requested resources we simply provide the resources. In case the utilization exceeds eighty percent then we scale the resources of the host and then we provide the requested resources of the virtual machine. Thus we investigate for varying workloads, host failure and also for additional resource request.

5.1.1 CONFIGURATION

The CPU and memory utilizations of VM are created by rand based function. Every host in the datacenter is configured with mips, memory, bandwidth, storage.

TABLE- II: Resource of Physical Machine

IPS	MEMORY	BANDWIDTH	STORAGE
10000	100000	100000	1000000

5.1.2 RESULTS

The proposed algorithm produces novel results when compared with CompVM. Table. III gives an account of the energy consumption for proposed and an existing algorithm. The From “Fig.6”, it can be seen that DACOVMP outperforms CompVM. The energy consumption of the datacenter is less compared to another algorithm because the cpu utilization is not beyond 80 percent which helps in mitigating energy consumption. We also plot graph for the table of values. Table. IV details about number of migrations for different numbers of VMs for the methods discussed above. From “Fig.7”, it can be seen that DACOVMP outperforms CompVM because of using EPDM metric as heuristic information. We also plot graph for the table of values. The number of migrations is less because we migrate only in case of host failure and scale if the virtual machines require more resources.

TABLE-III: Energy Consumption with virtual machines

No Of VMs	DACOVMP	CompVM
500	150	360
1000	270	455
1500	320	570
2000	420	630
2500	550	715
3000	670	790

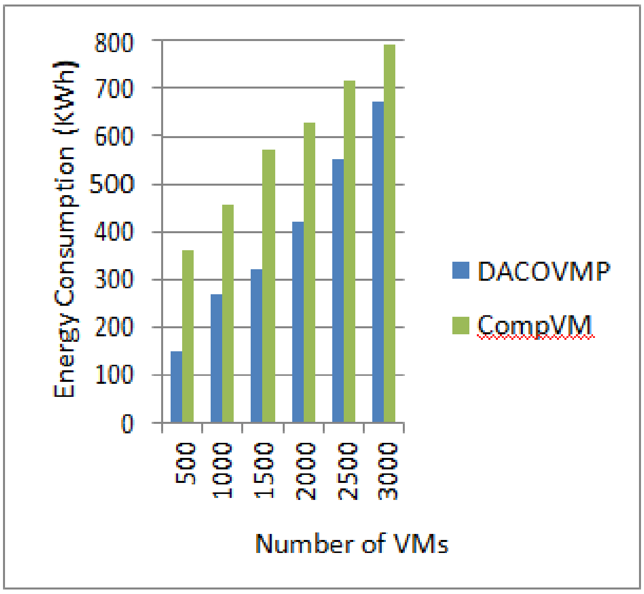

Fig.6. Energy consumption for number of virtual machines

TABLE- IV: No of Migrations for no of virtual machines

No Of VMs	DACOVMP	CompVM
500	25	42
1000	40	65
1500	63	105
2000	89	145
2500	110	170
3000	140	230

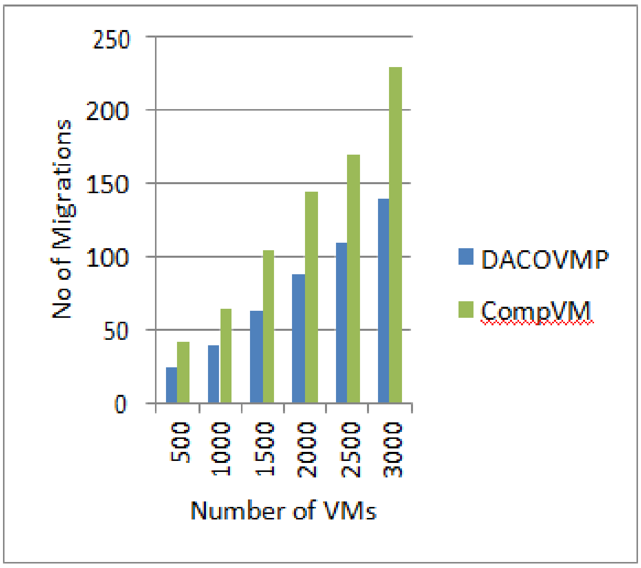

Fig.7. No of migrations for number of virtual machines

Most of the applications have a similar workload hence the CPU utilization would not change. Hence in the future we can predict such workloads and can allocate the resources to them. It has been observed that virtual machine resources are not utilized fully this is because they do not have an idea of how much resource they will require and also are cautious of overload in case the resources are not available. So they endup consuming more resources than they need. In such cases in a server if there are virtual machines that are not using their resources and the utilization of resources are not going to increase for some time then we can allocate the resources of the virtual machines to another virtual machine or just free up the resources so that the host utilization at least will go down. Over commitment and prediction based consolidation can be explored further for virtual machine consolidation.

VI. CONCLUSION

In this paper the proposed algorithm focuses on minimizing SLA violation and energy consumption. It uses a combined metric which consists of energy and PDM metric as heuristic information. The algorithm has produced novel results of reducing energy consumption and SLA violation.PDM metric is used for mitigating SLA it could also be combined with other SLA metrics for better results. But this is experimented only with homogeneous physical machines. The work can be extended to include security and financial objectives that impact business. The work can be extended to focus on thermal, air management and storage objectives. Over commitment and prediction based consolidation can be explored further for virtual machine consolidation.

REFERENCES

1. K. Venkatesh and V. Nagarajan, A framework to support dynamic routing and traffic provisioning in optical networks using virtualization and software defined networking, International Journal of Pure and Applied Mathematics, vol. 118 no. 11, pp.733-737, 2018
2. L. Chen, and H. Shen, Consolidating complementary VMs with spatial/temporal-awareness in cloud datacenters. IEEE International Conference on Computer Communications (INFOCOM), pp.1033-1041. 2014
3. M. SayeeKumar and T. Purusothaman, Hybrid broadcast group management protocol for secure, scalable and efficient group communication, Journal of Computer Science, vol. 11, no. 2, pp.344-350, 2015
4. Q. Zhang, H. Wang, F. Zhu, S. Yi, K. Feng, and L. Zhai, Energy-aware VM Placement with Periodical Dynamic Demands in Cloud Datacenters,
5. IEEE 19th Int. Conf. High Performance Computing and Communications, pp.162-169, 2017.
6. R. Buyya, R. Ranjan, and R. N. Calheiros, "Modeling and simulation of scalable cloud computing environments and the CloudSimtoolkit:challengesandopportunities,"inProceedingsoftheInterna tionalConferenceonHighPerformanceComputing and Simulation (HPCS '09), pp. 1–11, Leipzig, Germany, June 2009.
7. A. Beloglazov and R. Buyya, "Adaptive threshold-based approach for energy-efficient consolidation of virtual machines in cloud data centers," in[17] R. Gioiosa, J. C. Sancho, S. Jiang, and F. Petrini. Transparent, Proceedings of the ACM 8th International Workshop on Middleware for Grids, Clouds and e-Science (MGC '10), pp. 1–6, Bangalore, India, December 2010.
8. incremental checkpointing at kernel level: A foundation for fault tolerance for parallel computers. In IEEE/ACM International Conference on High Performance Computing and Networking (SC), page 9, 2005. URL http://hpc.pnl.gov/people/ fabrizio/papers/sc05.pdf
9. A. Beloglazov and R. Buyya, "Energy efficient resource management in
10. virtualized cloud datacenters,"inProceedingsofthe 10thIEEE/ACMInternationalSymposiumon Cluster,Cloud,and Grid Computing (CCGrid '10), pp. 826–831, IEEE, Melbourne, Australia,May2010.
11. A. B. Nagarajan, F. Mueller, C. Engelmann,and S. L. Scott, Proactive fault tolerance for HPC with Xen virtualization, In Proceedings of the 21st annual international conference on Supercomputing(ICS07), pp. 23-32, 2007
12. A. Rai, R. Bhagwan,and S. Guha, "Generalized resource allocation for the cloud." in Proc. of SOCC, 2012
13. C. Liu, C. Shen, S. Li, and S. Wang, A new evolutionary multiobjective algorithm to virtual machine placement in virtualized data center, Proc. 5th IEEE Int. Conf. Softw. Eng. Service Sci., pp. 272–275, 2014
14. D. Dong and J. Herbert, Energy efficient VM placement supported by data analytic service, Proc. 13th IEEE/ACM Int. Symp. Cluster, Cloud, Grid Comput., pp.648–655, 2013
15. F. López Pires, E. Melgarejo, and B. Barán, "Virtual Machine Placement. A Multi-Objective Approach," XXXIX Latin American Computing Conference (CLEI 2013),2013
16. G.M. Karthik, M. Sayee Kumar, and A. Aruna, Fully Homomorphic Public Key Encryption Based on Arbitrary Key Aggregation Cryptosystem (AKAC) for Secured Data Communication in Cloud Infrastructure, Int. J. Pure and Applied Mathematics, vol. 117, no. 15, pp.1143-1149, 2017
17. H. Hanson, S. W. Keckler, S. Ghiasi, K. Rajamani, F. Rawson, and J.Rubio,"Thermal response to DVFS :analysis with an Intel Pentium M," in Proceedings of the International Symposium on Low Power Electronics and Design (ISLPED '07), pp. 219–224, August2007.
18. J.Kang and S.Ranka,"Dynamic slack allocation algorithms for energy minimization on parallel machines," Journal of Parallel andDistributedComputing,vol.70,no.5,pp.417–430,2010
19. K. Venkatesh, L. N. B. Srinivas, Mukesh Krishnan , and Shanthini, QoS improvisation of delay sensitive communication using SDN based multipath routing or medial applications, Future Generation Computer Systems, pp. 256-265, 2018.
20. R. Wang, J.A. Wickboldt, R.P. Esteves, L. Shi, B. Jennings, and L.Z. Granville, Using empirical estimates of effective bandwidth in network- aware placement of virtual machines in datacenters, IEEE
21. Trans. Netw. Service Manag., vol. 13, no. 2, pp.267–280, June 2016.
22. S. Rani, C. Leangsuksun, A. Tikotekar, V. Rampure, and S. Scott. Toward efficient failre detection and recovery in hpc. In HighAvailabilityandPerformanceComputing Workshop, page (accepted), 2006.
23. S. Wang, Z. Liu, Z. Zheng, Q. Sun, and F. Yang, Particle swarm optimization for energy-aware virtual machine placement optimization in virtualized data centers, Proc. Int. Conf. Parallel Distrib. Syst., pp. 102–109, 2013
24. V. Shrivastava, P. Zerfos, L . Kang-won, H. Jamjoom, L. Yew-Huey, and S. Banerjee, "Application-aware virtual machine migration in Data Centers," In Proceedings of 2011 IEEE INFOCOM, pp. 66-70,
25. April 1015, 2011.
26. X. Dai, J.M. Wang, and B. Benasou, Energy-efficient virtual machines scheduling in multi-tenant data centers, IEEE Trans. Cloud Comput., vol. 4, no. 2, pp.210–221, June 2016.
27. Y. Ajiro and A. Tanaka, Improving packing algorithms for server consolidation, Proc. 33rd Int. Comput. Meas. Group Conf., pp. 399–406, 2007.
28. Y. C. Lee and A. Y. Zomaya, "Energy conscious scheduling for distributed computing systems under different operating conditions," IEEE Transactions on Parallel and Distributed Systems,vol.22,no.8,pp.1374–1381,2011.
29. Z. Shen, S. Subbiah, X. Gu, and J. Wilkes, " Elastic resource scaling for multi-tenant cloud systems." in Proc. of SOCC, 2011.
30. Z. Zhou, Z. Hu, and K. Li, Virtual machine placement algorithm for both energy-awareness and SLA violation reduction in cloud data centers, Sci. Program., vol. 2016, pp.1–11,2016.
31. Z. Zhou, Z-G.Hu, T.Song, and J.-Y. Yu, "A novel virtual machine deployment algorithm with energy efficiency in cloud computing," Journal of Central South University, vol. 22, no. 3, pp.974–983,2015.

AUTHORS PROFILE

Sushmitha Govindaraju completed B.Tech from K.C.G College of Technology and M.Tech(Cloud Computing) from SRM IST Kattangulathur,Chennai in the years 2015 and 2019 respectively.

G.M.Karthik completed B.E(CSE) from Madurai Kamaraj University and M.E (CSE) from Anna University Chennai, in the years 2003 and 2005 respectively. He Completed his Ph.D . from Anna University, Chennai in the year 2015. He has track record of 10+ conferences and 18+ Journal publication. He has been a reviver for International Journals. His area of interest includes Data Mining, Data Analytics,time Series Data Mining, Cloud Computing, Machine Learning, IoT and Cognitive Radio.

M.Sayeekumar completed B.E(CSE) from Madras University and M.E (CSE) from Government College of Technology, Coimbatore, in the years 2001 and 2008 respectively. He Completed his Ph.D . from Anna University, Chennai in the year 2015. He has track record of 40+ conferences and 20+ Journal publication. He has been a reviver for International Journals. His area of interest includes Networks, Network Security, Cloud Computing , Machine Learning, IoT and CognitiveRadio.