

Modified Salp Swarm Algorithm based Energy-Efficient Resource Allocation in Cloud-Computing Data Centers

Rajesh P. Patel, Harshad B. Bhadka



Abstract: In a cloud data centre the consolidation of the virtual machines (VMs) assist to optimize the resources need and diminish the energy consumption. In the consolidation of the VMs the VM placement acts an important role. By considering optimized energy consumption the researchers have developed various algorithms for VM placement. However, these algorithms be deficient in the exploitation mechanism use resourcefully. This paper attend to VM placement issues by offering meta-heuristic algorithms that is, the Modified Salp Swarm Algorithm (MSSA) presenting the comparative analysis relating to energy optimization. The comparison are made adjacent to the existing particle swarm optimization (PSO), and salp swarm algorithm (SSA) and the energy consumption results of all the contributing algorithms confirm that the proposed MSSA is more efficient than the other algorithms. The simulation result demonstrates that MSSA outperforms effectively than other presented approaches in optimal VM placement in cloud computing environment with maximal resource use, minimal energy consumption, minimum SLA violation and reduced migration cost.

Keywords: Energy efficient, virtual machine placement, migration, dynamic resource allocation, cloud computing, data centers.

I. INTRODUCTION

The most recent emerging technology is cloud computing in which application and resources are shared and safely managed in cost effective way. In the course of last few years, because of the parallel computing paradigms and evolving data centers the cloud computing services have become progressively popular [1-3]. There are three noteworthy sorts of services gave by cloud are IaaS (Infrastructure-as-a-Service), SaaS (Software-as-a-Services) and PaaS (Platform-as-a-Service) [4, 5]. The four fundamental organization models in cloud computing are public, private, community and hybrid clouds [6]. To convey cloud computing services numerous computing services providers including Yahoo, Microsoft, IBM and Google are quickly sending data centers in various locations [7, 8]. With a specific end goal to increase high efficiency and save power through expansions of IT the cloud computing has walked to the IT business.

The cloud computing worldwide uptake has subsequently driven remarkable increments in datacenter power consumption. By the datacenters thousands of interconnected servers are composed and worked to give different cloud services [9]. Energy consumption has become very serious as development of cloud technology and creation of large number of data centres. The performance and efficiency of data center can be articulated in terms of total of abounding electrical energy [10]. To operate the data centers need a too much amount of energy. At lower than 50% CPU utilization most of the host in data center operates. This suggests that both energy efficiency and high performance of the data centers are important problems. The quality of service and energy consumption is included by energy efficiency [11]. With the aim to diminish the energy exploitation many organization for example, Green Grid has recently addressed the data center energy consumption problems. For managing the energy consumption the virtual machine relocation and server consolidation methodology present a new approach in data centers [12].

The objective of this work is to get better the energy effectiveness and surety of SLA by presenting a method which is based on heuristic knowledge for the allocation of resources dynamically by migrating the VMs from overloaded host sequentially to guarantee the response time and deadline of the work. The rest of the paper is prepared as follows. A review of the related work is presented in Section 2. Section 3 described our cloud architecture and proposed methodology. In part 4, we design a heuristic method by formulating the MSSA and its fitness function in order to relocate VMs in overload host. In Section 5, we evaluate our approach and analysis of simulation result. Finally Section 6 concludes the paper.

II. RELATED WORK: A BRIEF REVIEW

To solve the high energy consumption dilemma, an energy-efficient virtual machine consolidation prediction-based VM deployment algorithm for energy efficiency (PVDE) was presented by Zhou et.al [13]. For the classification of servers in the data center the linear weighted method was utilized and predicts the host load. They performed high performance analysis. In their work, the algorithm lessens the energy consumption and control SLA violation when compared to other energy saving algorithms in the experimental result. Li et.al [14] presented an elaborate thermal model to address the complexity of energy and thermal modeling that investigate the temperature distribution of airflow and server CPU.

Revised Manuscript Received on October 30, 2019.

* Correspondence Author

Rajesh Patel*, Research Scholar, C. U. Shah University, Wadhwan
rpp.kirc@gmail.com

Dr. Harshad Bhadka, Dean and Associate Professor, Faculty of
Computer Science, C. U. Shah University, Wadhwan,
harshad.bhadka@yahoo.com

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC-BY-NC-ND license <http://creativecommons.org/licenses/by-nc-nd/4.0/>

To minimizing the total datacenter energy consumption the author presented GRANITE - a holistic virtual machine scheduling algorithm. The algorithm was expressed with respect to other existing workload scheduling algorithm IQR, TASA and MaxUtil and Random using real cloud workload characteristics take out from Google datacenter trace log. The GRANITE consumes less total energy and reduces the critical temperature probability when compared with existing that demonstrated in result.

A new scheduling approach named Pre Ant Policy was introduced by Hancong Duan et.al [15]. Based on fractal mathematics their method consists of forecast model and on the basis of enhanced ant colony algorithm (ABC) a scheduler. To activate the implementation of the scheduler by virtue of load trend prediction was determined by prediction model and under the premise of guaranteeing the quality-of-service, for resource scheduling the scheduler is responsible while maintaining energy consumption. The performance results demonstrate that their approach present resource utilization and excellent energy efficiency. In sequence to elevate the exchange between energy consumption and application performance Rossi et.al [16] presented an automated coordination of different energy savings techniques. They implemented Energy-Efficient Cloud Orchestrator-e-eco and by using enlarging applications on a dynamic cloud in a real environment the infrastructure test were carried out to evaluate e-eco. Their evaluation result demonstrates that e-eco was able to reduce the energy consumption.

In cloud computing for energy saving, a three dimensional virtual resource scheduling method (TVRSM) was introduced by Zhu et.al [17]. For the cloud data center they build the resource and dynamic power model of the PM in their work. There are three stages of virtual resource scheduling process as follows; virtual resource optimization, virtual resource scheduling and virtual resource allocation. For diverse aim of each stage they design three different algorithms respectively. The TVRSM can successfully diminish the energy consumption of the cloud data center when compared with various traditional methods. For the runtime consolidation of VMs in cloud data centers, Khoshkholghi et.al [18] has exhibited several novel algorithms. Their aim is to diminish energy consumption and improve the computing resources consumption under SLA constraints regarding bandwidth, RAM and CPU. By conducting extensive simulation the efficiency of their algorithm is validated. While providing a high level of commitment their algorithm significantly reduces energy consumption. When judge against to the benchmark algorithms, the energy consumption can reduce by up to 28% and SLA can improve up to 87% based on their algorithms.

III. CLOUD ARCHITECTURE AND PROPOSED METHOD

In this work, the goal is energy effectiveness to save cost in our private cloud. At IaaS level in cloud, the fundamental methodology utilized for enhancing resource effectiveness in data centers is virtualization; over the similar hardware infrastructure it permits the deployment of existing computing environments. Nonetheless, the co-existence of environments frequently constructs scenarios of high contest for resources between operational workloads, leading to

performance degradation. This phenomenon affects the data centers energy-efficiency. A new workflow scheduler service is introduced

in this paper and in all kinds of workload energy efficiency is enhanced in our private Cloud data centers by dynamic VM replacement.

A. Cloud Architecture

In SaaS level, for consumer the cloud provides eternal services, and the consumer can pay for as little or as much computing power as they require. The unified resources (virtual machines) are virtualized as underlying heterogeneous resources. According to the request of workflow the number of virtual machine can easily scale. Cloud can set up on the earlier Grid infrastructure. Meanwhile, IaaS cloud service suppliers like Google and Microsoft, Amazon some of the virtual machines can be created. At the unified resource layer the virtual machines in the private Cloud are supervised for optimizing the VM to physical host. In this paper, the main objective is minimizing energy consumption in data center.

So as to accomplish this aim and maintain the cloud elasticity, allocation strategies and dynamic provisioning are required to control the internal settings of the cloud to address overload and under workload in cloud. We propose MSSA to guarantee the response time and deadline of the resource. A procedure of mapping migrated virtual machines to physical machines is migrated virtual machine placement and improves power efficiency and resource utilization.

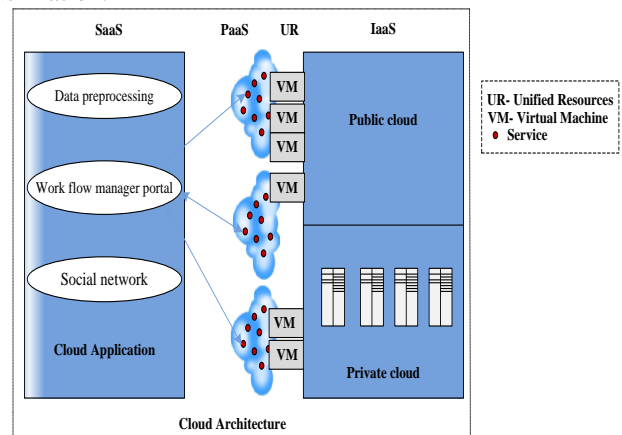


Fig.1. Architecture of cloud

B. Proposed Methodology

In IaaS level to accomplish these aims the resource allocation ought to be enhanced, that usually are sorted as, find overload host, for migration select VM in overload host, VM placement, find underload host, VM placement and power off underload host. The VM replacement is the focus of this study. The bin packing issue is a VM replacement with variable prices and sizes of bin, where bins represent the physical nodes; VMs are the allocated items; bin sizes are the nodes existing ability of CPU; and costs match up to the power utilization by the nodes. An ordinal technique isn't appropriate to solve it in an adequate complexity as the bin packing issue is NP-hard, hence that heuristic methods are suggested.

In this paper to discover new placement for all candidates of VM the evolutionary algorithm as a heuristic method is chosen. By implementing our heuristic technique dynamically, we need selecting a strategy with minor complexity and adjacent optimum solution; therefore, we utilize MSSA optimization which converges faster than many other global optimization algorithms [19].

In contrast to Genetic Algorithm (GA) [20], MSSA is usually quicker and it is more prominent in more complex cases. In further, MSSA is less complex to comprehend and execute because of its lower computational costs. At last, contrasted with different metaheuristics optimization strategies the MSSA has fewer parameters to change. Under all host of the data centre our algorithm has the entire perspective of placement of VM migrated and thinks the best likely permutation of them. In the final step, to lessen the number of hosts we utilize iterative technique and for the sake of cost saving the underload host is switched off.

In different stages author utilized the best algorithm based on the result of Wu et al. [21]. The technique of our optimization algorithm to enhance VM allotment to hosts is presented below.

Method: To Enhance VM allocation to hosts.

```

Migration Map optimize  $V_m$  Allocation( $V_m$  List,
H (host) list)
{
Exclude all VMs in  $V_m$  List that in relocation or just
assign
Discover all overload hosts (with local regression
algorithm)
Select VM in overload host to be migrated (with MMT)
Placement this VM (call MSSA for this VM)
Sort ascending host by use
For (all under load host)
Until possible (capacity not full)
Transfer all VMs from first host of queue
Exclude host from first of queue and power it off
}
    
```

The diminishing performance is avoided in the beginning of this strategy and from the list of VM choice under migrated VM was deleted. After that the host overload was noticed with Robust Local Regression method that forecast the prospect workload. In local regression method, localized plain models fitting to subsets of data to build up a curve that estimated the novel data. The version of Local Regression is susceptible to outliers that can be reasons by leptokurtic distributions. In Robust LRA algorithm, to build Loess robust, the addition of the robust estimation method bisquare to the least-squares method for fitting a parametric family Cleveland has proposed. This alteration change loess into an iterative method [22]. After the overload host is noticed, we choose particular VMs to transfer from this host, and in this step to choose the best VM in the active host we utilize MMT (minimum migration time) strategy, and we recur the technique until the host status is not overload. To decrease power in VMs selection location the new host is very essential, we do it by MSSA, and the fitness function is designed by total of power increase and cost of relocation to host. In the end, VMs merge to fewer hosts to diminish power consumption by lie dormant under load hosts.

IV. PROPOSED MODIFIED SALP SWARM ALGORITHM FOR VM MIGRATION

MSSA model is explained for the VM migration problem. The MSSA algorithm is planned in this paper for choosing the optimal VM. The SSA is a meta-heuristic algorithm, which relies on the conduct of swarming and salps population [23]. It could be implemented and useful to solve various function optimization problems. In MSSA, the fundamental structure of the SSA is customized by enhancing the updating phase of the population’s location. Hence, this alteration adds more flexibility in exploring the population and quickly reaches the optimal solution. The steps for MSSA technique is discussed beneath.

Step 1: Initialization

In the first step, the input parameters required in MSSA for VM placement are initialized as utilization of CPU, RAM and BW (bandwidth) of the PM.

Step 2: Population Initialization

Then, produce the population which signify a set of solution for VM placement and it is represented as,

$$P_M(I) = \{P_1, P_2, P_3, K, P_M\}; \text{ for } 1 \leq M \leq S \tag{1}$$

Where, the random number within the interval [0, 1] is represented as S . In the population, the solution resembles the VM. The meta-heuristic behavior of the salps the solution updates its position over the course of iteration.

Step 3: Fitness Function

In this section, the fitness evaluation function for proposed VM migration strategy is described. To surmount the PM overload problem with the decrease in relocation cost and energy utilization, and increase in the resource use, the fitness function is defined. The objective function of the proposed technique is,

$$\text{Fitness Function} = \text{Min}\{\text{resource utilization, energy consumption, migration cost}\} \tag{2}$$

Where,

$$\text{Resource utilization} = \frac{1}{M \times N} \left[\sum_{i=1}^N \sum_{j=1}^M \frac{BCPU_j}{ACPU_i} + \frac{BRAM_j}{ARAM_i} + \frac{BBW_j}{ABW_i} \right] \tag{3}$$

$$\text{Migration Cost} = \frac{1}{M} \left[\sum_{i=1}^N \frac{\text{Numbers of VMs movements}}{\text{Total no. of VMs}} \right] \tag{4}$$

Here, $A_i; 1 \leq i \leq N$ be the set of PM in cloud data centre, $B_j; 1 \leq j \leq M$ be the set of VM accessible in PM.

A_j^{CPU} represent CPU utilized by PM, B_j^{CPU} represent CPU use by VM, A_j^{RAM} entire memory of PM, B_j^{RAM} total memory of VM, A_j^{BW} total bandwidth of PM, B_j^{BW} total bandwidth of VM and N is number of VMs.

Next, depends on the quality of the fitness function (probability in eq. (11)) the proposed algorithm is to revise the present population by the SSA or modified algorithm.

Step 4: Updating SSA Position

A. Updating Leader Position

Based on the accompanying formula, the leader salps position is updated with respect to the source of food.

$$Pli = \begin{cases} \{Fpj + r1 [(uj - lj)r2 + lj] & r3 \geq 0 \\ \{Fpj - r1 [(uj - lj)r2 + lj] & r3 < 0 \end{cases} \quad (5)$$

Where, u_i and l_i represents the upper and lower bounds respectively, Pli represents the salp leader position, Fpj is the position of food source in the dimension of i and r_1 to r_2 are the random numbers at uniform. The initial random number which makes the exploration and exploitation in the balanced state and it is derived as,

$$R_i = 2 \cdot \exp \left[- \left(\frac{4m}{i_{\max}} \right)^2 \right] \quad (6)$$

Where, m indicates iteration at current and i_{\max} is the maximum number of iterations.

B. Updating Follower Position

After the evaluation process, the position of the followers is updated and the best and worst leader salps of the inputs are evaluated based on the best and worst leader salps. By relying on the stochastic attractor of leader salps, the searches for the best position of follower salp which are represented in the following equation:

$$P_{j,i} = (x_i + v_0) \cdot \frac{t}{2} \quad \forall i \geq 2 \quad (7)$$

A. Where, $P_{j,i}$ is the i^{th} follower salp position in the j^{th} dimension, t denotes time, v_0 is the original speed which is normally assumed to be 0.

$$P_{j,i} = \frac{P_{j,i} + P_{j,i-1}}{2} \quad \forall i \geq 2 \quad (8)$$

Equation (7) and (8) can be expressed to update salps follower.

Step 5: Modified Updating position

By producing a set of random particles with generating positions the swarm begins in the dimension. The SSA evaluates the food fitness for every solution as X_i ; $for i = 1, 2, 3, \dots, N$. The modified updating position is expressed as below.

$$X_{ij}^{(t+1)} = X_{ij}^{(t)} + V_{ij}^{(t+1)} \quad (9)$$

$$V_{ij}^{(t+1)} = WV_{ij}^{(t)} + r_1 C_1 (X_{ij}^{(t)P} - X_{ij}^{(t)}) + r_2 C_2 (X_j^{(t)g} - X_{ij}^{(t)}) \quad (10)$$

Where, the i^{th} position of particle and velocity at j^{th} dimension is expressed as X_{ij} and V_{ij} , an inertia weight is denoted as W , which is applied to enhance the population convergence speed. The acceleration coefficients are C_2 and C_1 constant.

Step 6: Probability of fitness function

The probability of each fitness function is computed as follows,

$$PB_i = \frac{(fitness\ function)_i}{\sum_{i=1}^N (fitness\ function)_i} \quad (11)$$

Where, the probability of fitness function is denoted as PB_i and N is dimension. In our research, according to PB_i value the MSSA algorithm is performed. If $PB_i > 0.5$ then the current solution will be updated using the SSA (in step 4). Otherwise ($PB_i < 0.5$), the current solution will be updated using the modified function (in step 5). Hence, for each solution the fitness function is calculated and determined the finest resolution after revise the population.

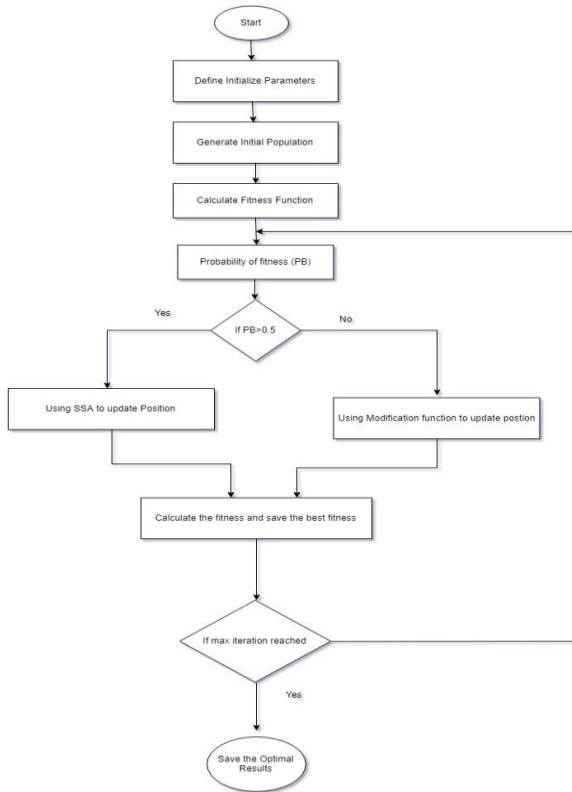


Fig.2. Flowchart of MSSA

Step 7: Termination

This section checks the termination condition. If the end situation is fulfilled then return by the best solution, or else, replicate the preceding steps from computing the probability to the end.

After the termination, the result with the finest fitness is selected as the best result. In viewpoint of VM, the optimal solution is the solution which is located optimally with minimal power and relocation rate as per the fitness function. The general technique of MSSA is appeared as the flowchart in figure 2.

5.1 Problem Formulation

The consolidation of the VMs assists to optimize the handling of resources and hence lessen the energy utilization in a cloud data centre. To apply the MSSA fruitfully for the located VM assignment in a suitable host problem, one of the main issue is how to plan the problem solution to the particle space, which straight affects its viability and performance. The search space is set up for m dimension in placement problem (host- m , VM- n). Each dimension is limited to $1n$. During the course of the MSSA each salps position dimension maps one VM, and the value of the location indicates the power of host plus cost of relocation which this VM is assigned. Our fitness function is illustrated as bellow:

Algorithm 1: Fitness function Evaluation

```

Fitness function( $V_m$  List,  $H$  (Host))
{
for (all  $V_m$  in  $V_m$  List)
{
if ( $H$  is appropriate for  $V_m$  in CPU (MIPS) and RAM and
BW)
{
maximum Utilization= get Maximum Utilization After
Allocation ( $H$ ,  $V_m$ );
if (maximum Utilization < Utilization Threshold())
{
compute host powerDiff with add  $V_m$  to  $V_m$ 
if(powerDiff < minimum power) { save  $V_m$ ;
minimum power= powerDiff; }
}
}
if (Discover appropriate  $V_m$ )
assign  $V_m$  to  $H$ ;
}}
if(old Host == new Host)
{Migration rate = 0}
else {compute Migration rate}
return(minimum power + Migration cost);
}
    
```

Where, MIPS represent million instructions per second of all processor in VM and host (PM). BW denotes bandwidth. In short, numerous steps perform in fitness function; first the host finds that by calculation the current VM utilization not departed of threshold and the power with adding the present VM minimize between VMs, final the cost of VM migration plus return the amount increased power.

V.RESULTS AND CONSIDERATION

In this part outcome of work and discussion of the proposed MSSA based VM migration technique are discussed.

A. Experimental Setup

It describes about the experimental outcomes of the proposed method implemented in a PC with 2 GB physical memory, 64-bit Windows 8 operating system and Intel(R) core(TM) i5 processor and for implementation the JAVA software tool is used. Here, over 3 cloud setups the experimentation of the proposed VM migration is determined. Utilizing CloudSim toolkit with JAVA the cloud setup with a various configuration is simulated.

- First Cloud setup: Simulation of cloud is executed with 20 VMs and 5 PMs.
- Second Cloud setup: Simulation of cloud is executed with 15 VMs and 4 PMs.
- Third Cloud setup: Simulation of cloud is executed with 10 VMs and 3 PMs.
- Fourth Cloud setup: Simulation of cloud is executed with 100 VMs and 30 PMs.

B. Performance Measures of Proposed Methodology

The performance of the planned MSSA method is evaluated with existing VM migration technique to confirm the efficiency of the proposed method. For the validation, the relative methodologies used are PSO (Particle Swarm Optimization), and SSA (Salp Swarm Algorithm). In the experimentation, for all the considered cloud setups the performance validation is conducted. The migration cost, resource use, and energy consumption are the performance measures utilized for the evaluation of the proposed method.

Migration cost

The cost of VM migration is called migration cost. The migration rate is defined in equation (3).

Resource use

It is the evaluation that identifies the utilized resource by the VM from the available PM resources. Resource usage is expressed in eq. (4).

6.3. Comparative Analysis of Various Methodologies

In this section the comparative analysis of various methodologies are deliberated. For different cloud setup by varying the number of rounds the analysis is performed distinctly and evaluates the corresponding performance measure value. Based on energy utilization, migration charge and resource exploit in this section the proposed analysis is presented.

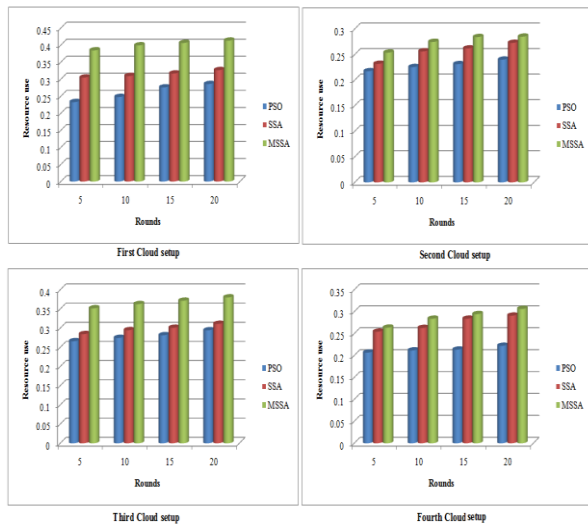


Fig.3. Performance analysis of resource use of various methodologies

The performance analysis of resource usage of various methodologies with four cloud setup is shown in figure 3. Between the resource use value and number of rounds the resource evaluation graph is plotted. In figure that the performance of the proposed method (MSSA) is higher of 39.4%, 37.9%, 32.1%, 30.7% and 20.7%, 22.5%, 22%, 20.9% with varying rounds for First Cloud setup when compared with PSO and SSA. Similarly, 14.3%, 17.9%, 18.6%, 15.7% and 8.6%, 6.7%, 7.8%, 4.2% for Second Cloud setup, 24.3%, 24.31%, 24.2%, 22.5% and 19.12%, 18.74%, 18.9%, 18.14% for Third Cloud setup and 21.6%, 25.4%, 27.49%, 27.46% and 3.4%, 7.4%, 3.5%, 4.9% for

Fourth Cloud setup when compared with PSO and SSA. The theoretical cause for better performance is the proposed method utilized the modified function for optimal VM selection.

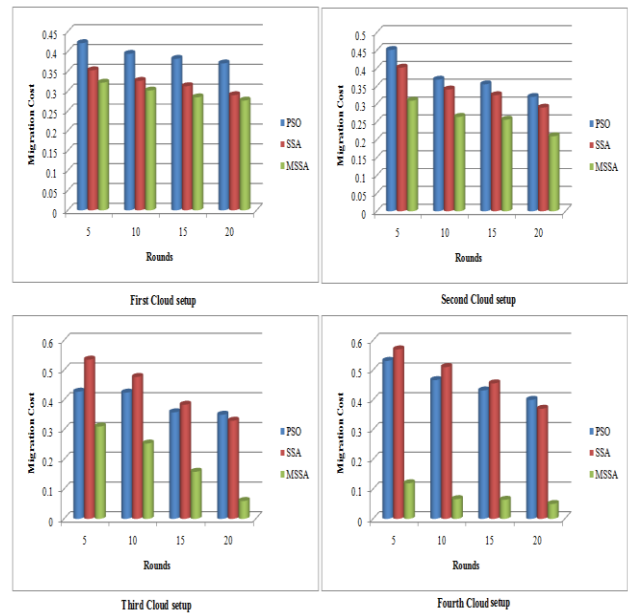


Fig.4. Performance analysis of migration cost of Various Methodologies

Figure 4 shows the performance analysis of migration cost of various methodologies with four cloud setup. Between the migration cost value and number of rounds the migration cost evaluation graph is plotted. In the figure the performance of the proposed method is better for First Cloud setup as 23.6%, 23.4%, 25.4%, 25.2% and 8.8%, 7.4%, 8.9%, 4.6% with varying number of rounds when compared with PSO and SSA. For Second Cloud setup 31.4%, 28.2%, 27.8%, 34.3% and 23%, 22.4%, 21.1%, 27.5% with varying number of rounds when compared with PSO and SSA. In Third Cloud setup the performance of the proposed method is better of 27.4%, 40.3%, 55.8%, 82.8% and 42%, 46.9%, 58.7%, 81.8%, similarly for Fourth Cloud setup it is better of 77.4%, 85.8%, 85.1%, 87.5% and 79%, 87%, 85.9%, 86.4%, with varying number of rounds when compared with existing techniques.

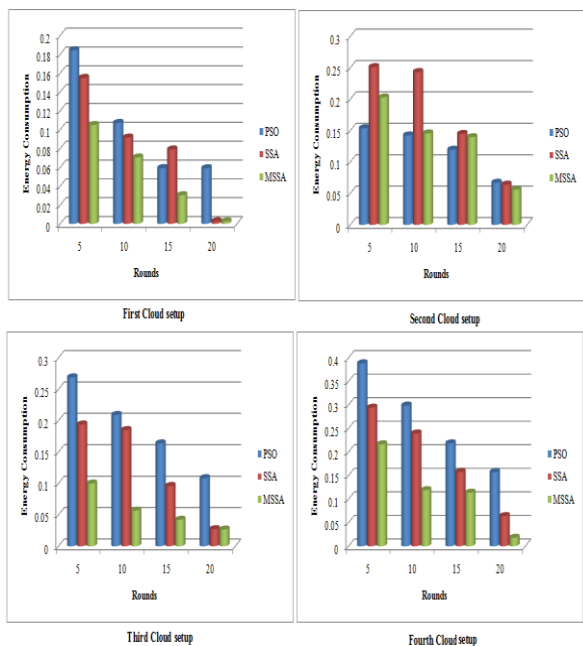


Fig.5. Performance analysis of energy utilization of various methodologies

The performance examination of energy utilization of various methodologies is shown in figure 5. With varying number of rounds the energy consumption of proposed method decreases. Between the energy consumption value and number of rounds the energy consumption evaluation graph is plotted. We observed that, the performance of the projected method is improved of 43%, 34.2%, 48.5%, 95.5% and 32.2%, 23%, 61.4%, 2.5% for First Cloud setup when compared with PSO and SSA, for Second Cloud setup the proposed method shows better when compared with PSO and SSA, 62.8%, 72.8%, 74.2%, 75.4% and 48.4%, 69.2%, 56.02%, 2.2% for Third Cloud setup when compared with PSO and SSA and 44.2%, 60%, 47.8%, 88.5% and 26.4%, 50.2%, 27.9%, 72% for Fourth Cloud setup when compared with PSO and SSA.

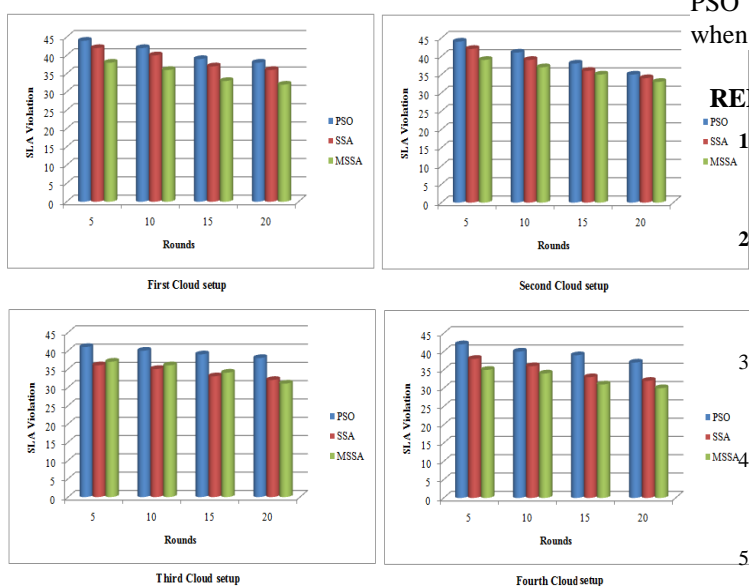


Fig.6. Performance analysis of SLA violation of various methodologies

Figure 6 shows the performance analysis of SLA violation of various methodologies with different cloud setup. Between the SLA violation value and number of rounds the SLA violation evaluation graph is plotted. It is noticed in figure that the performance of the proposed method (MSSA) is higher of 14.7%, 10.36% for First Cloud setup when compared with PSO and SSA, 8.6%, 4.4% for Second Cloud setup when compared with PSO and SSA, 12.7% for Third Cloud setup when compared with PSO and SSA and 17.7%, 6.4% for Fourth Cloud setup when compared with PSO. The theoretical reason for better performance of proposed method is utilization of modified updation function for optimal VM migration. It is clear that from the analysis for all considered cloud setups for evaluation, the proposed MSSA resulted in VM migration with minimum relocation cost, minimum energy utilization and maximum resource use.

VI. CONCLUSION

This paper presents, an MSSA based VM migration for the cloud data center. The issue of VM placement was presented by resource utilize and migration cost of the VM, also energy consumption of PM. Based on the formulated VM migration algorithm named MSSA, for changing the VM from the overloaded PM the optimal VM is selected. The VM migration is designed by modifying SSA. Moreover, to select the VM based on relocation cost, energy utilization and resource use a new fitness function was formulated. Using CloudSim the experimentation of the proposed VM migration strategy is verified over generated four cloud setups. We compare our work with existing methodologies and measures resource use, energy consumption and migration cost the performance of the proposed system. Additionally, the SLA Violation performance of the proposed method (MSSA) is higher of 14.7%, 10.36% for First Cloud setup when compared with PSO and SSA, 4.4% for Second Cloud setup when compared with PSO and SSA, 12.7% for Third Cloud setup when compared with PSO and SSA and 17.7%, 6.4% for Fourth Cloud setup when compared with PSO.

REFERENCES

1. K. Gai, M. Qiu, H. Zhao, L. Tao and Z. Zong, "Dynamic energy-aware cloudlet-based mobile cloud computing model for green computing", Journal of Network and Computer Applications, vol. 59, pp. 46-54, 2016.
2. R. Buyya, C. Yeo, S. Venugopal, J. Broberg and I. Brandic, "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility", Future Generation Computer Systems, vol. 25, no. 6, pp. 599-616, 2009.
3. M. Dikaiakos, D. Katsaros, P. Mehra, G. Pallis and A. Vakali, "Cloud Computing: Distributed Internet Computing for IT and Scientific Research", IEEE Internet Computing, vol. 13, no. 5, pp. 10-13, 2009.
4. C. Wu, R. Chang and H. Chan, "A green energy-efficient scheduling algorithm using the DVFS technique for cloud datacenters", Future Generation Computer Systems, vol. 37, pp. 141-147, 2014.
5. S. Garg, S. Versteeg and R. Buyya, "A framework for ranking of cloud computing services", Future Generation Computer Systems, vol. 29, no. 4, pp. 1012-1023, 2013.

6. H. Li, G. Zhu, C. Cui, H. Tang, Y. Dou and C. He, "Energy-efficient migration and consolidation algorithm of virtual machines in data centers for cloud computing", *Computing*, vol. 98, no. 3, pp. 303-317, 2015.
7. D. Boru, D. Kliazovich, F. Granelli, P. Bouvry and A. Zomaya, "Energy-efficient data replication in cloud computing datacenters", *Cluster Computing*, vol. 18, no. 1, pp. 385-402, 2015.
8. R. Calheiros, R. Ranjan, A. Beloglazov, C. De Rose and R. Buyya, "CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms", *Software: Practice and Experience*, vol. 41, no. 1, pp. 23-50, 2010.
9. S. Wang, A. Zhou, C. Hsu, X. Xiao and F. Yang, "Provision of Data-Intensive Services Through Energy- and QoS-Aware Virtual Machine Placement in National Cloud Data Centers", *IEEE Transactions on Emerging Topics in Computing*, vol. 4, no. 2, pp. 290-300, 2016.
10. M. Abdullahi, M. Ngadi and S. Abdulhamid, "Symbiotic Organism Search optimization based task scheduling in cloud computing environment", *Future Generation Computer Systems*, vol. 56, pp. 640-650, 2016.
11. C. Fiandrino, D. Kliazovich, P. Bouvry and A. Zomaya, "Performance and Energy Efficiency Metrics for Communication Systems of Cloud Computing Data Centers", *IEEE Transactions on Cloud Computing*, vol. 5, no. 4, pp. 738-750, 2017.
12. S. Hosseinimotlagh, F. Khunjush and R. Samadzadeh, "SEATS: smart energy-aware task scheduling in real-time cloud computing", *The Journal of Supercomputing*, vol. 71, no. 1, pp. 45-66, 2014.
13. Z. Zhou, Z. Hu, J. Yu, J. Abawajy and M. Chowdhury, "Energy-efficient virtual machine consolidation algorithm in cloud data centers", *Journal of Central South University*, vol. 24, no. 10, pp. 2331-2341, 2017.
14. X. Li, P. Garraghan, X. Jiang, Z. Wu and J. Xu, "Holistic Virtual Machine Scheduling in Cloud Datacenters towards Minimizing Total Energy", *IEEE Transactions on Parallel and Distributed Systems*, vol. 29, no. 6, pp. 1317-1331, 2018.
15. H. Duan, C. Chen, G. Min and Y. Wu, "Energy-aware scheduling of virtual machines in heterogeneous cloud computing systems", *Future Generation Computer Systems*, vol. 74, pp. 142-150, 2017.
16. F. Rossi, M. Xavier, C. De Rose, R. Calheiros and R. Buyya, "E-eco: Performance-aware energy-efficient cloud data center orchestration", *Journal of Network and Computer Applications*, vol. 78, pp. 83-96, 2017.
17. W. Zhu, Y. Zhuang and L. Zhang, "A three-dimensional virtual resource scheduling method for energy saving in cloud computing", *Future Generation Computer Systems*, vol. 69, pp. 66-74, 2017.
18. M. Khoshkholghi, M. Derahman, A. Abdullah, S. Subramaniam and M. Othman, "Energy-Efficient Algorithms for Dynamic Virtual Machine Consolidation in Cloud Data Centers", *IEEE Access*, vol. 5, pp. 10709-10722, 2017.
19. A. Abraham, H. Guo and H. Liu, "Swarm Intelligence: Foundations, Perspectives and Applications", *Swarm Intelligent Systems*, pp. 3-25.
20. M. Tang and S. Pan, "A Hybrid Genetic Algorithm for the Energy-Efficient Virtual Machine Placement Problem in Data Centers", *Neural Processing Letters*, vol. 41, no. 2, pp. 211-221, 2014.
21. Z. Wu, X. Liu, Z. Ni, D. Yuan and Y. Yang, "A market-oriented hierarchical scheduling strategy in cloud workflow systems", *The Journal of Supercomputing*, vol. 63, no. 1, pp. 256-293, 2011.
22. 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", *Concurrency and Computation: Practice and Experience*, vol. 24, no. 13, pp. 1397-1420, 2011.
23. S. Mirjalili, A. Gandomi, S. Mirjalili, S. Saremi, H. Faris and S. Mirjalili, "Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems", *Advances in Engineering Software*, vol. 114, pp. 163-191, 2017.
24. S. Dashti and A. Rahmani, "Dynamic VMs placement for energy efficiency by PSO in cloud computing", *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 28, no. 1-2, pp. 97-112, 2015.

AUTHOR PROFILE



Mr. Rajesh Patel pursued Bachelor of Computer Engineering from HNGU University of Gujarat, India in 2005 and Master of Computer Science and Engineering from Nirma University in year 2011. He is currently pursuing Ph.D. and working as an Assistant Professor in the Department of Computer Engineering, GPERI, Mehsana since 2012.

He is a member of CSI since 2013. He has published more than 10 research papers in reputed international journals and presented in conferences including IEEE which are available online. His main research work focuses on Computer Network, Sensor Network and Cloud Computing. He has 14 years of teaching experience and 5 years of Research Experience.