BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **21**, No 1
Sofia • 2021 Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.2478/cait-2021-0005

Virtual Machine Placement Using Energy Efficient Particle Swarm Optimization in Cloud Datacenter

Madhumala R. B.[1], *Harshvardhan Tiwari*[2], *Devaraj Verma C.*[1]

[1]*Department of Computer Science Engineering, JAIN (Deemed to be University), Bangalore, Karnataka, India*
[2]*CIIRC, Jyothy Institute of Technology, Bangalore, Karnataka, India*
E-mails: madhumala8887@gmail.com tiwari.harshvardhan@gmail.com c.devaraj@jainuniversity.ac.in

Abstract: *Efficient resource allocation through Virtual machine placement in a cloud datacenter is an ever-growing demand. Different Virtual Machine optimization techniques are constructed for different optimization problems. Particle Swam Optimization* (*PSO*) *Algorithm is one of the optimization techniques to solve the multidimensional virtual machine placement problem. In the algorithm being proposed we use the combination of Modified First Fit Decreasing Algorithm* (*MFFD*) *with Particle Swarm Optimization Algorithm, used to solve the best Virtual Machine packing in active Physical Machines to reduce energy consumption; we first screen all Physical Machines for possible accommodation in each Physical Machine and then the Modified Particle Swam Optimization* (*MPSO*) *Algorithm is used to get the best fit solution.. In our paper, we discuss how to improve the efficiency of Particle Swarm Intelligence by adapting the efficient mechanism being proposed. The obtained result shows that the proposed algorithm provides an optimized solution compared to the existing algorithms.*

Keywords**:** *Cloud computing, Virtual Machine optimization, Particle Swarm Optimization* (*PSO*)*, Energy Efficiency, Resource Allocation, Fitness Function.*

1. Introduction

With the information technology covering the everyday tasks of any individual, the IT service provider companies are increasing by leaps and bounds. Any business like educational service provider, transportation management, retail business, finance, etc. has a requirement of IT infrastructure. Many service providers are willing to transfer the IT infrastructure overhead requirement to the third-party service providers and these service providers are termed as cloud service providers. Cloud service providers provide IT infrastructure such as CPU cycles, RAM and storage, software, services on a rental basis. With the increase in cloud service consumers, the cloud service providers started to experience a new problem such as optimized allocation of the resources, proper positioning of the data center, optimized scheduling of the jobs, etc. Out of all the above, said problems resource allocation is one of the confining problems in the data center. In the earlier days the service providers came up with a bundled solution, each with a predefined amount of processing capacity, RAM, and Storage as in the case of Amazon EC2 Instances. These individual instances are termed as Virtual Machines and since these came with preconfigured resource allocations, it was easy for the cloud service providers to allocate each such instance or Virtual Machine (VM) to a real Physical Machine (PM).

As a physical machine may house thousands of such VMs (Virtual Machines), now again the scenario changed: apart from the fixed instances there was a demand from the users to increase just a single resource such as more computing power, more RAM, or more Storage, the earlier algorithms were intended to place these fixed resource virtual machines in the cloud, and they neglected to provide for dynamically requested resources over and above, the fixed VM resources. The objective of the paper is to accommodate dynamically requested resources along with the fixed-sized VMs while reducing the power consumption of the data center and increasing the resource utilization in each PM.

Out of all these algorithms, on the base of careful comparison, we decided to use Particle Swarm Optimization (PSO) with custom initial particles, number of iterations, and the fitness function. The results seem to be promising and they are provided in the result section using nice visual charts.

PSO is one of the stochastic swarm intelligence algorithms to find the optimal solution for the given problem over the problem search space. PSO has been proposed in 1995 by Kennedy and Eberhart [1]. The PSO Algorithm has four main components that will decide the efficiency of the given algorithm, namely – initial position, velocity, and weight parameters, and the fitness function. Here in this paper we will discuss how to set the initial position and initial velocity so that the candidate solution obtained is the best one. To verify the authenticity of the obtained solution we use the fitness function, PSO is optimized for the parameters we have intended. To solve the said problem we use PSO to where each particle maintains a local best and the global best solutions and after n number of iterations, the global best solution will be selected. Being an approximation algorithm PSO performs better when there are a lot of Virtual Machines (VMs) instances to be allocated on an active PM while satisfying the given objective by considering the energy-aware techniques used.

In the paper current Section 1 emphasizes the related works already done, their models and methodologies and discusses the optimization results. In Section 2 a survey of the related literature sources is done. In Section 3: we explain the big picture of our problem statement using a block diagram that clearly explains the inputs to the system process that follows and the expected output (Subsectin 3.1); then in Subsection 3.2 we elaborate on the block diagram and define the model and the process that follows. We have explained the lower-level implementation details of our algorithm. In Section 4 we have a result discussion, comparing the proposed method result with the existing methodologies. In the last section, we conclude our paper by mentioning the learning outcome of the selected problem and how it is better than earlier algorithms, and the scope of the future works.

2. Related work

Optimizing resource sharing and allocation is an NP-hard problem as it has multidimensional properties; if it were single dimensional problem Bin Packing algorithm would have solved the problem efficiently. In our problem statement, we consider computational elements, RAM, and Bandwidth that are the three major dimensions, the physical machine to be optimized for. Many authors have proposed different algorithms to solve the said problem and the notable ones are MBFD [2, 3], and MBFH [4], FFD, and so on. Beloglazov and Buyya [2] and Beloglazov, Abawajy and Buyya [3], have done considerable work on the VM-PM problem, allocation of VMs done in two ways, VM Provisioning and placement and other one is VM resource allocation optimization. In this paper, the author provides a complete overview of the VM placement problem by modifying the Bin packing problem. The MBFD [2], name is because all Virtual Machines are in decreasing order based on the dynamic utilization and by allocating each VM to a PM with minimum power consumption. In VM Optimization First part is to select the VMs that needs to be migrated second part is to place the VMs by using the MBFD algorithm, that is arranging all the VMs in decreasing order and finding the least power consumed node selecting that node as an optimal node for placement of VM. They have proposed heuristics to choose VMs such as Single Threshold (ST) [3], lower and upper thresholds settings based on this CPU utilization is measured. If the threshold is high only a few VMs will be migrated to manage the SLA violation.

In this paper the algorithm fails to check the overloaded probabilities of VMs before mapping with the active nodes, in turn, there is an increase in VM migrations, and also it increases the SLA violations [2]. One robust algorithm is needed to improve resource utilization by reducing energy consumption and also needs improvement in SLA violations. Srikantaiah, Kansal and Zhao [4] have introduced the Modified Best Fit Heuristic Algorithm (MBFH Algorithm) to minimize energy consumption in datacenter by optimizing the cloud resources. Energy efficiency is calculated using the current selection and optimal selection within the datacenter [4]. The energy consumed and resources utilized mainly focus on the CPU cycles and storage. In the paper of Blondin [5], true velocity is the difference between two successive particle positions is found. In VM selection one of the critical tasks that surfaces is migrating the (selected) VM to the other host without imposing overload [6].

Li and Pan [7] explain the concept of the PSO Algorithm, mainly concentrating on the global optimal solution by introducing extreme perturbations to find the optimal solution for the given problem. The results show that the virtual machines provide a greater response concerning the cloud resource utilization in the cloud datacenter when compared to the other swarm intelligence algorithms. Server consolidation is widely used to cut down the energy consumption [8].

Braiki and Youssef [9] have developed a Multi-objective Virtual Machine placement algorithm by using PSO where authors address the problem of maximizing the VM-PM mapping ratio by minimizing the energy utilization, explain how to pack the computing resources efficiently in the cloud datacenter by using a lower number of PMs thereby reducing energy consumption. Luo, Song and Yin [10] have investigated a reliable virtual machine placement for a single dimension of resource request failure over the cloud datacenter by considering the physical resource utilization and the loss rate of the optimization. Wang et al. [11], have investigated the applications and use of heterogeneous virtual machines by redefining the PSO parameters and operators to maximize resource utilization over the cloud datacenter. Kumar and Raza [12] have proposed two-dimensional encoding scheme for VM placement consisting of a number of the physical machine and the subset of compromising VMs in the VM placement problem. Tripathi, Pathak and Vidyarthi [13] use in this paper, BPSO to optimize the VM allocation by modifying the particle position and updating the particle velocity. CPU and Memory utilization dimensions are considered for the VM-PM placement problem.

Pandey et al. [14] present in this paper the PSO algorithm for VM placement by considering both communication costs and data transmission cost to eliminate the overall cost estimation of the scheduling process thereby improving the resource utilization over the cloud datacenter.

Gupta and Amgoth [15] have introduced Scheduled Virtual Machine Placement by using MPSO with semi scheduled mechanism where initial selection is based on the demands of VMs. Xiaoqing [16] have proposed Energy-efficient algorithms based on static algorithms such as DVPS, MPA to resolve the SLA violations in user-customized applications. Server consolidation aims to minimize the number of servers required for placing Virtual Machines [17]. The changing shape strengthens the particles that move toward the clustering vector of local and global optimal solutions [18].

3. Proposed work

3.1. Particle swarm optimization

PSO is a population-based optimization method based on Swarm Intelligence (SI). The basic idea is to find the particle's potential position according to its own experience and that of its neighbours. The PSO algorithm searches in parallel as well as in the group of individuals. Individuals or particles over a problem space, approach the optimal value through their current velocity, with their previous experience, and with the experience of their neighbours. The global method of PSO is to update the particle's position and velocity at each iteration. *pbest* and *gbest* are the particle's personal best and global best positions in the given problem space [19]. The Virtual Machine problem is addressed by our proposed model as Energy Efficient Particle Swarm Optimization (EEPSO) where the main aim is to minimize energy consumption and maximize the utilization of resources. The following section updates the process of how the particle's movement is adapted by the modified PSO Algorithm.

Updating Velocity in problem space:

(1) $$V_{id} = V_{id}+C_1*Rnd_1(0, 1)*(VMpb_{id}–PM_{id})+C_2*Rnd_2(0, 1)*(VMgb – PM_{id}).$$

Updating Position in problem space:

(2) $$PM_{id} = PM_{id} + V_{id},$$

where

id is the *i*-th number of particles in *d* dimension, V_{id} – Particle's velocity at iteration *d*,

PM_{id} – Particle's position at iteration d, C_1C_2 – Social components,

Rnd_1, Rnd_2 – uniformly distributed random numbers,

$VMpb_{id}$ –Particle's personal best position, VMgb – global best position of the particle.

This section presents the data flow of the Particle Swarm Optimization Diagram. In the cloud data center, the main aim of resource allocation is to minimize energy consumption and to maximize resource utilization. The selected Physical Machines (PMs) are called PM candidate list. This PM candidate list will be fed as input to the PSO Algorithm. In each iteration, PSO checks for the availability of resources, and based on the demand VM-PM mapping will be done. At each time the number of active PMs will be reduced which indicates decreasing of power consumption of VM-PM mapping and also increases the maximum utilization of resources [15]. The main intention of the selection and allocation of VM is to optimize the energy consumption by switching off the underutilized servers. In VM-PM Mapping, instead of selecting the available PMs, here we select only a set of PMs which match the required demands of VM. Based on this strategy primary PM candidate list will be prepared with the heterogeneity of resources.

3.2. Proposed solution

In our algorithm unlike all other previous authors we use a dynamic fitness function; our fitness function dynamically changes to filter the noise as well as to accommodate requested resources. In VM placement problem *N* represents number of VMs and *M* represents the number of PM in the cloud data center. Based on this the problem space can be denoted as M^N, and

$$\sum_m S_m^n = 1, \tag{3}$$

S_m^n =1 indicates availability of VM in the cloud datacentre otherwise S_m^n =0. VM placement is an optimization problem in which total power consumption must be reduced and the resource must be maximized in the cloud datacentre.

Energy Efficient PSO Algorithm

Step 1. //initialize all the particles
Step 2. **for each** particle *i* in *S* **do**
Step 3. **for each** dimension *d* in *D* **do**
Step 4. //initialize all particles position and velocity
Step 5. PM_{id} = Rnd(PM_{min}, PM_{max})
Step 6. **end for**
Step 7. //initialize particles best position
Step 8. $VMpb_{id}$= **MFFD** (PM_{id}, [VM list])
Step 9. //update the global best position
Step 10. **if** $f(VMpb_{id}) < f(VMgb)$ **then**
Step 11. VMgb=$VMpb_{id}$
Step 12. **end if**
Step 13. **end for**
Step 14. **repeat**

Step 15. **for each** particle i in S **do**
Step 16. //update particles best position
Step 17. **if** $f(\mathrm{PM}_i) < f(\mathrm{VMpb}_{id})$ **then**
Step 18. $\mathrm{VMpb}_{id}=\mathrm{PM}_{id}$
Step 19. **end if**
Step 20. //update global best position
Step 21. **if** $f(\mathrm{VMpb}_{id}) < f(\mathrm{VMgb})$ **then**
Step 22. $\mathrm{VMgb}=\mathrm{VMpb}_{id}$
Step 23. **endif**
Step 24. **end for**
Step 25. //update particles position and velocity
Step 26. **for each** particle i in S **do**
Step 27. **for each** dimension d in D **do**
Step 28. $V_{id} = V_{id} + C_1 * \mathrm{Rnd}_1(0, 1) * [\mathrm{VMpb}_{id} - \mathrm{PM}_{id}] + {} + C_2 * \mathrm{Rnd}_2(0, 1) * [\mathrm{VMgb} - \mathrm{PM}_{id}]$
Step 29. $\mathrm{PM}_{id} = \mathrm{PM}_{id} + V_{id}$
Step 30. **end for**
Step 31. **end for**
Step 32. //advance iteration
Step 33. it = it+1
Step 34. **until** it > MAX_ITERATIONS

Equation (3) denotes the VM n is allocated to PM m. and note that each VM can be allocated to only one PM. We propose our new Fitness Function for Particle Swarm Optimization (PSO FF) as given below.

Resource optimization in cloud datacenter is represented by using the equation

$$\text{PSO FF} = \begin{cases} \sum_n \mathrm{VM}_n^{\mathrm{CPU}} * S_m^n \le \mathrm{PM}_m^{\mathrm{CPU}}, \\ \text{and} \\ \sum_n \mathrm{VM}_n^{\mathrm{RAM}} * S_m^n \le \mathrm{PM}_m^{\mathrm{RAM}}, \end{cases} \quad (4)$$

where: $\mathrm{VM}_n^{\mathrm{CPU}}$, $\mathrm{VM}_n^{\mathrm{RAM}}$ denote demand of CPU and RAM; $\mathrm{PM}_m^{\mathrm{CPU}}$, $\mathrm{PM}_m^{\mathrm{RAM}}$ represent the available resources for PM. From the above equation we can see that the total resources demand of VMs must be less than the available PMs. To measure the quality of the result, swarm needs a fitness function. Based on the fitness function each particle can update its velocity and position in the given problem space to obtain the optimal solution. The whole process of allocation can be represented with the flowchart below.

Energy Efficient PSO flowchart is shown in Fig. 1, mainly the four steps in the method being proposed:

i) Initialize Swarm particles: set of VMs are generated randomly. Each VM is allocated to the PM by using Modified First Fit Algorithm.

ii) Evaluate fitness function for each particle using PSO FF. If all the particles meet the criterion given in Equation (3) then update the position and velocity of the particles.

iii) Obtain the personal best and global best values. Otherwise particles cannot be updated.

iv) If maximum number of iterations is reached, then stop the process; otherwise go to step two.

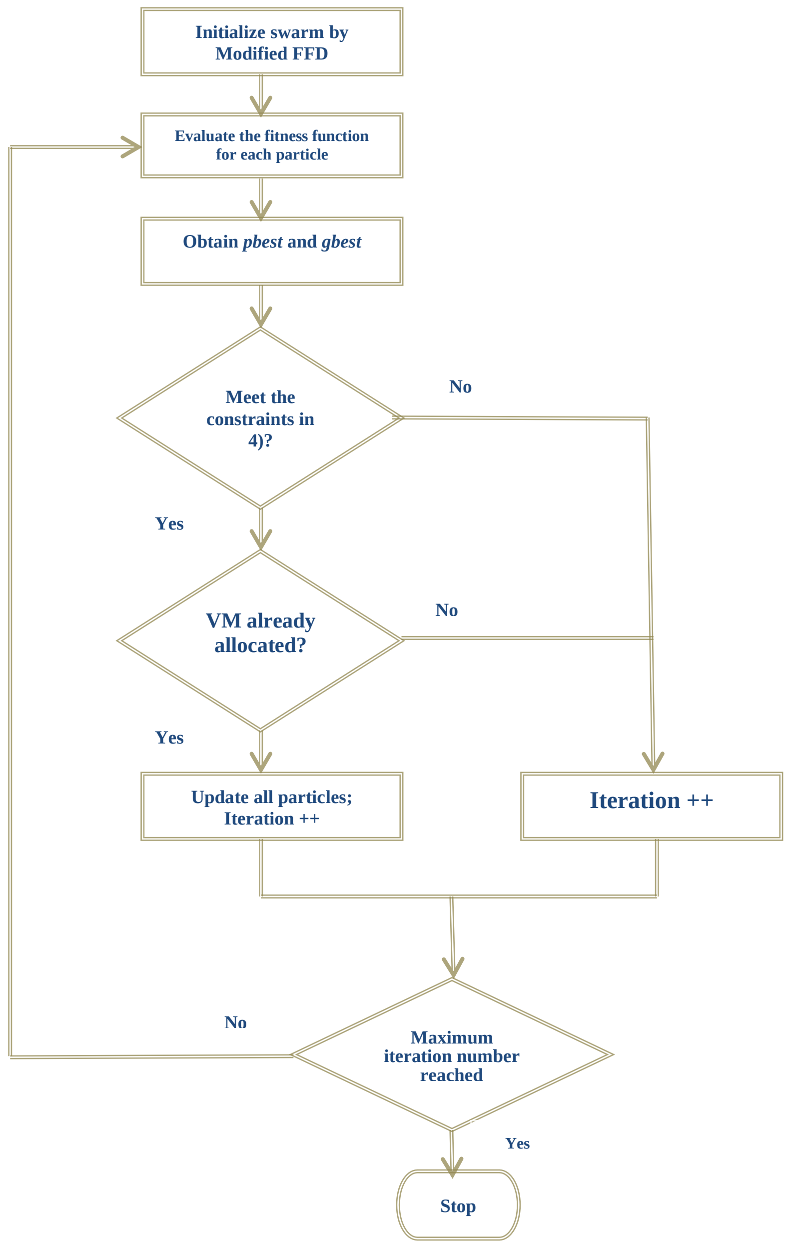

Fig. 1. Flowchart of Energy Efficient PSO

Cloud computing services always demand for the resources very concurrently. The combination of heuristic and meta-heuristic algorithms gives very good results when comparing the set of solutions. Heuristic methods try to fix the VM placement problem and a meta-heuristic algorithm allows mapping the demanded resources over the cloud datacentre. We have Modified FFD (MFFD) with predefined threshold values for individual resources in every host. Modified FFD accepts VM list and PM list as arguments and returns a list of mapped VM-PM pairs. We present the Custom Modified First Fit Decreasing algorithm based on user/cloud provider requirement:

Here our algorithm chooses either to sort VMs based on their CPU requirement or RAM requirement. If (CPUreqd > RAMreqd) we sort all the VMs in descending order of CPU requirement or else we sort all the VMs in descending order of RAM requirement.

4. Results and discussions

The algorithm proposed is evaluated using the Cloud simulation toolkit [21], which allows making use of customized policies for resource allocation over the cloud datacenter. The performance ratio of the proposed algorithm has been evaluated with MBFD [2, 3], MBFH [4], and MREE-PSO [20] heuristic algorithms. For the performance evaluation, we have considered the same set of VMs and PMs for all algorithms. We simulate the heterogeneous servers over the cloud resources such as CPU, BW, and Memory. In the simulation environment we have considered 100 VMs and 60 PMs, horizontal axis represents number of VMs and Vertical column represents parameter specification respectively. We have used Power VC Standard Version 1.4.1 values for the energy utilization of the hosts and experiments have been simulated on Intel i5, Seventh generation processor with 1.7 GHz quad-core CPU with 8GB of memory. The platform used for the experiment setup is Windows 10 Pro and Eclipse IDE is used for JAVA code editing.

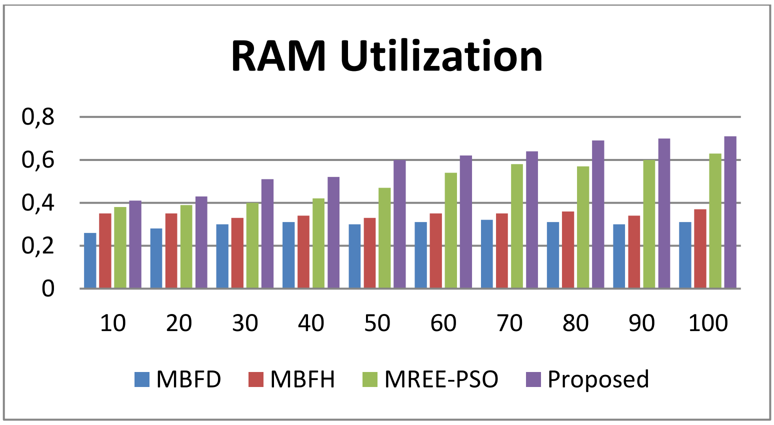

Fig. 2. RAM Utilization

The comparison of RAM utilization of MBFD [2, 3], MBFH [4], MREE-PSO [20], and our proposed algorithm is based on PSO evaluated, and results are shown in Fig. 2. As we can observe from the chart our proposed algorithm performs better when compared to other two algorithms from the beginning. This is mainly due to the piping of output of MFFD to PSO and the efficient fitness function proposed in our PSO Algorithm. As we increase the number of VMs the performance increases drastically compared to other two algorithms and our proposed algorithm is plotted based on the usage.

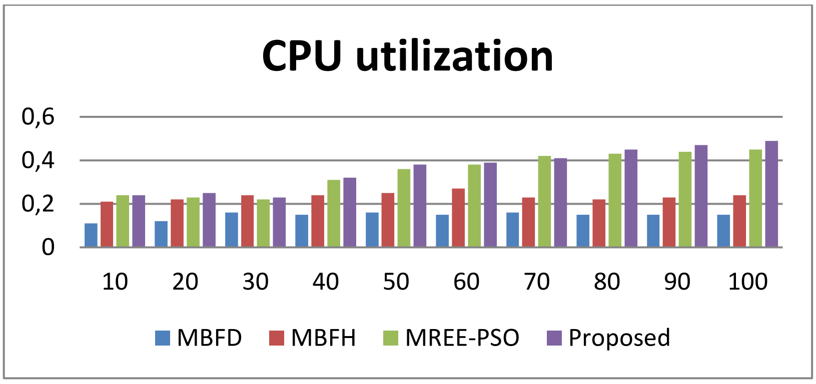

Fig. 3. CPU Utilization

Fig. 3 shows the comparison of CPU utilization among two well-known algorithms namely, MBFD [2, 3], and MBFH [4], MREE-PSO [20] along with our proposed new algorithm. As we can observe from the chart MBFD's performance is lagging behind MBFH and our proposed algorithm from VM values starting from 10 whereas MBFH and our proposed algorithm continue to be nearly the same till the VM count has reached 30. Above 30 VMs our proposed algorithm performs much better and has a linear increase in performance till the VMs count has reached 100. The given graph shows CPU usage rate, which gives the better idea and the results look promising as we increase the number of VMs.

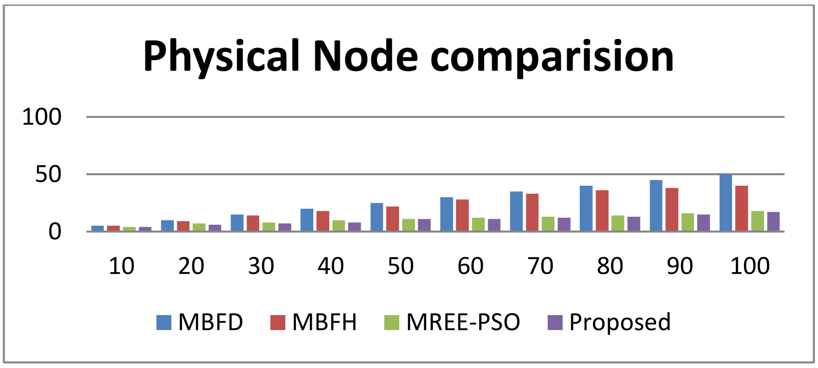

Fig. 4. Physical nodes comparison

Utilization of Physical nodes is plotted in Fig. 4. Our intended requirement is to reduce the number of active physical nodes so that the energy consumption to be reduced. From the above chart, we can conclude that for VM numbers above 50 our proposed algorithm drastically have reduced the number of active PMs. From 10 VMs to 50 VMs though our algorithm is not outperforming, the results it is in line with the results of the other two algorithms, this is because the swarm algorithms tend to perform better as we increase the swarm size. The physical nodes usage in our proposed algorithm is less, even if we increase the number of VMs when compared to MBFD [2, 3], MBFH [4], and MREE-PSO [20] heuristics. Thus, our proposed algorithm provides optimal allocation of resources which is very much required in a distributed cloud datacenter.

5. Conclusion

Optimization of Virtual Machines leads to better reduction in energy consumption over cloud datacenters. In our proposed algorithm we have introduced an optimal solution for the Virtual Machine placement problem by minimizing the energy component and by maximizing the resource utilization over the cloud user-customized services. The comparison is drawn upon the proposed and other popular algorithms such as MBFD, MBFH, and from the results, we can see that the proposed algorithm results are in accordance with other algorithms for a lesser number of Virtual Machines. However, as we increase the number of Virtual Machines, the proposed algorithm performs much better. The future scope of the work is to combine other heuristic methods to improve performance.

References

1. Kennedy, J., R. Eberhart. Particle Swarm Optimization. – In: Proc. of IEEE International Conference on Neural Networks, Vol. **4**, December 1995.
2. Beloglazov, A., R. Buyya. Energy Efficient Allocation of Virtual Machines in Cloud Data Centers. – In: Proc. of 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing, Melbourne, VIC, 2010, pp. 577-578. DOI: 10.1109/CCGRID.2010.45.
3. Beloglazov, A., J. Abawajy, R. Buyya. Energy-Aware Resource Allocation Heuristics for Efficient Management of Data Centers for Cloud Computing. – Future Generation Computer, Systems, Vol. **28**, 2012, No 5, pp. 755-768.
4. Srikantaiah, S., A. Kansal, F. Zhao. Energy-Aware Consolidation for Cloud Computing. – In: Proc. of IEEE Conference on Power-Aware Computing and Systems, IEEE Computer Society Press, San Diego, CA, USA, 2010, pp. 577-578.
5. Blondin, J. Particle Swarm Optimization: A Tutorial. 2009. **http://cs.armstrong. edu/saad/csci8100/pso tutorial. pdf.**
6. Govardhan, P., P. Srinivasan. Enhanced Evolutionary Computing Assisted Robust SLA-Centric Load Balancing System for Mega Cloud Data Centers. – Cybernetics and Information Technologies, Vol. **19**, 2019, No 3, pp. 74-93. DOI: 10.2478/cait-2019-0027.
7. Li, S., X. Pan. Adaptive Management and Multi-Objective Optimization of Virtual Machine in Cloud Computing Based on Particle Swarm Optimization. – J. Wireless Com Network, Vol. **102**, 2020. **https://doi.org/10.1186/s13638-020-01722-4**
8. Vijaya, C., P. Srinivasan. A Hybrid Technique for Server Consolidation in Cloud Computing Environment. – Cybernetics and Information Technologies, Vol. **20**, 2020, No 1, pp. 36-52.
9. Braiki, K., H. Youssef. Multi-Objective Virtual Machine Placement Algorithm Based on Particle Swarm Optimization. – In: Proc. of 14th International Wireless Communications & Mobile Computing Conference (IWCMC'18), Limassol, 2018, pp. 279-284. DOI: 10.1109/IWCMC.2018.8450527.
10. Luo, J., W. Song, L. Yin. Reliable Virtual Machine Placement Based on Multi-Objective Optimization with Traffic-Aware Algorithm in Industrial Cloud. – IEEE Access, Vol. **6**, 2018, pp. 23043-23052. DOI: 10.1109/ACCESS.2018.2816983.

11. W a n g, S., Z. L i u, Z. Z h e n g, Q. S u n, F. Y a n g. Particle Swarm Optimization for Energy-Aware Virtual Machine Optimization in Virtualized Data Centers. – In: Proc. of International Conference on Parallel and Distributed Systems, Seoul, 2013, pp. 102-109. DOI: 10.1109/ICPADS.2013.26.
12. K u m a r, D., Z. R a z a. A PSO Based VM Resource Scheduling Model for Cloud Computing. – In: Proc. of IEEE International Conference on Computational Intelligence & Communication Technology, Ghaziabad, 2015, pp. 213-219. DOI: 10.1109/CICT.2015.35.
13. T r i p a t h i, A,. I. P a t h a k, D. P. V i d y a r t h i. Energy Efficient VM Placement for Effective Resource Utilization using Modified Binary PSO. – The Computer Journal, Vol. **61**, June 2017, No 6, pp. 832-846. DOI: 10.1093/comjnl/bxx096.
14. P a n d e y, S., L. W u, S. M. G u r u, R. B u y y a. A Particle Swarm Optimization-Based Heuristic for Scheduling Workflow Applications in Cloud Computing Environments. – In: Proc. of 24th IEEE International Conference on Advanced Information Networking and Applications, Perth, WA, 2010, pp. 400-407. DOI: 10.1109/AINA.2010.31.
15. G u p t a, M. K., T. A m g o t h. Scheduled Virtual Machine Placement in IaaS Cloud: A MPSO Approach. – In: Proc. of 5th International Conference on Image Information Processing, Shimla, India, 2019, pp. 448-453
16. X i a o q i n g, Z. Energy-Aware Virtual Machine Management Optimization in Clouds. – In: Proc. of 3rd IEEE International Conference on Computer and Communications (ICCC), Chengdu, 2017, pp. 2434-2438. DOI: 10.1109/CompComm.2017.8322972.
17. B o o m i n a t h a n, P., M. A r a m u d h a n, R. K. S a r a v a n a g u r u. Fuzzy Bio-Inspired Hybrid Techniques for Server Consolidation and Virtual Machine Placement in Cloud Environment. – Cybernetics and Information Technologies, Vol. **17**, 2017, No 4, pp. 52-68.
18. T o s h e v, A. Particle Swarm Optimization and Tabu Search Hybrid Algorithm for Flexible Job Shop Scheduling Problem – Analysis of Test Results. – Cybernetics and Information Technologies, Vol. **19**, 2019, No 4, pp. 26-44.
19. M a d h u m a l a, R. B., H. T i w a r i, C. D e v a r a j v e r m a. A Reliable Frame Work for Virtual Machine Selection in Cloud Datacenter Using Particle Swarm Optimization. – International Journal of Mathematics and Computer Science, Vol. **16**, 2021, No 2.
20. X i o n g, An-ping, C.-X. X u. Energy Efficient Multi Resource Allocation of Virtual Machine Based on PSO in Cloud Data Center. – Mathematical Problems in Engineering, Vol. **2014**, 2014, Article ID 816518. 8 p.
https://doi.org/10.1155/2014/816518, 2014
21. C a l h e i r o s, R. N., R. R a n j a n, A. B e l o g l a z o v, C. A. F. D e R o s e, R. B u y y a. CloudSim: A Toolkit for Modeling and Simulation of Cloud Computing Environments and Evaluation of Resource Provisioning Algorithms. – Software: Practice and Experience, Vol. **41**, 2011, No 1, pp. 23-50.

Received: 29.09.2020; Second Version: 30.12.2020; Accepted: 19.02.2021