International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 10 Issue: 1s
DOI: https://doi.org/10.17762/ijritcc.v10i1s.5846
***Article Received**: 01 September 2022 **Revised**: 05 November 2022 **Accepted**: 30 November 2022*

Cloud Host Selection using Iterative Particle-Swarm Optimization for Dynamic Container Consolidation

[a] G Ravikumar, [b] Zunaira Begum, [c] A. Shiva Kumar, [d]V.Kiranmai, [e]M.Bhavsingh, [f]O.Kiran Kumar
[a] Assistant Professor, CSE Dept, CMR College of Engineering & Technology, Medchal, Hyderabad. Telangana. India.
gravikumar@cmrcet.ac.in
[b]Former Assistant Professor, Department of Computer Science and Engineering , Lords Institute of Engineering and Technology , Hyderabad. Telangana. India.
anandan.se@velsuniv.ac.in
[c] Assistant Professor, IT Dept, CMR College of Engineering & Technology, Medchal, Hyderabad. Telangana. India.
drshivakumar1978@gmail.com
[d]Assistant Professor, CSE Dept, Sri Indu College of Engineering & Technology, Hyderabad. Telangana. India
kiranmaibolishetti18@gmail.com
[e] Assistant Professor, Department of Computer Science and Engineering, Ashoka Women's Engineering College, Kurnool, Andhra Pradesh, India
bhavsinghit@gmail.com
[f]Lecturer , St. Joseph's Degree College, Sunkesula Road, Kurnool, Andhra Pradesh.India
kiran.k.o@gmail.com

Abstract— A significant portion of the energy consumption in cloud data centres can be attributed to the inefficient utilization of available resources due to the lack of dynamic resource allocation techniques such as virtual machine migration and workload consolidation strategies to better optimize the utilization of resources. We present a new method for optimizing cloud data centre management by combining virtual machine migration with workload consolidation. Our proposed Energy Efficient Particle Swarm Optimization (EE-PSO) algorithm to improve resource utilization and reduce energy consumption. We carried out experimental evaluations with the Container CloudSim toolkit to demonstrate the effectiveness of the proposed EE-PSO algorithm in terms of energy consumption, quality of service guarantees, the number of newly created VMs, and container migrations.

Keywords- Container as a Service (CaaS), Cloud Computing, Energy Efficiency, Dynamic Container Consolidation, and Particle Swarm Optimization.

1. INTRODUCTION

A cloud service that provides users with the capacity to manage and deploy containerized applications and clusters is known as containers-as-a-service (CaaS). There are some who believe that CaaS is a subset of the Infrastructure as a Service paradigm, but where the key commodity is containers rather than physical hardware and virtual machines. Containers are essentially an alternative to the classic virtualization strategy, in which containers virtualize the operating system rather than the hardware stack. Virtual machines run far more slowly than containers. When compared to virtual machines, which have to start up a whole OS every time they start, they use only a small amount of resources and memory.

Virtualization is the most significant paradigm shift in software development over the last decade. It allows for increased utilization of resources, shortened development times, and minimizes the amount of repeated work required to supply services[1]. Thanks to virtualization technology, development teams can now more easily duplicate production environment circumstances and perform focused applications at a cheaper cost. Due to the fact that each virtual environment requires its own operating system in order to function, running six instances of an operating system on the same hardware can be extremely resource intensive. Virtualization allows users to distribute their processing power among multiple virtual environments running simultaneously on the same machine. Virtualization may now be more precisely controlled thanks to containers [2]. A programme and all of its important dependencies, such as binaries and configuration files, are packaged together into a single package, rather than a whole machine, including the operating system and hardware, is virtualized using containers.

It's possible to run containers on physical or virtual machines. However, there are certain disadvantages to employing this architecture, such as the dependency between containers and operating system type, and security concerns because containers do not provide the same level of isolation as VMs [3]. Fig. 1 depicts the two-level

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 10 Issue: 1s
DOI: https://doi.org/10.17762/ijritcc.v10i1s.5846
***Article Received**: 01 September 2022 **Revised**: 05 November 2022 **Accepted**: 30 November 2022*

virtualization architecture used by several cloud providers.

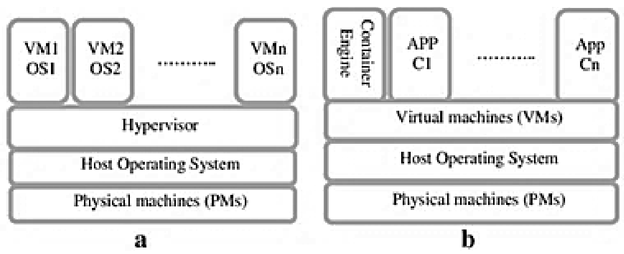

Figure 1. Two-level virtualization architecture

The construction of data centres throughout the world has increased significantly to accommodate the rising demand for cloud services. Power consumption in a cloud data centre is mostly due to the servers[4]. As a result, dynamic consolidation of containers (or virtual machines) can drastically cut power consumption while maintaining service quality by reducing the number of actively running servers. Any container consolidation framework should be able to address the following issues: [2]

- When the host is discovered as being overcrowded and unable to supply the requisite resources for containers and virtual machines running on this host?
- Which containers should be chosen to transfer from an overburdened host?
- When the host is regarded as being underloaded? Is it possible to move all hosted containers and shut down this host?
- How do I select a destination (host/VM) for moved containers?

According to the above questions, there are four sub problems in dynamic container consolidation, in this paper we will focus on the sub-problem of dynamic destination host selection.

The main contribution of the paper is to address the dynamic consolidation of VMs in cloud computing data centres for improved resource utilization and better energy efficiency. A detailed description of the proposed solution and its evaluation results is provided in the paper.

The remainder of the paper is divided into the following sections: Section 2 presents related work, Section 3 provides the motivation of the paper and Section 4 presents the methodology of the proposed work Section 5 analyses the results, and Chapter 6 concludes the paper.

2. RELATED WORK

In comparison to the extensive research on energy efficiency of computing for virtualized cloud data centers, only a few studies investigated the challenge of energy-efficient container management. This work [3] considers the container consolidation problem as a multi-objective optimization problem with the goals of minimizing total energy usage and total container migrations over a given time period. The experimental evaluations using real-world workload confirm that our proposed methodology may reduce the number of container migrations and save energy when compared to alternative methods.

In order to reduce the amount of power consumed by servers during this new deployment model, the authors of this research [4] suggest a system that centralises containers on virtual machines. They introduce the container consolidation issue and then evaluate several algorithms according to several metrics, including energy consumption, SLA non-compliance, average migration rates of containers, and the average number of virtual machines produced. The suggested architecture and algorithms can reduce energy use and virtual machine rental hours in a public cloud.

Data centres in distributed cloud systems must be consolidated, as the author explore in it article [5]. We provide a general overview of cloud service consolidation at various levels of IT infrastructure. To begin, here is a high-level look at virtualized data centres and consolidation. In the next section, the author provide brief topic taxonomy and an illustration of different consolidation solutions that have been published. A discussion of certain research concerns and a proposal for several future directions in this field will follow the presentation, which we believe is vital in order to help resolve the problem addressed in this article. In this research, the author present a prediction model called MR-PSO (Multiple Regression Particle Swarm Optimization) [6].

There are two metrics that are tracked by MR-PSO: (a) the amount of processing time and (b) the quantity of memory usage. By making better use of data centre assets, this model helps cut down on energy costs. The host load prediction model is based on the Multiple Regression (MR) method, and the Particle Swarm Optimization (PSO) technique is provided for establishing the upper and lower limits of host utilization. When running a CloudSim simulation with the same number of hosts, VMs, and tasks, MR-PSO was observed to reduce energy consumption by 7.61 percent and ESV by 1.5 percent.

3. MOTIVATION

As part of the research into server consolidation, several different strategies for detecting and dealing with overloading, selecting and placing virtual machines, and combining these two tasks are presented and executed [5]. While optimizing the CDC's energy consumption or resource utilization efficiency is the end goal of server

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 10 Issue: 1s
DOI: https://doi.org/10.17762/ijritcc.v10i1s.5846
Article Received: 01 September 2022 Revised: 05 November 2022 Accepted: 30 November 2022

consolidation, each component of the plan serves a unique function. There are both ways based on static thresholds and dynamic thresholds that use statistical approaches to determine if a host is overloaded. In order for SLAV to occur, there must be excessive rivalry for resources across VMs, and this competition need not take place simultaneously with the onset of the overloaded state among hosts. Therefore, SLAV does not always occur alongside overloading, but overloading is always present whenever SLAV does. When hosts are identified as likely to overload, some virtual machines (VMs) are moved out of the SLAV state or out of the SLAV state altogether. Therefore, the purpose of VM migration can be adjusted to prevent the incidence of SLAV rather than simply alleviating host congestion. Selecting which virtual machines (VMs) to migrate can be done in one of three ways: using the Minimum Migration Time (MMT) strategy, the Least Number of Migrations (LNM) strategy, or the Migration Correlation (MC) strategy. Many strategies have been proposed to solve the VM placement problem, which has long been a focus of cloud computing research.

4. METHODOLOGY

In this paper, we present a model of energy consumption, represent VM migration and consolidation as a Markov decision process (MDP)[7], and show that minimizing the rise in energy consumption during consolidation is an important objective.

Environment Model

Multiple time intervals, represented by [0,1,2,...t,t+1,...end], extend the cloud cluster system's lifespan. Each time slot has a duration of T seconds. In this cluster, there are N virtual machines spread across M hosts.

Energy Consumption Model

In our energy model, processing unit usage plays a vital role. for vm_i its energy consumption in t is calculated as

$$ec_{vm_{i,t}} = \int_{t\cdot T}^{(t+1)\cdot T} u_{vm_i}(x)\,, \qquad (1)$$

where $u_{vmi}(x)$ is vm_i 's time-based CPU utilization x . The total power usage of all hosts in a certain time interval is retrieved. :t

$$\mathrm{EC}_{\mathrm{host}_t} = \sum_{j=1}^{M} \chi_{j,t} \times \left(\mathrm{base}_{\mathrm{host}_j} + \sum_{vm_i \in h_{j,t}} ec_{vm_{i,t}} \right) \qquad (2)$$

Notably, hostj doesn't use any power. if $\chi_{j,t}$ =0

The expense of migrating during time interval t must be included in[8]. The CPU use of a VM on the source host will increase by 10% while the migration is in progress. As a result, the full expense of virtual machine migration is,

$$\mathrm{MC}_t = 10\% \times \sum_{vm_i \in MIG_t} ec_{vm_{i,t}}. \qquad (3)$$

$$\mathrm{SLAVC}_t = c_{slav} \times \left[\sum_{j=1}^{M} \Upsilon_{j,t} \times \left(T \times \sum_{vm_i \in h_{j,t}} d_{vm_i} \right) \right]. \qquad (4)$$

To find the cluster's cumulative energy consumption over time EC_t, we use Eqs. (2), (3), and (4):

$$EC_t = EC_{hostt} + MC_t + SLAVC_t$$

Due to the complexity of the issue, the dynamic consolidation of virtual machines in cloud computing data centers has been broken down into four phases [7]:

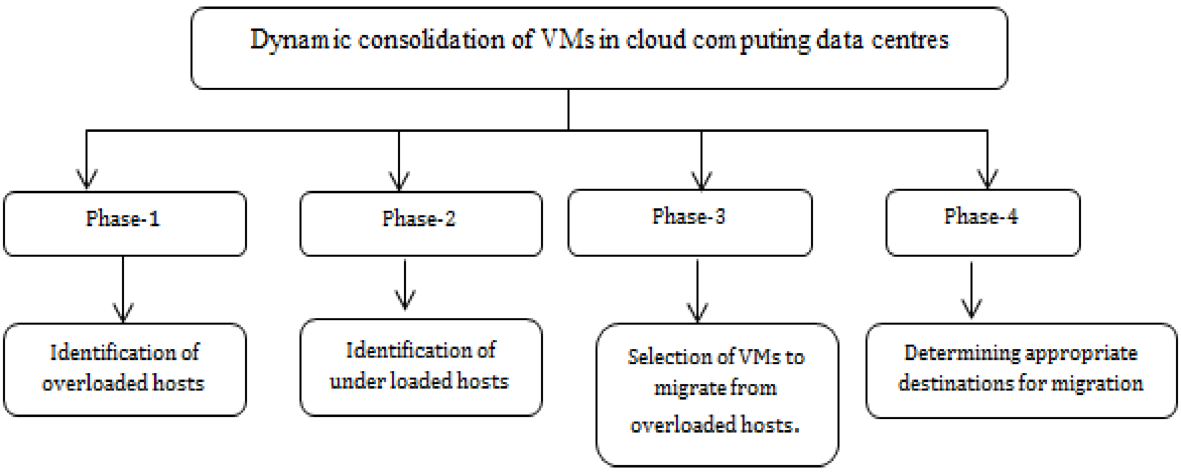

Figure 2. Dynamic consolidation of VMs in cloud computing data centers

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 10 Issue: 1s
DOI: https://doi.org/10.17762/ijritcc.v10i1s.5846
***Article Received**: 01 September 2022 **Revised**: 05 November 2022 **Accepted**: 30 November 2022*

Phase- 1 and 2

An overloaded host is one that has exceeded its capacity and needs to be removed from the global list. Before an SLA violation occurs, it is imperative that overloaded hosts are discovered and remedied[9]. Figure 2 depicts how the CPU usage of each host is measured at regular intervals. The CPU use of each host can also be forecasted for the following time-stamped. Selecting VMs for migration when a host is overcrowded is done by the virtual machine selection module. All virtual machines on one host are moved to another host if that host is projected to be under loaded. An underutilized host can be removed from the active hosts list and put into sleep or hibernation mode once the migration is complete[10]. A host that isn't overloaded or underutilized stays the same until the next time period.

Algorithm :1 Host Overload detection
Input: CPU utilization and Host H_m
Output : *Boolean Return True / False* that Host is Overloaded or Not
Step 1: Function is HostOverUtilized(H_m, *SafetyParameter*)
Step 1.1 $U_h \leftarrow getU_h$
Step 1.2 $U_p \leftarrow U_h$
Step 1.3 $U_p \leftarrow U_p * Safteyparameter$;
Step 1.4 add HistoryEntry (H.U_p)
Step 1.5 if($U_p \geq 1$)*then*
Re*turn* True;
Else Return False;
End Function.

Where U_h Utilization History, U_p Predicted Utilization, H_m Host Machine

Phase 3: Selection of VM

Virtual machines that need to be moved from hosts that are overcrowded are identified in this process. An algorithm must be developed in order to select a group of virtual machines from an overloaded host. The characteristics of the workload, the CPU, the RAM, or the bandwidth consumption can be used to select VMs. In Beloglazov et al. [8], they employed fixed usage thresholds, minimizing migration counts, the highest potential growth, and random choice.

The action is combined by two steps, and in each step, there are corresponding constraints.Regarding VM selection policies, it must obey the following constraint:

VM selection constraint: for each$host_j$,$MIG_{j,t}$,$CH_{j,t}$ which means that the policy selects and only can select the VMs running on$h_{j,t}$ to be migrated.

Input: hostlist, vmList (MIG_t) *Output: allocation of the VMs*

Step 1: *for each VM in vmList do*

Step 2: *minPower* ← *MAX*

Step 3: *allocatedHost* ← *NULL*

Step 4: *for host in hostList do*

Step 5: *if no SLAV on this host and not the source host for VM then*

Step 6: *power* ← *estimatePower* (*host*, *VM*)

Step 7: *if power* < *minPower then*

Step 8: *allocatedHost* ← *host*

Step 9: *manpower* ← *power*

Step 10: *end if*

Step 11: *end if*

Step 12: *end for*

Step 13: *if allocatedHost* ≠ *NULL then*

Step 14 : *allocation.add* (*VM*, *allocatedHost*)

Step 15: *end ifStep* 16: *end for return allocation*

Phase 4:

Authors can remove underutilised hosts from the pool of potential migration targets to simplify the process outlined in [9]. According to the authors of [9], researchers attempted to avoid leaving overloaded and potentially overloaded hosts behind by omitting them and selecting underloaded and moderately burdened sites as migration destinations. As potential migration targets[11], we removed the underutilised hosts as well as the overloaded and potentially unstable ones. The prior analyses determined that the goal was to select VMs with a manageable workload. Data centre energy usage was significantly lowered, underutilised computers were spared from being powered down, and the number of VM migrations was significantly reduced.

4.2 Standard Particle Swarm Optimization

The standard PSO makes use of randomly generated starting particles. Because of this, the algorithm's likelihood of discovering the optimum solution decreases as a result. Implementing efficient initialization solutions can significantly boost its performance [10]. Fig 3 illustrates the methodology we use to choose a container from the migrated container list and assign it to a host.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 10 Issue: 1s
DOI: https://doi.org/10.17762/ijritcc.v10i1s.5846
***Article Received**: 01 September 2022 **Revised**: 05 November 2022 **Accepted**: 30 November 2022*

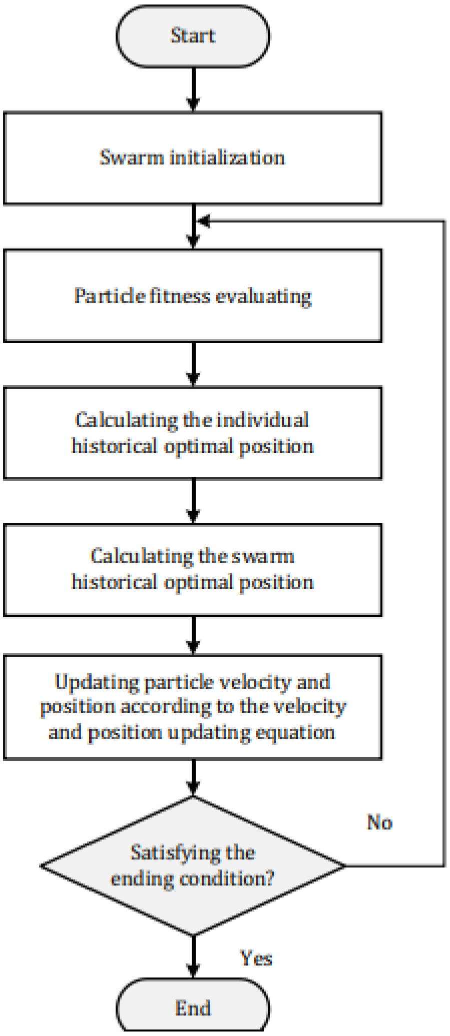

Figure 3.Flow model of Proposed PSO Technique

4.3 Fitness Function

TOPSIS [11] is a multi-criteria method that we employ to create the fitness function for our proposed approach. As a result of this procedure, the optimal solution is determined to be the one that is most distant from the negative-ideal solution and least distant from the positive-ideal solution, in that order of magnitude. The particles are ranked according to four criteria, which are presented in Table 1. First, we compute the value of each parameter for each individual particle in the swarm of particles. Afterwards, these values are normalised by dividing them by the maximum value of each parameter identified in the swarm, which is computed using the following equation: The score of the particle is calculated using the following equation:

$$Score_{Particle_i} = \frac{\bar{D}_{Particle_i}}{\overset{+}{D}_{Particle_i} + \bar{D}_{Particle_i}} \quad (5)$$

This score is regarded as the fitness function value that the algorithm seeks to maximize.

Table-1: PSO Parameters

Number of particles	**100**
Initial inertia weight w	1.4
Minimum Value of w	0.4
Learning factors c1,c2	2
Number of iterations	100

5. PERFORMANCE EVALUATION

5.1 Simulation Setup

We'll utilize the Container CloudSim toolbox [12] to see how well our policy proposal performs. There are 100 PMs, 200 VMs, and more than 1000 containers in our cloud datacenter. Container selection and placement policies are based on MU and First Fit, respectively, and are implemented using these algorithms. Correlation Threshold Host Selection (CorHS) and First Fit Host Selection (FFHS) are compared to our suggested algorithm (EE-PSO)[13][14].

6. RESULTS AND ANALYSIS

. The upper and lower thresholds are set at 80% and 70% in this set of studies. It's possible that each performance metric will return 10 results because there are 10 days of data. The method uses the average of these outcomes as the metric's final result.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 10 Issue: 1s
DOI: https://doi.org/10.17762/ijritcc.v10i1s.5846
***Article Received**: 01 September 2022 **Revised**: 05 November 2022 **Accepted**: 30 November 2022*

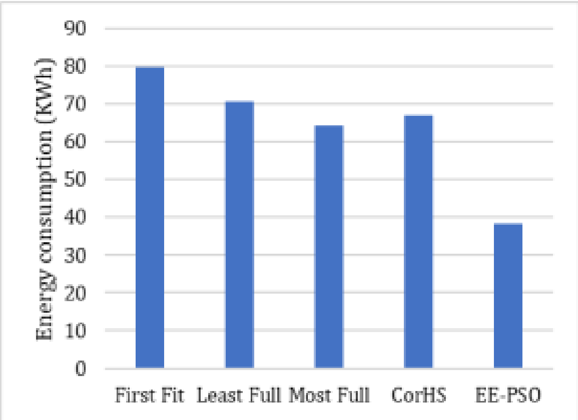

Figure 4: Energy Consumption

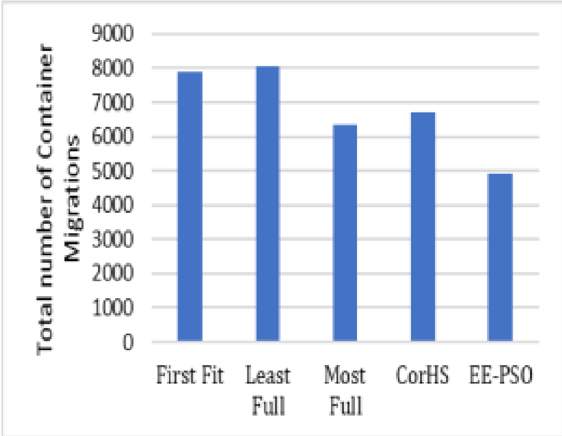

Figure 5. Total number of container migrations

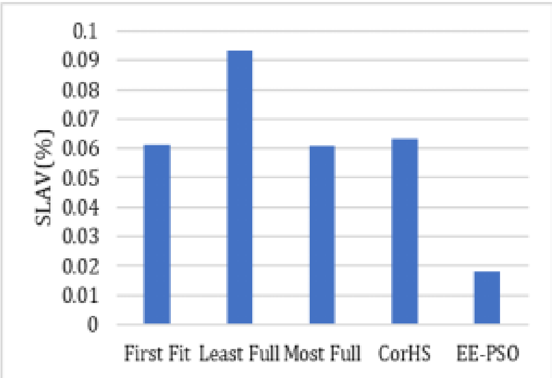

Figure 6: SLAV in scenario

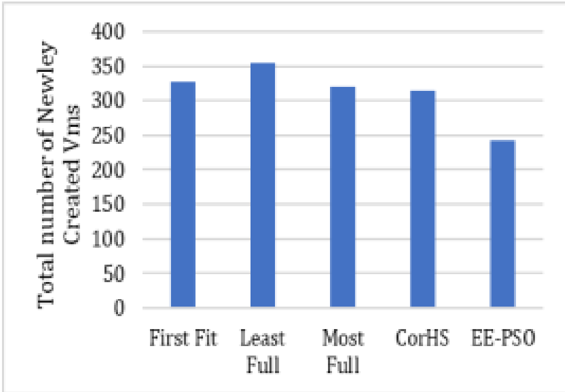

Figure 7: Total number of newly created VMs in scenario

Finally, this algorithm uses the metric's average as its final outcome. The results (Figure [4-7]) show that our proposed method outperforms all other algorithms for all metrics because our algorithm aims to minimize the power consumption in its fitness function. A reduced number of overloaded hosts and SLA breaches, and a higher number of under loaded hosts that will be shut down, are both benefits of EE-container PSO's migration optimization approach[15][16][17]. On the other hand, selecting the most energy efficient hosts means not only minimizing the energy but also higher capacity, so the container will get its required resources and the number of migrations and created VMs will decrease.

6. CONCLUSION AND FUTURE WORK

The energy efficiency of resource management algorithms in the Container as a Service (CaaS) paradigm has gotten little attention, despite the growing popularity of this service model. A unique host selection approach for container consolidation was developed in this research, Which took advantage of particle swarm optimization and the energy efficiency of hosts? The simulation experiments were carried out in order to evaluate the performance of our technique with existing algorithms. As a result, our suggested technique exceeds all competitive methods in terms of energy consumption, total number of migrations, SLAV, and the number of virtual machines (VMs) generated. Improve our algorithm so that it can think about operating system type as a new constraint for the problem, and we'll keep working to solve other sub problems of container consolidation that haven't been solved yet, as well.

REFERENCES

[1]. O. Smimite and K. J. a. p. a. Afdel, "Container placement and migration on cloud system," 2020.

[2]. S. F. Piraghaj, A. V. Dastjerdi, R. N. Calheiros, and R. Buyya, "A framework and algorithm for energy efficient container consolidation in cloud data centers," in 2015 IEEE International Conference on Data Science and Data Intensive Systems, 2015, pp. 368-375: IEEE

[3]. Shi T., Ma H., Chen G. (2018) Multi-objective Container Consolidation in Cloud Data Centers. In: Mitrovic T., Xue B., Li X. (eds) AI 2018: Advances in Artificial Intelligence. AI 2018. Lecture Notes in Computer Science, vol 11320. Springer, Cham. https://doi.org/10.1007/978-3-030-03991-2_71

[4]. Piraghaj, Sareh & Dastjerdi, Amir & N.Calheiros, Rodrigo & Buyya, Rajkumar. (2015). A Framework and Algorithm for Energy Efficient Container Consolidation in Cloud Data Centers. 10.1109/DSDIS.2015.67.

[5]. Ruslan Malikov, Nigora Abdiyeva,Jurabek Abdiyev, International Journal of Engineering and Information Systems (IJEAIS) ISSN: 2643-640X Vol. 5 Issue 3, March - 2021, Pages: 64-87

[6]. Akram Saeed Aqlan Alhammadi, Dr. V. Vasanthi, Multiple Regression Particle Swarm Optimization for Host Overload and Under-Load Detection , The Mattingley Publishing Co., Inc.(2020), January - February 2020, ISSN: 0193 - 4120 Page No. 10253 – 10261.

[7]. A. Greenberg, J. Hamilton, D. A. Maltz, and P. Patel, “The cost of a cloud: research problems in data centre networks,” ACM SIGCOMM computer communication review, vol. 39, no. 1, pp. 68-73, 2008.

[8]. Beloglazov A, Buyya R. Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers. Concur Comput Prac Exper. 2012;24(13):1397-1420.

[9]. Dow EM, Matthews JN. A host-agnostic, supervised machine learning approach to automated overload detection in virtual machine workloads. In: 2017 IEEE International Conference on Smart Cloud (Smartcloud) IEEE; 2017:13-23.

[10]. Wang JV, Ganganath N, Cheng C-T, Chi KT. Bio-inspired heuristics for VM consolidation in cloud data centers. IEEE Sys J. 2019;14: 152-163

[11]. Masdari M, Gharehpasha S, Ghobaei-Arani M, Ghasemi V. Bio-inspired virtual machine placement schemes in cloud computing environment: taxonomy, review, and future research directions. Clust Comput. 2020;23:2533-2563

[12]. Beloglazov A, Buyya R. Energy efficient allocation of virtual machines in cloud data centers. In: 2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing IEEE; 2010:577-578

[13]. Jasuja, V., & Singh, R. K. (2019). Enhanced MBFD algorithm to minimize energy consumption in cloud. *International Journal of Computer Engineering in Research Trends*, *6*(2), 266–271. doi:10.22362/ijcert/2019/v6/i02/v6i0202

[14]. Shere, R., Shrivastava, S., & Pateriya, R. K. (2017). CloudSim Framework for Federation of identity management in Cloud Computing. *International Journal of Computer Engineering in Research Trends*, *4*(6), 269–276.

[15]. PRAVEEN KUMAR, & S.NAGA LAKSHMI. (2015). Efficient Data Access Control for Multi-Authority Cloud Storage using CP-ABE. *International Journal of Computer Engineering in Research Trends*, *2*(12), 1182-1187. Retrieved from https://www.ijcert.org.

[16]. Prakash, P. S., Janardhan, M., Sreenivasulu, K., Saheb, S. I., Neeha, S., & Bhavsingh, M. (2022). Mixed linear programming for charging vehicle scheduling in large-scale rechargeable WSNs. *Journal of Sensors*, *2022*, 1-13. doi:10.1155/2022/8373343

[17]. Yedukondalu, G., Samunnisa, K., Bhavsingh, M., Raghuram, I. S., & Lavanya, A. (2022). MOCF: A multi-objective clustering framework using an improved particle swarm optimization algorithm. *International Journal on Recent and Innovation Trends in Computing and Communication*, *10*(10), 143-154. doi:10.17762/ijritcc.v10i10.5743