Heliyon 5 (2019) e02066

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.heliyon.com

# Energy-aware virtual machines allocation by krill herd algorithm in cloud data centers

Minoo Soltanshahi [a,*], Reza Asemi [b], Nazi Shafiei [c]

[a] Department of Computer Science and Engineering, Payame Noor University(PNU), Kerman, Iran
[b] Department of Software Engineering, Science and Research Kermanshah Branch, Kermanshah, Iran
[c] Department of Software Engineering, Islamic Azad University, Kerman, Iran

ARTICLE INFO

*Keywords:*
Computer science
Green computing
Cloud computing
Virtualization
Data center
Krill herd algorithm

ABSTRACT

The growing demand for computational power has led to the emergence of large-scale data centers that consume massive amounts of energy, thus resulting in high operating costs and CO2 emission. Furthermore, cloud computing environments are required to provide a high Quality of Service (QoS) to their clients and, therefore, need to handle power shortages. An optimized virtual machine allocation to physical hosts lowers energy consumption and allows for high-quality services. In this study, a novel solution was proposed for the allocation of virtual machines to physical hosts in cloud data centers using the Krill Herd algorithm, which is the fastest collective intelligence algorithm recently introduced. The performance of the proposed method was evaluated using the *CloudSim* simulator, and the results are suggestive of a 35% reduction in energy consumption.

## 1. Introduction

Energy consumption has increased exponentially in the IT sector in recent years. Data centers, as the main component of information and communication technology, have proliferated at an unprecedented rate as IT developers including IBM, Microsoft, Google, and other similar large corporations have expanded data centers in recent years to support their cloud computing and grid computing services. These data centers are equipped with thousands of servers and switches that use up massive amounts of energy, thus raising operating costs and increasing carbon dioxide emission into the environment. On the other hand, cooling equipment must be used to handle the heat produced by these data centers, which also consumes energy [1, 2]. Several studies have shown that servers, in the idle state, consume as much as 70 percent of the energy they use at peak demand. Therefore, it is not economically wise to leave servers running with a small workload. Methods used to detect idle servers are studied in the field of host domain with a low amount of tracking algorithm. Therefore, maintaining servers with lower efficiency is costly in terms of energy consumption [3]. Reducing energy consumption is a significant challenge in cloud computing. The aim is not only to reduce energy consumption but also to take into account the environmental regulations and standards and contracts between users and service providers. Given the fact that the virtualization technology, as well as the unification and live migration of virtual machines, can dramatically reduce energy consumption, designing data center with this approach has attracted much attention.

Anton Beloglazov et al. [4] presented an energy-conscious method for allocating virtual machines based on answering three central questions:

1) When should virtual machines migrate?
2) Which virtual machines will migrate?
3) Where should virtual machines migrate to?

The same structure was adopted in this study, presenting an optimal method for the third subproblem based on the Krill Algorithm (KH) a fast, recently-introduced metaheuristic algorithm. In order to have an accurate evaluation, famous underloaded host detection algorithms, single-threshold algorithms, over-loaded physical host detection algorithms, Interquartile Range algorithm (IQR), and the Median Absolute Deviation Algorithm (MAD) were used. MAD [5] is a powerful statistical analysis that offers the highest flexibility to distant points in a set compared to the standard deviation. Moreover, the Interquartile Range algorithm (IQR) [5], also referred to as the midspread or middle fifty-fifty, is a measure of statistical dispersion in descriptive statistics. The remainder of this paper is organized as follows. Section 2 discusses the previously-proposed methods, Section 3 elaborates on the proposed method, and Sections 4

* Corresponding author.
*E-mail address:* minoo.soltanshahi@gmail.com (M. Soltanshahi).

https://doi.org/10.1016/j.heliyon.2019.e02066
Received 25 October 2018; Received in revised form 6 April 2019; Accepted 5 July 2019

and 5 present the simulation results and conclusions, respectively.

## 2. Related work

Much research work was focused on lowering energy consumption, which, similar to our approach, were based on a number of metaheuristic algorithms. A Round Robin method, referred to as RR Dynamic, was presented [6] for scheduling and aggregating virtual machines and was used to place migrating virtual machines. The simulation results indicate that RR Dynamic reduces energy consumption by 34.7% compared to Round Robin. In 2016, a new, genetic-algorithm-based approach was presented [7] to find an optimal solution to multidimensional bin packing problems aiming to improve resource allocation and the integration of cloud virtual machines. Moreover, a novel and effective evolutionary approach was proposed [8] for Virtual Machine (VM) allocation to reduce the energy consumption that was capable of maximizing energy efficiency in a cloud data center while incorporating more reserved VMs. This approach can provide fast access to an optimized allocation solution for a batch of reserved VMs, while at the same time, consolidate more VMs with fewer physical machines to achieve better energy efficiency than with existing methods. Basu et al. (2019), presented a method that focused on an improved Genetic Algorithm (GA), and improper load balancing will lead to losses in terms of both memory and energy consumption [9]. Alharbi et al. [10] proposed an Ant Colony System (ACS) embedded with new heuristics as an energy-efficient solution to the optimization problem formulating the placement of VMs to Physical Machines in a data center as a constrained combinatorial optimization problem. This is similar to previous attempts at curtailing energy consumption, including one by Hongjian Li et al. (2015) who focused on a modified particle swarm optimization (PSO) method for a VMs reallocation algorithm that reduces energy consumption of the entire system and designed a double-threshold method with multi-resource utilization to trigger the migration of VMs [11]. A similar VM scheduling method, by the name of GRANITE, based on the greedy algorithm was proposed [12] which is capable of minimizing the overall data center energy consumption by analyzing the temperature distribution in the airflow and the CPU.

Two algorithms, known as the "Combinatorial Ordering First-Fit Genetic Algorithm" (COFFGA) and" Combinatorial Ordering Next Fit Genetic Algorithm (CONFGA), were developed and combined. The aim was to minimize the total number of running servers and the waste of resources in each server. In comparison to latest solutions from similar studies, their results indicated that the proposed COFFGA algorithm had an optimal performance compared to multidimensional bin packing schemes, such as Permutation Pack (PP), First Fit (FF), and First Fit Decreasing (FFD), improving the results by 4%, 31%, and 39%, respectively. Moreover, this algorithm provides better performance compared with available GAs for multi-capacity aggregation resources of the virtual machine (VM) aggregation in terms of performance and robustness.

In 2012, Wang et al. [13] attempted to improve the Bin-Packing algorithm by introducing a threshold to prevent inconvenient VM migration. In this scheme, if the total resources of virtual machines from a host are below the threshold value, VMs on the host would migrate to another one with sufficient resources. This method takes into account neither exchange costs nor migration costs, ignores the increase in the potential of using a host that enhances the productivity of the host, and can even violate the Service-Level Agreement (SLA).

In their 2012 study, Anton Beloglazov and et al. [14, 15] introduced the MBFD algorithm for which simulation results show that dynamic VM reallocation techniques and switching off small idle servers can help in energy saving. Their methodology has great potential since as far as productivity is concerned, it offers excellent performance as regards response time and energy cost savings.

Marco Dorigo [16] also presented a termite-algorithm-based method for the dynamic VM reallocation. The results of the study are suggestive of a 10% improvement in energy efficiency by using the proposed method under the first scenario compared to the PSO and genetic algorithms, and 13.33% and 2.5% improvements under the second scenario, compared to PSO and GA, respectively.

Görent et al. (2012) [17] suggest that the use of virtualization technology is important for improving the energy efficiency of data centers and that VM placement is the key in server consolidation. The results of the study indicate that the GA performs 3.5–5.5% better than the FFD.

Moreover, Yang Quiang and et al. (2012) proposed an algorithm based on Simulated Annealing that aimed to improve problem insertion. The simulation results show the SA algorithm to be 25% more energy-efficient than the FFD algorithm [18]. In 2013, Giriano and et al. proposed an approach based on the self-Adaptive Particle Swarm Optimization (SAPSO) algorithm for VM allocation to a set of servers in a dynamic pool. The algorithm reduced the overall processing energy consumption without compromising on the main objectives [19].

Furthermore, given their many applications, researchers have been consistently working to improve the performance of cloud data centers [20, 21]. However, several issues still remain in this area, including the different energy consumption optimization algorithms, using VMs to optimize energy consumption, and energy consumption reduction solutions for cloud data centers.

In a 2015 study, Xiong FU and Chen ZHOU [22] proposed a new VM selection policy (MP) which takes into account resource satisfaction and can reduce energy consumption, VM migration time, SLA violation, as well as a VM placement policy (MCC) to find the target host with the smallest correlation coefficient with the migratable VM.

## 3. Instrumentation

### 3.1. Service-Level Agreement (SLA)

Service-Level Agreement (SLA) specifies the level of service quality agreed between the user and the cloud service provider and states all service features that must be provided and, consequently, the relevant policies to be adopted. An SLA generally relies on the response time, or how fast responses must be made to requests, as a performance metric [23].

### 3.2. Energy SLA Violation(ESV)

Energy SLA Violation (ESV) is an important characteristic in measuring the usage of VM allocation to physical hosts. In fact, ESV shows the overall behavior of algorithm in the point of energy consumption and the number of SLA contract violations and its aim is to reduce it [24]. ESV is calculated from Eq. (1):

$$\mathrm{ESV} = \mathrm{Energy(kw/h)} * \mathrm{SLA}\ (\%) \tag{1}$$

### 3.3. Energy consumption

Power consumption by physical machines can be accurately described by a linear relationship of CPU utilization [24]. They also point out that a free physical machine uses about 70% of its energy consumption when it is fully utilized. Due, the CPU utilization rate in the physical host Pj is obtained according to Eq. (2):

$$\mu j = \frac{P_j^{w_{cpu}}}{P_j^{cpu}} \tag{2}$$

Thus the energy consumption of the physical host can be calculated using Eq. (3):

$$\mathrm{E}\left(\mathrm{p_j}\right) = \mathrm{k_j}.\mathrm{e_j^{max}} + \left(1 - \mathrm{k_j}\right).\mathrm{e_j^{max}}.\mu_\mathrm{j} \tag{3}$$

## 4. Methods

We aim to obtain an appropriate scheme for allocating resources to physical hosts using relying on the krill algorithm. In addition to the time complexity of the algorithm, energy consumption is reduced in data centers as a result of the load congestion of the system at any given moment. Given that VM allocation to physical hosts is an NP-Hard problem, metaheuristic algorithms are the best solution strategies to take. Accordingly, the krill algorithm was employed in this study.

The krill algorithm (Fig. 1) is the metaheuristic with the best performance among similar algorithms and is used in the approaches to solving these types of problems. Therefore, this algorithm was used for acceleration and to provide a convenient solution with a more efficient function than existing solutions.

Also Fig. 2 shows the basic representation of the KH algorithm.

In the cloud computing environment, there are heterogeneous and dynamic virtual machines. In the proposed krill-algorithm-based method, krill represent virtual machines and the baits represent physical machines. Therefore, virtual machines in the V set are defined by Eq. (4):

$$V = \{V1, V2, \ldots, Vm\} \tag{4}$$

Where m denotes the total number of virtual machines and the set of physical machine is defined by Eq. (5):

$$H = \{H1, H2, \ldots, Hn\} \tag{5}$$

Where n denotes the total number of physical machines.

This study addresses the problem of mapping virtual machines to physical hosts in a way that each virtual machine is assigned to only one physical host while making sure the minimum number of physical hosts are running. To solve the problem of allocating virtual machines to physical hosts, each answer was assumed to be a net which was modeled by an array (Table 1). As evident from Table 1, the index represents the virtual machine number and the number inside it refers to the number corresponding to the physical host on which the virtual machine will be placed. In other words, if Index 2 reads 4, it suggests VM 2 will be placed on the physical host no. 4.

Vm_ID = Index.

Host_ID = Value.

In the proposed method, an array is attributed to each krill to represent the length of the net traveled by the krill. The first index indicates the array corresponding to the point where each krill starts its travel, whereas the last index shows the destination or the food location. The proposed method consists of the following steps:

```
1.  Start
2.  Initialize parameters
3.  Make individuals and initialization
4.  For all iterations do
    4.1. For all krill do
         4.1.1.  Movement induced
         4.1.2.  Foraging motion
         4.1.3.  Physical diffusion
         4.1.4.  Crossover & mutation
         4.1.5.  Update krill position
         4.1.6.  Calculate fitness
         4.1.7.  Compare with best krill for
                 replacement
5.  Return best krill
```

**Fig. 1.** Pseudocode of krill algorithm [24].

1) Initialization.
2) A physical host is selected randomly as the starting point for each krill. In fact, for the first index, the array corresponding to each krill is a random value representing the physical host number.
3) The first krill selects a physical host for each virtual machine, and when it is finished pairing hosts with virtual machines, the update the krill location locally.
4) If the new answer is optimal compared with existing answers, save it and update the krill location globally.
5) If all krill have gone through the above steps, enter Step 6; otherwise, return to Step 3.
   NH = NH+1
6) If $NH > NH_{max}$, then return the best answer as output; otherwise, return to Step 2.
   NH = Number of physical hosts.

### 4.1. Selection mechanism

The Kth krill selects a host from the order of physical hosts defined in the list, assigning a virtual machine to the source j according to the probability from Eq. (6). In Eq. (6), $T_j$ (t) is the distance from the food source j at time t.

$$p_j^k(t) = \frac{[t_j\ (t)]^a.[n_j]^\beta}{\Sigma[t_j\ (t)]^a.[n_j]^\beta} \tag{6}$$

Where $n_j$ is the inherent capacity of the source j, α is the parameter controlling the influence of $T_j$ (t), and β is a parameter for controlling the effect of $n_j$.

### 4.2. Global update

When every krill has found a solution to the VM allocation problem, krill positions (foraging distances) corresponding to all physical hosts selected by the best krill with the lowest energy consumption can be updated using Eq. (7):

$$t_j^{new} = (1-\rho)*t_j + \Delta t_j \tag{7}$$

In the above equation, $\Delta\tau_j$ increases, and ρ represents the coefficient of distance to the food. Moreover, the size of the foraging distance is limited to facilitate convergence.

### 4.3. Calculating the fitness criterion

In order to determine whether a solution can provide a convenient answer, it must be evaluated by a fitness function that attributes a value to the solution based on the parameters effective in the quality of the solution. By applying this function to all solutions produced by the proposed algorithm, the value of each solution is calculated, and the solution with the best value which can be either the maximum or minimum based on the parameter placement policy is accepted as the most convenient one. For example, the parameters of Tables 2, 3, 4, 5, and 6 can be used.

Therefore, the fitness function for measuring the value of each answer is presented in Eq. (8):

$$\text{fitness} = \frac{1}{\sum_1^N E(p_j)} \tag{8}$$

## 5. Experimental

To carry out the simulations, the CloudSim toolkit, an extensible simulation tool, was used that helps to model and simulate cloud computing systems and prepare applications. CloudSim is capable of

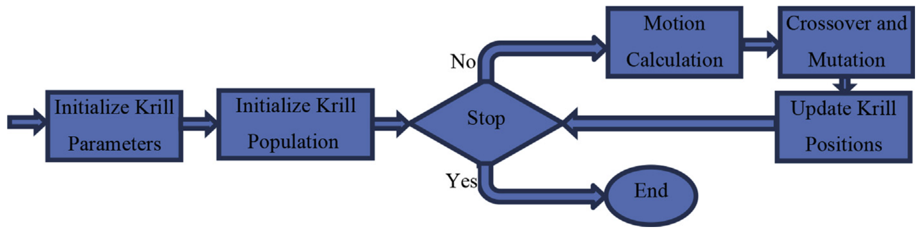


**Fig. 2.** A flowchart of the krill herd algorithm.

**Table 1**
Example of the answer or net used in the proposed algorithm.

| Value | 5 | 2 | 4 | ….. | N |
|---|---|---|---|---|---|
| Index | 0 | 1 | 2 | ….. | M |

**Table 2**
Parameters used in the proposed method.

| *Parameters* | *Description* |
|---|---|
| *V* | *Virtual machines* |
| *P* | *Physical host* |
| *ViCpu* | The required processor for virtual machine (i) |
| *ViMem* | The amount of required memory for virtual machine (i) |
| *PjCpu* | The processing power for physical host (j) |
| *Pjmem* | The amount of required memory for physical host(i) |
| *Pjwcpu* | The total CPU workload of physical host(j) |
| *Pjwmem* | The total amount of used memory for physical host(j) |

**Table 3**
Data center specifications.

| Scenarios | VMs | Physical Hosts | Data Center |
|---|---|---|---|
| A | 290 | 100 | 1 |
| B | 1175 | 800 | 1 |

**Table 4**
Virtual machine specification.

| Scenarios | RAM(MB) | CPU | MIPS |
|---|---|---|---|
| A | 128 | 1 | 250–1000 |
| B | 613–3840 | 1 | 500–2500 |

**Table 5**
Physical hosts specifications.

| Scenarios | RAM(MB) | CPU | MIPS |
|---|---|---|---|
| A | 8192 | 1 | 1000–3000 |
| B | 4096 | 2 | 1860–2660 |

**Table 6**
Works specifications.

| Scenarios | MIPS)Tasks) | Number of Tasks |
|---|---|---|
| A | 250–1000 | 290 |
| B | 500–2500 | 1175 |

modeling the system and behavior of the cloud computing system components, including data centers, virtual machines, resource policies, and application delivery techniques [25]. A review of the previous works revealed that the traditional and widely-used genetic and MBFD algorithms are some of the best methods in this field, which offered excellent results. Therefore, for a more accurate evaluation, the proposed method was simulated and compared with the said algorithms. In addition, to allow for a better comparison of the algorithms, two scenarios (A and B) were used that drew their data from existing studies [26], and since evolutionary algorithms are based on random answers, each algorithm was executed in ten runs with each scenario the average was taken as the result of the proposed method.

In the following, the three algorithms (the proposed algorithm, the GA, and the MBFD algorithm) are presented separately under both scenarios using the IQR and MAD algorithms, and the results are presented in charts and tables. Figs. 3, 4, and 5 illustrate the energy consumption, the violation of SLA contracts, and the Energy SLA Violation (ESV) "Energy consumption and the number of violations in SLA contracts", which is a key factor in measuring the use of VM allocation to physical hosts and is obtained by multiplying energy consumption in kW.h by SLA contracts in percent (Eq. 1), in the first scenario.

As illustrated by the diagrams, the proposed method offers better

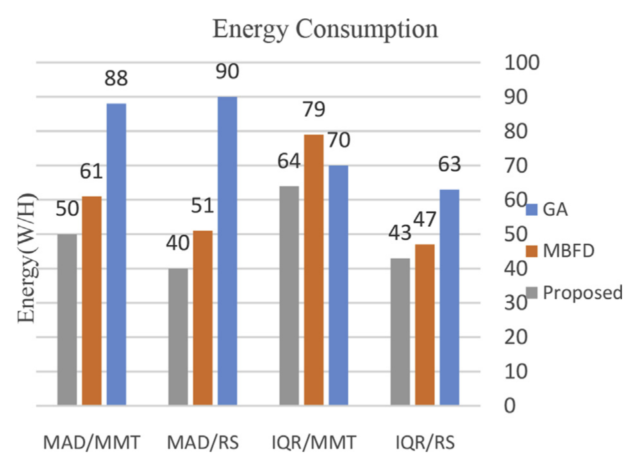


**Fig. 3.** Average rate of energy consumption in scenario A.

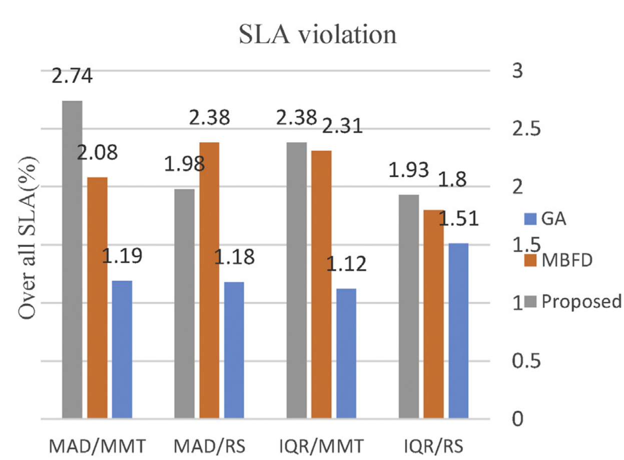


**Fig. 4.** The average rate of SLA violation in scenario A.

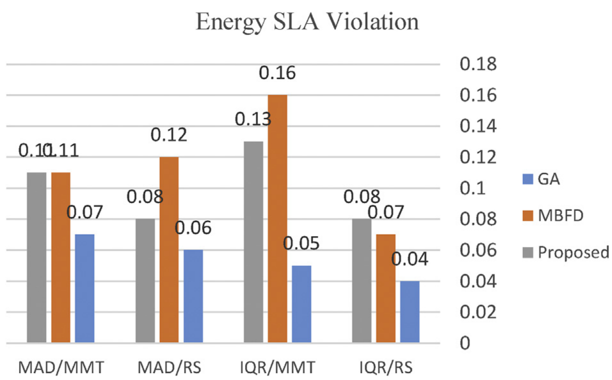


**Fig. 5.** Average of ESV in scenario A.

**Table 7**
Percentage of improvement in energy consumption of suggested algorithm compared to genetic algorithm.

| | MMT | RS |
|---|---|---|
| IQR | 8.57 | 31.74 |
| MAD | 43.18 | 55.55 |
| AVG | %35 | |

**Table 8**
Percentage of improvement in energy consumption of suggested algorithm compared to MBFD algorithm.

| | MMT | RS |
|---|---|---|
| IQR | 18.98 | 8. 89 |
| MAD | 18.03 | 21.56 |
| AVG | %17 | |

performance, reducing energy consumption by 35% and 17% compared with the genetic and MBFD algorithms, respectively. Tables 7 and 8 show the improvement of energy consumption by the proposed algorithm in comparison with genetic and MBFD algorithms, with IQR and MAD as overloaded host detection, random policy search, and minimum migration time algorithms, for VM selection in the ten runs.

Moreover, Figs. 6 and 7 illustrate energy consumption and SLA and ESV violations with IQR and MAD as overloaded host detection algorithms under the second scenario.

As regards the results under the second scenario (Tables 9 and 10), it is also evident that the proposed algorithm performs better, as the tables indicate a 9–10% improvement in energy consumption compared to the genetic and MBFD algorithms, with IQR and MAD as overloaded host detection, random selection, and minimal migration time algorithms for VM selection in the ten runs.

Tables 11 and 12 show the minimum energy consumption in each algorithm for allocating virtual machines to physical hosts and to improve the proposed method compared to other methods, respectively.

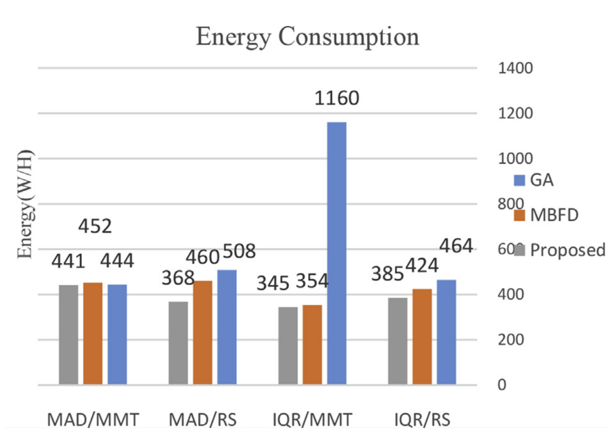


**Fig. 6.** Average of energy consumption in scenario B.

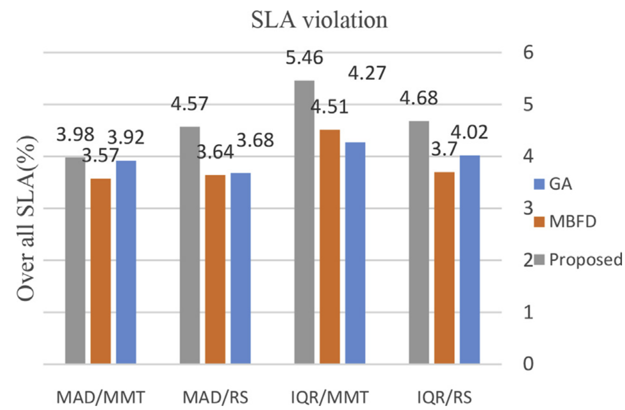


**Fig. 7.** Average of SLA violation in scenario B.

**Table 9**
Percentage improvement of energy consumption of suggested algorithm compared to genetic algorithm.

| | MMT | RS |
|---|---|---|
| IQR | 70.48 | 17.02 |
| MAD | 0.67 | 27.55 |
| AVG | 29% | |

**Table 10**
Percentage improvement of energy consumption of suggested algorithm compared to MBFD algorithm.

| | MMT | RS |
|---|---|---|
| IQR | 2.54 | 9.19 |
| MAD | 2.4 | 20.19 |
| AVG | 9% | |

**Table 11**
Minimum energy consumption for each algorithm for virtual machine insertion into physical hosts.

| Scenarios | Energy consumption | Algorithms | | |
|---|---|---|---|---|
| A | 49.25 | MMT-RS | IQR-MAD | Proposal Method |
| | 63.5 | MMT-RS | IQR-MAD | MBFD |
| | 77.75 | MMT-RS | IQR-MAD | GA |
| B | 387 | MMT-RS | IQR-MAD | Proposal Method |
| | 422.5 | MMT-RS | IQR-MAD | MBFD |
| | 644 | MMT-RS | IQR-MAD | GA |

**Table 12**
Percentage improvement of proposed method compared to other methods.

| Scenarios | MBFD | Genetic |
|---|---|---|
| A | 17% | 35% |
| B | 9% | 29% |

## 6. Conclusion

Green computing is a considerable challenge in modern cloud computing that puts cloud service providers under pressure. A review of the previous solutions to this problem revealed that more satisfactory answers can be obtained. Accordingly, a method was presented to curtail energy consumption by optimizing the aggregation of virtual machines and shutting down idle servers, without compromising on the service quality. The simulation results show that an efficient integration and the selection of convenient virtual machine migration strategies can help improve energy efficiency. A comparison of the proposed algorithm with genetic and MBFD algorithms with IQR and MAD as overloaded host detection, random selection, and minimum migration time algorithms for virtual machine selection in the ten runs shows energy consumption to be reduced by 35% and 17%, respectively. Moreover, using deep learning algorithms can be helpful in the timely diagnosis of host overload and is recommended to interested researchers for further study and evaluation.

## Declarations

### *Author contribution statement*

Minoo Soltanshahi: Conceived and designed the experiments; Performed the experiments; Analyzed and interpreted the data; Contributed reagents, materials, analysis tools or data; Wrote the paper.

Reza Asemi: Conceived and designed the experiments; Performed the experiments.

Nazi Shafiei: Contributed reagents, materials, analysis tools or data.

### *Funding statement*

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

### *Competing interest statement*

The authors declare no conflict of interest.

### *Additional information*

No additional information is available for this paper.

## References

[1] S. Mustafa, B. Nazir, A. Hayat, A. ur Rehman Khan, S. Madani, Resource management in cloud computing: taxonomy, prospects, and challenges, Comput. Electr. Eng. (29 July 2015).
[2] D. Liu, X. Sui, in: "An Energy-Efficient Virtual Machine Placement Algorithm in Cloud Data Center", 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery, ICNC-FSKD, 2016, pp. 719–723.
[3] A. Beloglazov, Energy-efficient management of virtual machines in data centers for cloud computing, Submitted in total fulfilment of the requirements of the degree of Doctor of Philosophy, Department of Computing and Information Systems. The University of Melbourne, 2013.
[4] A. Beloglazov, B. Rajkumar, Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers, Concurr. Comput. Pract. Ex. 24 (13) (2012) 1397–1420.
[5] B. Guenter, N. Jain, C. Williams, Managing cost, performance and reliability tradeoffs for energy-aware server provisioning, in: INFOCOM, Proceedings IEEE, April 2011, pp. 1332–1340.
[6] C.-C. Lin, P. Liu, J.-J. Wu, Energy-aware virtual machine dynamic provision and scheduling for cloud computing", in: IEEE 4th International Conference on Cloud Computing, 2011, pp. 736–737.
[7] H. Hallawi, J. Mehnen, Hongmei He, Multi-Capacity Combinatorial Ordering GA in Application to Cloud resources allocation and efficient virtual machines consolidation, Future Gener. Comput. Syst. 10–025 (2016).
[8] Xinqian Zhang, Tingming Wu, Mingsong Chen, Tongquan Wei, Junlong Zhou, Shiyan Hu, Rajkumar Buyya, Energy-aware virtual machine allocation for cloud with resource reservation, J. Syst. Softw. 147 (2019) 147–161.
[9] Sayantani Basu, G. Kannayaram, Somula Ramasubbareddy, C. Venkatasubbaiah, Improved genetic algorithm for monitoring of virtual machines in cloud environment, in: S.C. Satapathy, et al. (Eds.), Smart Intelligent Computing and Applications, Smart Innovation, Systems and Technologies, 105, Springer Nature Singapore Pte Ltd, 2019.
[10] Fares Alharbi, Yu-Chu Tian, Maolin Tang, Wei-Zhe Zhang, Chen Peng, Minrui Fei, An Ant colony system for energy-efþcient dynamic virtual machine placement in data centers, Exp. Sys. Appl. (2018).
[11] Hongjian Li, Guofeng Zhu, Chengyuan Cui, Hong Tang, Yusheng Dou, He Chen, Energy-efficient migration and consolidation algorithm of virtual machines in data centers for cloud computing, Computing (2015). Springer-Verlag Wien.
[12] X. Li, P. Garraghan, X. Jiang, et al., (2 more authors), "holistic virtual machine scheduling in cloud datacenters towards minimizing total energy", IEEE Trans. Parallel Distrib. Syst. 29 (6) (2018) 1317–1331. ISSN 1045-9219.
[13] X. Xiaoli, Z. Zhanghui, An energy-aware VMs placement algorithm in cloud computing environment, Int. Conf. Intell. Syst. Des. Eng. Appl. (2012) 627–630.
[14] A. Beloglazov, J. Abawajy, R. Buyya, Energy-aware resource allocation heuristics for efficient management of data centers for Cloud computing, Future Gener. Comput. Syst. 28 (2012) 755–768.
[15] A. Beloglazov, R. Buyya, in: "Energy Efficient Allocation of Virtual Machines in Cloud Data Centers", 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing, The University of Melbourne, Australia, 2010.
[16] M. Dorigo, C. Blum, Ant colony optimization theory: a survey, Theor. Comput. Sci. 344 (2005) 243–278.
[17] Grant Wu, Maolin Tang, Yu-Chu Tian, Wei Li, Energy- efficient virtual machine placement in data centers by genetic algorithm, in: Lecture Notes on Computer Science, Springer, Renaissance Doha City Center Hotel, Doha, 2012, 323-315.
[18] Y. Wu, M. Tang, W. Fraser, A simulated annealing algorithm for energy efficient virtual machine placement, IEEE Int. Conf. Syst. Man Cybern. (October 2012) 1247–1250.
[19] R. Jeyarani, N. Nagaveni, R. Vasanth, Self adaptive particle swarm optimization for efficient virtual machine provisioning in cloud, Int. J. Intell. Inf. Technol. 7 (25-44) (2011).
[20] N. Akhter, M. Othman, Energy aware resource allocation of cloud data center: review and open issues, Clust. Comput. 19 (2016) 1163–1182.
[21] N. Moganarangan, R.G. Babukarthik, S. Bhuvaneswari, M.S. Saleem Basha, P. Dhavachelvan, A novel algorithm for reducing energy-consumption in cloud computing environment: web service computing approach, J. King Saud Univ. Comput. Inform. Sci. 28 (2015) 55–67.
[22] F.U. Xiong, Z.H.O.U. Chen, Virtual machine selection and placement for dynamic consolidation in Cloud computing environment", Publ. Front. Comput. Sci. 9 (2) (2015) 322–330.
[23] Tilley Scott, Parveen Tauhida, Migrating Software Testing to the Cloud", 26th IEEE International Conference on Software Maintenance in Timisoara, Romania, IEEE, 2010. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5610422.
[24] A.H. Gandomi, A.H. Alavi, Krill herd: A new bio-inspired optimization algorithm.", Commun. Nonlinear Sci. Numer. Simul. 17 (2012) 4831–4845.
[25] R.N. Calheiros, R. Ranjan, A. Beloglazov, C.A.F. DeRose, R. Buyya, CloudSim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms, Software Pract. Ex. 41 (2011) 23–50.
[26] Yongqiang Wu, "Energy Efficient Virtual Machine Placement In Data Centers.", Masters by Research Thesis, Queensland University of Technology, 2013.