ISSN 2007-9737

Energy Efficient Virtual Machine Placement in Dynamic Cloud Milieu Using a Hybrid Metaheuristic Technique

Bivasa Ranjan Parida[1], Amiya Kumar Rath[1,2], Bibudhendu Pati[3], Chhabi Rani Panigrahi[3], Hitesh Mohapatra[4], Rajkumar Buyya[5]

[1] Veer Surendra Sai University of Technology,
Department of Computer Science and Engineering,
India

[2] Biju Patnaik University of Technology,
India

[3] Rama Devi Women's University,
Department of Computer Science,
India

[4] KIIT University, School of Computer Engineering,
India

[5] University of Melbourne,
Cloud Computing and Distributed Systems (CLOUDS),
Australia

bivasa.parida@silicon.ac.in, akrath_cse@vssut.ac.in, patibibudhendu@rdwu.ac.in,
{panigrahichhabi, hiteshmahapatra}@gmail.com, rbuyya@unimelb.edu.au

Abstract. Energy consumption in cloud datacenters is an alarming issue in recent times. Although handful of researches have been conducted in this domain during virtual machine placement in cloud milieu, efficient techniques are still scarce. Hence, we have worked on a novel approach to propose a hybrid metaheuristic technique combining the salp swarm optimization and emperor penguins colony algorithm, i.e. SSEPC to place the virtual machines in the most suitable datacenters as well as servers in a cloud environment, while optimizing the energy consumption. The method we propose has been compared with certain relevant hybrid algorithms in this direction like Sine Cosine Algorithm and Salp Swarm Algorithm (SCA-SSA), Genetic Algorithm and Tabu-search Algorithm (GATA), and Order Exchange & Migration algorithm and Ant Colony System algorithm (OEMACS) to test its efficacy. It is found that proposed SSEPC is consuming 4.4%, 8.2%, and 16.6% less energy as compared to its counterparts GATA, OEMACS, and SCA-SSA respectively.

Keywords. Cloud computing; salp swarm optimization; emperor penguins colony algorithm; energy consumption; virtual machine placement.

1 Introduction

In the modern high-tech world, cloud computing has emerged as a crucial component of technology. It has proved itself to be a utility as pay-per-use model to deliver the resources to the cloud users from the datacenters virtually through Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS)

[1]. The significant cost of expense in the datacenters is the cost of consumption of energy [2]. Energy consumption in the cloud datacenters is increasing rapidly. Costs associated with consuming energy in the datacenters of Amazon Web Services measures nearly 42% of the total operational price [3].

According to the prediction of Natural Resources Defense Council (NRDC), the amount of electricity consumed by the cloud datacenters annually was assumed to be projected to 140 billion kilowatt-hours by 2020, analogous to the yearly production of electricity by nearly 50 numbers of power plants, costing around $13 billion in the form of electricity bills [4].

If the remedial measures have not taken and the technological trends remain unchanged for the coming decade, then between 2016 and 2030, the energy consumption by the datacenters may exceed with 12% approx. [5].

The strategy of virtual machine placement (VMP) directly influences the consumption of energy in a cloud datacenter [6].

Although a handful of researches conducted to minimize the consumption of energy in cloud datacenters, efficient methodologies are still scarce. Addressing this issue through VMP in the cloud datacenters is becoming the area of interest for researchers globally.

Hence, we are motivated to contribute a little to this vast field of emerging aspect of research in cloud milieu. This piece of work is devoted in exploring the strategy of VMP in minimizing the usage of energy in the cloud datacenters. The following points are the key contributions of this work.

- The proposed methodology introduces a hybrid nature inspired optimization technique for solving the problem of VMP in the cloud milieu. At first the Salp Swarm Algorithm (SSA) [7] is used to explore geographically and then exploit to find out the best datacenter possible for VMP.

  An algorithm of recent times inspired by the Emperor Penguins Colony (EPC) [8], enriched with the potential of local search is implemented to trace the most suitable server in the datacenter for placing the virtual machine (VM).

  Hence, our proposed hybrid algorithm Salp Swarm based Emperor Penguins Colony (SSEPC) is found to be better in terms of overcoming the limitations of each other to establish a robust methodology in comparison to other metaheuristic techniques.
- Unlike other existing VMP proposals, our model portrays both initial as well as the runtime VMP in the cloud milieu.

This paper is organized into different sections as follows. The literature of VMP along with consumption of energy in cloud milieu is reviewed in Section 2.

The proposed strategy along with the mathematical formulations of the proposed work are elucidated in Section 3.

Next, the Section 4 deliberates the simulation environment, as well as analyzes the experiments conducted in heterogeneous cloud environment and the obtained results.

Finally, the conclusive remarks along with the future directions in the research carried out in this domain are illuminated in the Section 5.

2 Related Work

We have gone through the existing works related to VMP in the cloud milieu. Some researchers [9-12] worked on intra datacenter VMP, i.e. VMP in the servers of a single datacenter, while others [6, 13-16] proposed methodologies for VMP in inter datacenter or multi datacenter environments. Jayasinghe *et al.* [17] stated that the process of VMP is further divided into two separate stages, i.e. (i) initial VMP and (ii) runtime VMP.

Unlike most of the other VMP proposals, we have considered both initial as well as runtime VMP in inter datacenter environment in our proposed work. Most of the authors [6, 9-16, 18-20] considered energy consumption in the datacenters as the basic QoS parameter due to the alarming situation of energy consumption and scarcity due to limited energy sources.

Authors in [6, 9-16, 20] prioritized the servers as the basic cause of energy consumption in the datacenters and proposed various techniques for its minimization. We have also prioritized the servers as the root cause of energy consumption.

Table 1. Existing Hybrid Algorithms for the Virtual Machine Placement in Cloud Milieu

Authors	Proposed Algorithm	Algorithms involved in Hybridization	Purpose of Hybridization
Liu *et al.* [9]	OEMACS	Order Exchange and Migration algorithm and the algorithm of Ant Colony System	OEM swaps VMs between servers locally, whereas ACS allocates VMs in minimum number of servers through global search
Zhao *et al.* [11]	GATA	Genetic Algorithm and Tabu-search Algorithm	The algorithm of Tabu-search works like a mutation operator of genetic algorithm for improving its local search ability
Gharehpasha *et al.* [14]	SCA-SSA	Sine Cosine Algorithm and Salp Swarm Algorithm	SCA explores and exploits the space of searching for finding the optimum solution, whereas SSA manages leader and followers in the population to reach at the optimal solution

The physical servers even in the idle state consume more than two-third of the energy, i.e. mostly 50-70% of the active state energy usage by the servers [21].

Hence, the insufficient utilization of the server with the waste of energy during idle state results in high energy consumption in the datacenters. Various strategies, such as dynamic voltage or frequency scaling, VM consolidation, and shutting down of servers or scheduling to sleep mode are implemented for the reduction of energy consumption in the cloud datacenters [22, 23].

We have forced the underutilized servers to switch into the sleep mode in this proposed work. An algorithm, which solves a problem set efficiently, may not be able to solve the problems of a different set. None of the existing algorithms is adequate enough for solving all optimization problems, which is proved by the No-Free-Lunch (NFL) [24].

This fact inspires in developing new metaheuristic techniques. New algrthms prove themselves to be better in solving problems as compared to the existing ones.

Hence, we have hybridized two recent metaheuristic techniques SSA (2017) [7] and EPC (2019) [8] to search the global space for VMP. Hybridizing multiple methodologies helps in overcoming the limitations of one another [25].

Table 1 analyzes the existing hybrid algorithms chosen for comparison with the proposed approach. Here, in our proposed methodology, SSA algorithm is implemented to explore the search space globally and exploit the positions further to find out the suitable datacenter for VMP in the Phase-I of SSEPC.

After this, the EPC metaheuristic technique enriched with the capability of local search is implemented in the second phase of SSEPC to find out the most appropriate server in the datacenter to host the VMs.

We have hybridized both the above-mentioned metaheuristic techniques to improve the robustness of the searching paradigm in the vast cloud-based search space with the objective to minimize energy consumption.

The proposed SSEPC algorithm is of multi-stage collaborative hybrid type, which hybridizes two stochastic collective algorithms based on swarm intelligence.

3 Proposed Methodology

The proposed algorithm along with the mathematical formulations of the proposed work are elucidated in this section.

The process of VMP can be briefly outlined into some steps, such as (1) Users submit the requests to allocate VMs for the execution of single or multiple tasks, (2) The VMP system invokes the SSEPC algorithm for selecting the most suitable datacenter and server for placement of the VMs, (3) The SSEPC based VMP method is executed for minimizing the energy consumption and (4) The data of resource utilization is updated after the execution of VMP.

3.1. Mathematical Formulations

Let the set of m datacenters is represented as a set $D = \{d_1, d_2, d_3, \dots, d_m\}$, where each $d_i \in D$, such that $1 \le i \le \mathrm{m}$.

Each datacenter has n number of physical servers, i.e. $S = \{\mathrm{s}_1, \mathrm{s}_2, \mathrm{s}_3, \dots, \mathrm{s}_\mathrm{n}\}$, where $\mathrm{s_j} \in S$, such that $1 \le \mathrm{j} \le \mathrm{n}$.

Each server allocates p number of VMs portrayed as set $VM = \{vm_1, vm_2, vm_3, \dots, vm_p\}$, where $vm_k \in VM$, such that $1 \le k \le \mathrm{p}$.

R is the set of q requests, i.e. $R = \{r_1, r_2, r_3, \dots, r_q\}$, where $r_l \in R$, such that $1 \le l \le \mathrm{q}$.

The consumption of energy by a server $\mathrm{s_j}$ in a datacenter d_i can be represented as $E_{i,\mathrm{s_j}}$ and is defined by the Eqn. (1) [26]:

$$E_{i,\mathrm{s_j}} = E_{j,Idle} + \left(E_{j,Max} - E_{j,Idle}\right) \times u_{\mathrm{s_j}}, \quad (1)$$

Here, $E_{j,Idle}$ and $E_{j,Max}$ are the energy consumed by the server $\mathrm{s_j}$ at idle time and peak time respectively. $u_{\mathrm{s_j}}$ is the utilization of j^{th} server.

The energy consumption of a datacenter (E_{d_i}) may be considered as the energy consumed in total by all the servers of that datacenter, as mentioned in the Eqn. (2):

$$E_{d_i} = \sum_{j=1}^{n} E_{i,\mathrm{s_j}}, \quad (2)$$

The energy consumed by all the datacenters E_D is formulated as in Eqn. (3):

$$E_D = \sum_{i=1}^{m} E_{d_i}, \quad (3)$$

The capacity of a VM vm_k in a server s_j, i.e. Cap_{j,vm_k} can be portrayed as given in Eqn. (4) [27]:

$$Cap_{j,vm_k} = num(PEs)_{vm_k} \times MIPS(PEs)_{vm_k} + BW_{vm_k}. \quad (4)$$

where $num(PEs)_{vm_k}$ is denoted as the count of processing elements in k^{th} VM, $MIPS(PEs)_{vm_k}$ stands for the million instructions per second of all the processing elements of k^{th} VM, and BW_{vm_k} represents the bandwidth of k^{th} VM.

Hence, the capacity of a server s_j in a datacenter d_i can be represented as in Eqn. (5):

$$Cap_{i,s_j} = \sum_{k=1}^{p} Cap_{j,vm_k}, \quad (5)$$

On regular intervals, the proposed technique calculates the values of energy consumption and capacity of the servers of the datacenter for the VMP.

3.2. Proposed SSEPC Algorithm

In the first phase of SSEPC, exploration and exploitation are executed to find out the appropriate datacenter for VMP. In 2017, Mirjalili *et al.* proposed SSA as an optimization technique to solve the design problems in engineering [7]. It got inspired from the foraging nature of a creature called salp living in the deep ocean.

In our proposed model, we have implemented the principle of SSA for resolving the issue of VMP in the dynamic cloud milieu. The user requests to create VMs are considered as the salps in the population foraging the food sources, where the computing resources in the datacenters mimic the foods.

The salps compete among themselves to discover the best source of food. Similarly, the requests also compete for availing the most suitable datacenters with respect to the minimization of consumption of energy.

From a user, the first request to create a VM is treated as the leader salp. Then the other requests from the same user behave like follower salps on the basis of updated data received from the leader.

A list is maintained and updated on regular intervals to monitor the consumption of energy by the datacenters. The list contains the datacenters in increasing order of energy consumption.

The datacenter consuming minimum energy is chosen for VMP. The first phase of proposed SSEPC is presented in the Algorithm 1.

Algorithm 1: Algorithm of **SSEPC Phase-I**
Input: The set D of datacenters d_i with size m, set R of requests r_l with size q
Output: Requests are allocated to appropriate datacenters

```
for r_l in R do
    for d_i in D do
        Add d_i to the sorted list D^E, where
        datacenters are sorted in the
        ascending order on the basis of
        consumption of energy
```

```
        end for
        Allocate $r_l$ to $d_0$ in $D^E$
        Set $r_l$ of user_account_id ($r_l$) as the
        leader request in the request queue
        user_account_id= user_account_id ($r_l$)
        for $r_{l+1}$ in $R$ do
            if user_account_id ($r_{l+1}$) = =
            user_account_id then
                $r_{l+1}$ is the follower request
                Allocate $r_{l+1}$ to $d_0$
                Delete $r_{l+1}$ from $R$
            end if
            $l = l + 1$
        end for
        Delete $r_l$ from $R$
    end for
```

After the first phase of SSEPC, the second phase is executed on the basis of emperor penguins colony algorithm to search the most suitable server in a datacenter to host the virtual machines. Sasan Harifi *et al.* have proposed a new metaheuristic technique in 2019 on the basis of the behavior of emperor penguins [8].

We have mimicked the herd of the penguins as the server and the penguins as the VMs. A penguin intends to be warmed goes to a nearby huddle randomly. Likewise, a VM is allotted to a server in the datacenter arbitrarily. If the huddle has maximum number of penguins, then the new penguin arrived recently will remain at the periphery and can't be warmed enough.

So, this penguin will again move to another huddle having less than maximum number of penguins to be warmed appropriately. But if a huddle contains very less number of penguins, then the penguins loss a lot of energy and enough heat can't be generated in that huddle. So, those penguins prefer to move to an existing huddle with sufficient capacity to be accommodated, by dissolving their huddle.

Similarly, a new VM migrates to an appropriate server accommodating less than maximum VMs, as it can't be allocated with resources by a server already reached to its maximum capacity. Likewise, if a server is allocated with very less number of VMs, it lets the VMs to migrate to another servers and goes to sleep mode for conserving energy.

This is the process of VM migration in running VMP. Servers accommodate the VMs till they have not reached the upper bound threshold value (Th_{UB}) of a server's capacity (Cap_{i,s_j}). If Cap_{i,s_j} will go beyond Th_{UB} after adding the new VM, then instead of allocating resources to that VM, the server will forward it to the VM migration list. Similarly, if Cap_{i,s_j} fall below the lower bound threshold value (Th_{LB}) in a server, then the allocated VMs join the migration list of VMs and this server goes to sleep mode for conserving energy. On the basis of priority, the VMs in the VM migration list will be allocated to other suitable servers. In this way we have implemented the VM migration technique for considering the running VMP in Phase-II along with the initial VMP conducted in the Phase-I of SSEPC.

Algorithm 2 represents the VM migration in running VMP among the servers of a datacenter in the second phase of proposed SSEPC metaheuristic technique.

Algorithm 2: Algorithm of **SSEPC Phase-II**

Input: The set S of servers s_j with size n, the set VM of virtual machines vm_k with size p, upper threshold of server capacity Th_{UB}, and lower threshold of server capacity Th_{LB}.

Output: Runtime VM migration for VMP inside the datacenter to allocate the VMs to the appropriate servers.

```
for $s_j$ in $d_i$ do
    if $Cap_{i,s_j} < Th_{LB}$ then
        for $vm_k$ in $s_j$ do
            Insert $vm_k$ to vm_migration list in
            descending order of priority
        end for
        $s_j$ goes to $sleep()$ mode
    else if $Cap_{i,s_j} > Th_{UB}$ then
        for $vm_k$ in $s_j$ do
            if $vm_k$ is requesting for scaling
            up resources then
                Insert $vm_k$ to vm_migration
                list in descending order of
                priority
            end if
        end for
    else
```

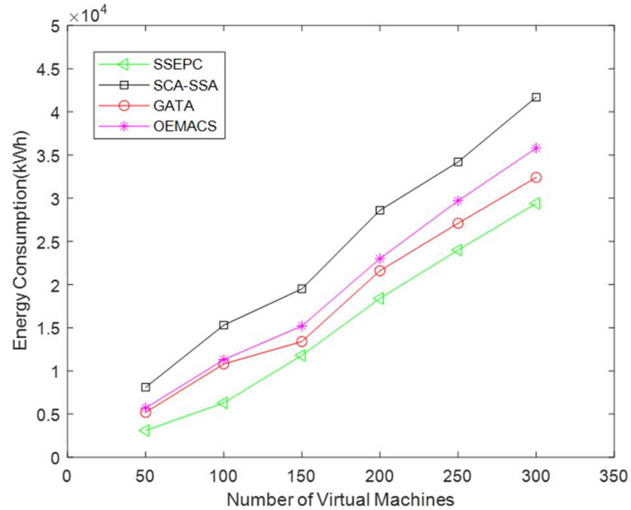

Fig. 1. Comparison of Energy Consumption of SSEPC with other VMP Algorithms

```
            j = j + 1
        end if
    end for
    for vm_k in vm_migration list do
        for s_j in d_i do
            if Th_LB ≤ Cap_{i,s_j} ≤ Th_UB then
                Allocate vm_k to s_j
            else
                j = j + 1
            end if
        end for
    end for
```

4 Performance Evaluation

This section illustrates the setup for the experimental work for the evaluation of the efficiency of proposed technique in comparison to some other metaheuristic techniques for VMP. As it is difficult to evaluate on real cloud infrastructure, we have conducted our experiments using CloudSim [28].

Here, we have considered each datacenter comprising of four numbers of servers with dissimilar configurations further virtualized for creating an abstract and isolated environment. 300 VMs with disparate specifications are deliberated for mapping onto the servers.

This research considers a heterogeneous cloud milieu, where the number of servers concurrently vary with the number of VMs to endorse the efficacy of the proposed metaheuristic technique. The energy usage is considered for validating the proposed work.

For evaluating the efficacy of SSEPC VMP algorithm, we have chosen SCA-SSA, OEMACS, and GATA as three recent hybrid algorithms of VM placement proposed by Gharehpasha *et al.* [14], Liu *et al.* [9], and Zhao *et al.* [11] respectively for comparison with our proposed methodology.

As these three algorithms are evolved by the hybridization of two other existing algorithms just like the hybridization applied in our proposed work, we have chosen them as the appropriate contemporaries of our work in the same direction for comparison.

After simulating our experiments in CloudSim, we have obtained the results for validating our work. In SCA-SSA, for the exploration and the exploitation of the search space, SCA is utilized for finding the optimum solution.

Similarly, ACS allocates VMs through global search in OEMACS. Genetic algorithm executes the searching mechanism for finding the optimal solution, where tabu search enhances the local search ability of genetic algorithm as a mutant in GATA.

In contrast, our proposed methodology SSEPC applies the divide and conquer principle, where it divides the entire search space broadly into two parts and implements two algorithms in multi-stage collaborative approach for them separately.

Hence, we found our proposed methodology outperforming its counterparts in VM placement strategy in the cloud milieu. Fig. 1 shows, SSEPC consumes less energy as compared to SCA-SSA, GATA, and OEMACS.

Here, the maximum value of consumption of energy is taken as 5×10^4 KWh from the analysis of the obtained results with the variation of the number of VMs as 50, 100, 150, 200, 250, and 300 for the mentioned algorithms considered for comparison.

The values of average energy consumption while implementing SSEPC, GATA, OEMACS, and SCA-SSA are found to be 1.48×10^4 KWh, 1.7×10^4 KWh, 1.89×10^4 KWh, and 2.31×10^4 KWh respectively.

It is already stated above that SSEPC is more efficient in searching than others, for which it avails most suitable resources with better utilization. SSEPC also implements the runtime VMP. So, it produces optimum result in energy consumption in comparison to other algorithms.

5 Conclusion and Future Work

This paper proposes a multi-stage collaborative hybrid algorithm to solve the problem of VMP in the cloud milieu. The proposed SSEPC methodology searches the suitable datacenter in the first phase by utilizing salp swarm optimization. In the second phase, it implements emperor penguins colony algorithm to search locally in a datacenter to find the most suitable server to place the VMs.

Along with initial VMP, runtime VMP is also applied in this strategy. The energy consumption is considered for the evaluation of efficacy of the proposed hybrid metaheuristic technique in comparison to some recent VMP algorithms, where it outperforms others.

As part of our future work, the cost of energy consumption and carbon footprint cost along with calculating their amounts to analyze the evaluation from both qualitative and quantitative perspective can be included. Renewable energy sources can also be utilized to move towards a greener cloud environment. Hybridization of some other techniques in this direction can be explored to optimize the metrics better.

References

1. **Ullman, J. D. (1975).** NP-complete scheduling problems. Journal of Computer and System sciences, Vol. 10, No. 3, pp. 384–393.
2. **Hamilton, J. (2010).** Perspectives-overall data center costs. Web Link: http://perspectives. mvdirona.com/2010/09/18/OverallDatacentre Costs. aspx.
3. **Tarahomi, M., Izadi, M., Ghobaei-Arani, M. (2021).** An efficient power-aware VM allocation mechanism in cloud data centers: a micro genetic-based approach. Cluster Computing, Vol. 24, pp. 919–934. DOI: 10.1007/s10586-020-03152-9.
4. **Heidari, S., Buyya, R. (2019).** Quality of Service (QoS)-driven resource provisioning for large-scale graph processing in cloud computing environments: Graph Processing-as-a-Service (GPaaS). Future Generation Computer Systems, Vol. 96, pp. 490–501. DOI: 10.1016/j.future.2019.02.048.
5. **Koot, M., Wijnhoven, F. (2021).** Usage impact on data center electricity needs: A system dynamic forecasting model. Applied Energy, Vol. 291. DOI: 10.1016/j.apenergy.20 21.116798.
6. **Khosravi, A., Andrew, L. L., Buyya, R. (2017).** Dynamic VM placement method for minimizing energy and carbon cost in geographically distributed cloud data centers. IEEE Transactions on Sustainable Computing, Vol. 2, No. 2, pp. 183–196. DOI: 10.1109/ TSUSC.2017.2709980.
7. **Mirjalili, S., Gandomi, A. H., Mirjalili, S. Z., Saremi, S., Faris, H., Mirjalili, S. M. (2017).** Salp swarm algorithm: A bio-inspired optimizer for engineering design problems. Advances in engineering software, Vol. 114, pp. 163–191. DOI: 10.1016/j.advengsoft.2017.07.002.
8. **Harifi, S., Khalilian, M., Mohammadzadeh, J., Ebrahimnejad, S. (2019).** Emperor penguins colony: a new metaheuristic algorithm for optimization. Evolutionary

Intelligence, Vol. 12, No. 2, pp. 211–226. DOI: 10.1007/s12065-019-00212-x.

9. **Liu, X. F., Zhan, Z. H., Deng, J. D., Li, Y., Gu, T., Zhang, J. (2016).** An energy efficient ant colony system for virtual machine placement in cloud computing. IEEE transactions on evolutionary computation, Vol. 22, No. 1, pp. 113–128. DOI: 10.1109/TEVC.2016.2623803.
10. **Elsedimy, E. I., Algarni, F. (2021).** Toward enhancing the energy efficiency and minimizing the SLA violations in cloud data centers. Applied Computational Intelligence and Soft Computing, Vol. 2021, pp. 1–14. DOI: 10.1155/2021/8892734.
11. **Zhao, D. M., Zhou, J. T., Li, K. (2019).** An energy-aware algorithm for virtual machine placement in cloud computing. IEEE Access, Vol. 7, pp. 55659–55668. DOI: 10.1109/ACCESS.2019.2913175.
12. **Samriya, J. K., Chandra-Patel, S., Khurana, M., Tiwari, P. K., Cheikhrouhou, O. (2021).** Intelligent SLA-aware VM allocation and energy minimization approach with EPO algorithm for cloud computing environment. Mathematical Problems in Engineering, Vol. 2021, pp. 1–13. DOI: 10.1155/2021/9949995.
13. **Xu, M., Buyya, R. (2020).** Managing renewable energy and carbon footprint in multi-cloud computing environments. Journal of Parallel and Distributed Computing, Vol. 135, pp. 191–202. DOI: 10.1016/j.jpdc.2019.09.015.
14. **Gharehpasha, S., Masdari, M., Jafarian, A. (2021).** Power efficient virtual machine placement in cloud data centers with a discrete and chaotic hybrid optimization algorithm. Cluster Computing, Vol. 24, No. 2, pp. 1293–1315. DOI: 10.1007/s10586-020-03187-y.
15. **Le, K., Bianchini, R., Nguyen, T. D., Bilgir, O., Martonosi, M. (2010).** Capping the brown energy consumption of internet services at low cost. International Conference on Green Computing, IEEE, pp. 3–14. DOI: 10.1109/GREENCOMP.2010.5598305.
16. **Feng, H., Deng, Y., Li, J. (2021).** A global-energy-aware virtual machine placement strategy for cloud data centers. Journal of Systems Architecture, Vol. 116, pp. 102048. DOI: 10.1016/j.sysarc.2021.102048.
17. **Jayasinghe, D., Pu, C., Eilam, T., Steinder, M., Whally, I., Snible, E. (2011).** Improving performance and availability of services hosted on IaaS clouds with structural constraint-aware virtual machine placement. 2011 IEEE International Conference on Services Computing, IEEE, pp. 72–79. DOI: 10.1109/SCC.2011.28.
18. **Wang, L., Zhang, F., Vasilakos, A. V., Hou, C., Liu, Z. (2014).** Joint virtual machine assignment and traffic engineering for green data center networks. ACM SIGMETRICS performance evaluation review, Vol. 41, No. 3, pp. 107–112. DOI: 10.1145/2567529.2567560.
19. **Belabed, D., Secci, S., Pujolle, G., Medhi, D. (2015).** Striking a balance between traffic engineering and energy efficiency in virtual machine placement. IEEE Transactions on Network and Service Management, Vol. 12, No. 2, pp. 202–216. DOI: 10.1109/TNSM.2015.2413755.
20. **Mhedheb, Y., Jrad, F., Tao, J., Zhao, J., Kołodziej, J., Streit, A. (2013).** Load and thermal-aware vm scheduling on the cloud. In: Kołodziej, J., Di Martino, B., Talia, D., Xiong, K. (eds) Algorithms and Architectures for Parallel Processing, ICA3PP 2013, Lecture Notes in Computer Science, Vol. 8285, Springer, Cham. DOI: 10.1007/978-3-319-03859-9_8.
21. **Ammar, A. M., Luo, J., Tang, Z., Wajdy, O. (2019).** Intra-balance virtual machine placement for effective reduction in energy consumption and SLA violation. IEEE Access, Vol. 7, pp. 72387–72402. DOI: 10.1109/ACCESS.2019.2920010.
22. **Kim, S., Park, S., Kim, Y., Kim, S., Lee, K. (2017).** VNF-EQ: dynamic placement of virtual network functions for energy efficiency and QoS guarantee in NFV. Cluster Computing, Vol. 20, No. 3, pp. 2107–2117. DOI: 10.1007/s10586-017-1004-3.
23. **Beloglazov, A., Buyya, R. (2012).** Managing overloaded hosts for dynamic consolidation of virtual machines in cloud data centers under quality of service constraints. IEEE transactions on parallel and distributed systems, Vol. 24, No. 7, pp. 1366–1379. DOI: 10.1109/TPDS.2012.240.

24. **Wolpert, D. H., Macready, W. G. (1997).** No free lunch theorems for optimization. IEEE, Transactions on evolutionary computation, Vol. 1, No. 1, pp. 67–82. DOI: 10.1109/4235.585893.
25. **Ting, T. O., Yang, X. S., Cheng, S., Huang, K. (2015).** Hybrid metaheuristic algorithms: past, present, and future. Recent advances in swarm intelligence and evolutionary computation, pp. 71–83. DOI: 10.1007/978-3-319-13826-8_4.
26. **Basmadjian, R., Ali, N., Niedermeier, F., De -Meer, H., Giuliani, G. (2011, May).** A methodology to predict the power consumption of servers in data centres. Proceedings of the 2nd international conference on energy-efficient computing and networking, pp. 1–10. DOI: 10.1145/2318716.2318718.
27. **Babu, L. D. D., Krishna P. V. (2013).** Honey bee behavior inspired load balancing of tasks in cloud computing environments, Applied Soft Computing Journal, Vol. 13, No. 5, pp. 2292–2303. DOI: 10.1016/j.asoc.2013.01.025.
28. **Calheiros, R. N., Ranjan, R., Beloglazov, A., De-Rose, C. A., Buyya, R. (2011).** 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. DOI: 10.1002/spe.995.

Article received on 10/06/2023; accepted on 19/10/2023.
Corresponding author is Chhabi Rani Panigrahi.