Hybrid Best-Fit Heuristic for Energy Efficient Virtual Machine Placement in Cloud Data Centers

Saikishor Jangiti[1] , Vijayakumar V[2] and Subramaniyaswamy V[1,*]

[1]School of Computing, SASTRA Deemed University, Thanjavur, Tamilnadu, India.
[2]Lead Data Scientist, Briteyellow Ltd., United Kingdom.

Abstract

Cloud Service Providers (CSPs) offers Information Technology services like infrastructure and software to users on a pay as you go basis. Energy consumption is one of the significant challenges faced by Cloud Service Providers (CSP). Virtual Machine (VM) placement is an energy-efficient practice performed in the cloud datacenters. Best-Fit Decreasing (BFD) is a VM placement and is known to give a near-optimal solution in a reasonable time by sorting the VMs in decreasing order. We propose a Hybrid Best-Fit (HBF) Heuristic for VM placements. Experimental results show that HBF is consuming 2.516% and 3.392% less energy compared to Best-Fit and BFD heuristics.

* Corresponding author. Email:vsubramaniyaswamy@gmail.com, jsaikishor@gmail.com

1. Introduction

Cloud computing is a growing technology which allows Cloud Service Providers (CSPs) to satisfy on-demand resource requests. Acquiring Resources is a tedious process in a computing environment; cloud computing eases this process. CSPs provide resources. These resources can be accessed anytime and anywhere by users through the Internet at a minimum cost. This scalability is one of the reasons why many users prefer cloud computing. Users need not worry about maintenance cost since CSPs take maintenance care. Also, other advantages include portability, increased storage, backup, recovery, etc.

There are four different types of clouds which can be deployed according to the needs of the user: Private cloud, Community Cloud, Public Cloud and Hybrid Cloud. A particular organization uses private cloud whereas a community of users uses community cloud. More substantial groups use public Cloud. Hence its resources are owned by government sectors or larger business sectors. Hybrid Cloud is a combination of the other three types.

The three primary services provided by Cloud are Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (Iaas) [1]. SaaS allows Cloud Service Providers (CSP) to deliver software applications to users. Users can use the software for a specific amount of time for which they will be charged. PaaS allows CSPs to rent out the environment to users for developing applications. PaaS can also be used testing, monitoring and managing software applications. IaaS allows CSPs to rent out infrastructure components like servers, storage, Virtual Machines, databases, etc. Users can subscribe to these services according to their necessity and cost. While providing services through Virtual Machines, several factors affect the performance of the system. Virtual Machine Placement is critical for system performance. Physical Machines should not be overloaded with Virtual Machines because it results in performance degradation. This is called overutilisation of a resource. Moreover, PM should not be underutilised since it leads to wastage of resources. There are several capacity planning tools like VMWare Capacity Planner[1], Lanamark Suite[2], etc. to monitor resource utilisation.

When a user makes a request, the request will be satisfied by a Cloud Service Provider (CSP) through instances which users can use for a certain amount of time and a certain charge. Failure to place VMs effectively increases operating costs, wastage of resources. Sometimes overloading will lead to SLA violation. Also, the underutilisation of PM causes an increase in the number of active PMs. This causes an increase in energy consumption. Hence Hybrid-Best-Fit (HBF) heuristic reduces energy costs.

The organisation of the paper is as follows. Section 2 briefs about the existing algorithms. Section 3 briefs about the proposed HBF, Section 4 discuss experimental setup and discussions. Section 4 provides experimental results and Section 5 provides a conclusion and future works.

2. Related work

VM placement can be either static or dynamic. In static VM placement, a set of VM requests are submitted. A set of PMs with known capacities is already known. Then each VM is tried to place in a suitable PM. Here the number of active PMs is tried to reduce during allocation. In Dynamic VM placement, VM requests are not known in advance. Hence the requests are placed on arrival.[2]

VM placement problem can be categorised into two: Power-based approach and QoS-based Approach. In the Power-based approach, during VM placement importance is given to power consumption reduction whereas in QOS-based Approach, SLA regulations are given importance.[3]

The variable-sized bin packing problem (VSBPP) is solved using an enhanced levy based particle swarm optimisation algorithm (PSOLBP) in [4]. OpenStack Neat is using Modified Best-Fit Decreasing (MBFD) algorithm for VM placement. A novel bin-packing heuristic Medium-Fit (MF) is proposed in [5] to reduce SLA violation and to run PMs.

Two VM placement algorithms based on chemical reaction optimisation algorithm were proposed in [6], namely CVP and CVV. These algorithms exhibited better results in terms of resource consumption and resource utilisation. SLA-aware Modified Best Fit Decreasing (MBFD) algorithm, an energy and SLA Aware resource allocation heuristic algorithm is proposed in [7]. MBFD reduces the power consumption of VMs using minPower and maxUtilization VM migration policies.

A hybrid genetic algorithm for minimising the running PMs is designed in [8]. Best Fit Multi-Valued Bin-packing Algorithm is designed in [9] for optimum allocation of resources like RAM, CPU and network bandwidth. Best Fit Sharing and Power-Aware (BFSPA) Algorithm for VM placement designed in [10] find the VM and PM combinations that will minimise the PMs used. Space aware best fit decreasing (SABFD) heuristic policy proposed in [11] for the management of

---

[1] https://www.vmware.com/products/capacity-planner.html
[2] http://www.lanamark.com/

host-overload detection, migration and VM placement. The whale optimisation algorithm is improvised with the levy and applied for VM placement as a variable-sized bin-packing algorithm for bandwidth optimisation [12].

LIFE-MP (Lowest Interdependence Factor Exponent Multiple Resources Predictive approach) for online VM placement is proposed in [13]. The particle swarm optimisation is used in [14] to reduce the active physical servers for energy-efficient management of data centres.

In our recent works, we proposed the following VM placement heuristics FFD-Aggregated rank [15], MCBVP [16], PMNeAR-Vector[17], BBPMM [18] and ENSEMBLE-HIDE-SPADE [19]. Best Fit Decreasing algorithm sorts the VM requests in descending order and allocates the largest VM first to active PM that gives the minimum residue. It reduces the number of active PMs by temporarily pushing the inactive PMs to rest mode. It tries to fit the VM request with the active PMs [20].

For example, consider a set of PMs with capacities <75, 74, 87, 52, 81>. A set of VMs requests <16, 8, 30, 22, 18, 19, 8, 24, 22> are submitted. These VM requests have to satisfied by those PMs in the PMList.

Initially, the VMList is sorted in decreasing order. This results in <30, 24, 22, 22, 19, 18, 18, 16, 8, 8>. Then the VM at the first of the list (here 30) is selected to find a suitable PM. The first VM if placed at PM4 leaves minimum residue comparatively. Hence VM1 is allocated to PM4. The process is iterated until the entire list of VMs is allocated to some PM in the PMList. Figure 1 shows the results of Best-Fit decreasing.
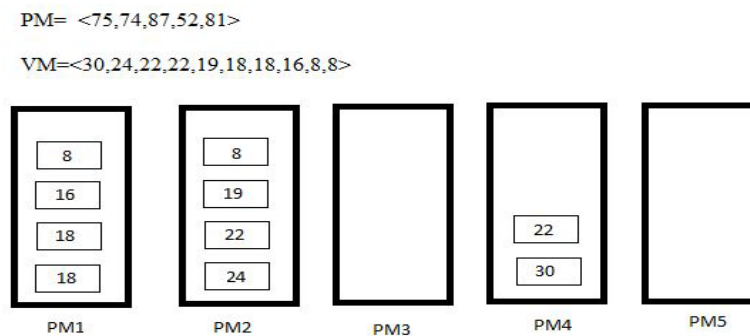
PM= <75,74,87,52,81>

VM=<30,24,22,22,19,18,18,16,8,8>

Figure 1. VM placement by Best-Fit Decreasing

Modified Best Fit Decreasing algorithm sorts the VM request in the decreasing order based on utilisation. It allocates VM to a PM, which gives a minimum rise in power consumption for that PM. On the VM request, active PMs are checked first. If a VM cannot be allocated in any of the active PMs, then the one of the PMs which is in sleep mode is activated and the VM is allocated [21]. In the Next Fit Algorithm, the last allocated PM is checked. If a new VM cannot be placed in last-placed PM, then it is placed in new PM.

In the First Fit Algorithm, all the previously allocated PMs are checked. If none PM can satisfy the VM request, then it is placed new PM [22].

In resource ratio based MCBVP, The VM with resource ratios near to residual resource ratios of PM is placed first.

3. Proposed Hybrid-Best-Fit (HBF) heuristic

The main aim is to achieve minimum energy consumption after resources are allocated. The order in which the VMs are processed is manipulated to achieve better results. The VMs are processed in increasing, decreasing and in the given order. Temporary allocation is simulated. The allocation which gives the minimum energy consumption is taken into account. Figure 2 presents the workflow of the proposed HBF heuristic and Figure 3 presents the HBF algorithm.

The algorithm works as follows: For each VM in the VMList, each PM is checked whether it is suitable to host the VM. If suitable, the residue fragment after the allocation is found out. The PM which leaves the least residue after that particular VM allocated is chosen to host in that PM. A temporary allocation list is generated. The process is iterated again until all VM requests are satisfied and energy consumption of PMs is calculated. The process is repeated for increasing and decreasing order of VM requests and each time the energy consumption of PMs are calculated. The order which results in minimum consumption is finalized

and correspondingly the VMs are allocated to the PMs.

After initial placement, all PMs are checked to see if it hosts at least one PM. If a PM does not host any VM then that particular PM is shut down. This reduces the energy consumption. In case of dynamic environments, if a new VM requests arrive which cannot be satisfied by the existing active PMs, then the PMs in rest mode is checked. If a suitable PM is found, it is activated.

4. Results and Discussion

4.1. Experimental Setup

The simulation experiments were developed in Java programming language. Java SE Development Kit (JDK) 8 is used. The experiment is run on 2.30 GHz Intel Core i36100U processor within a 64-bit Windows operating environment.

4.2. Simulation Results

PM capacities, PM energy requirements and VM requests are generated randomly. A comparison is made between BestFit, BestFit Decreasing, and Hybrid Best-Fit algorithms. The process is done for 100 times. The total energy consumption produced by the three heuristics is tabulated. Table 1 presents the results and Figure 4 highlights the energy savings by the proposed HBF heuristic compared to BF and BFD.
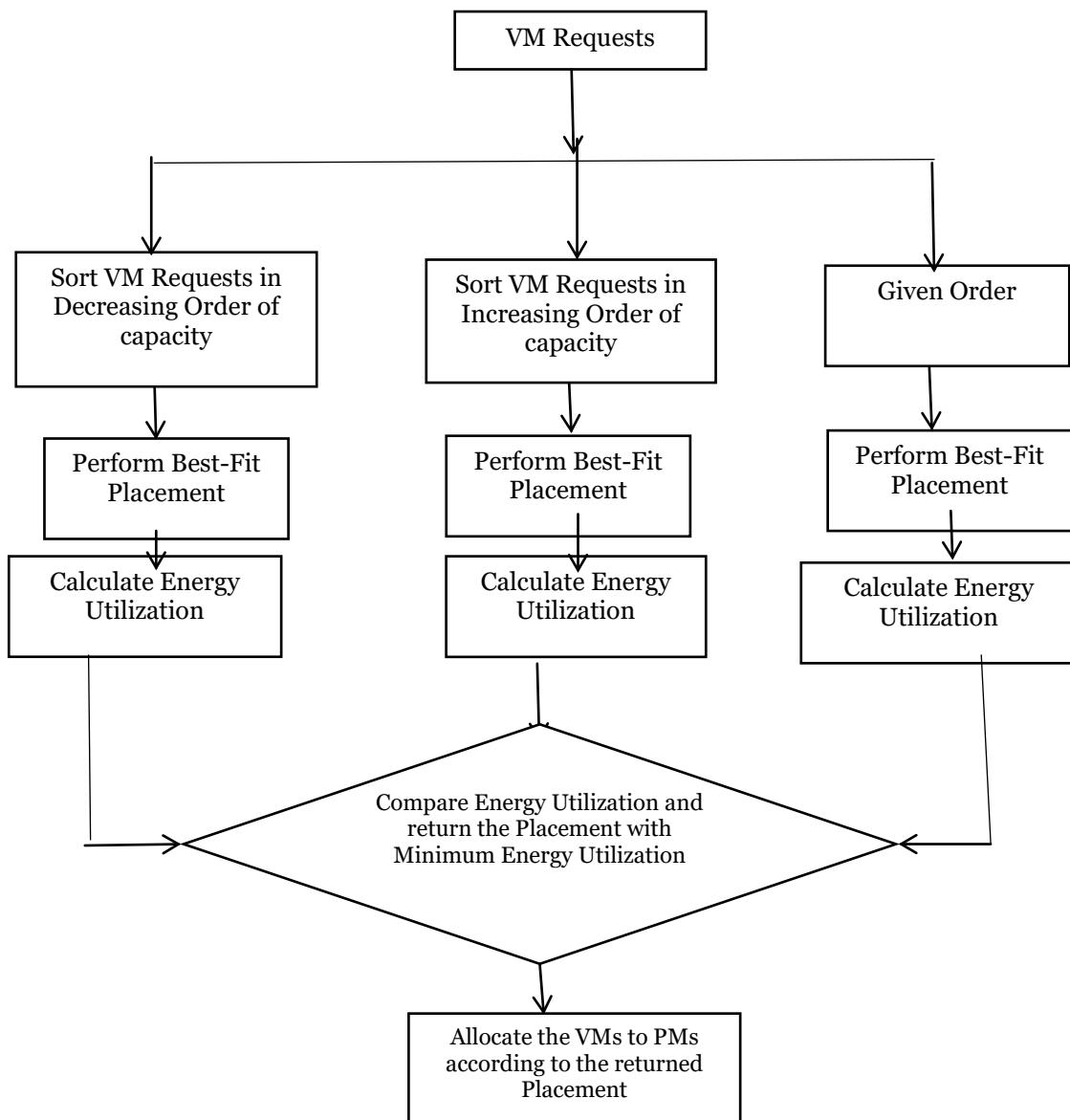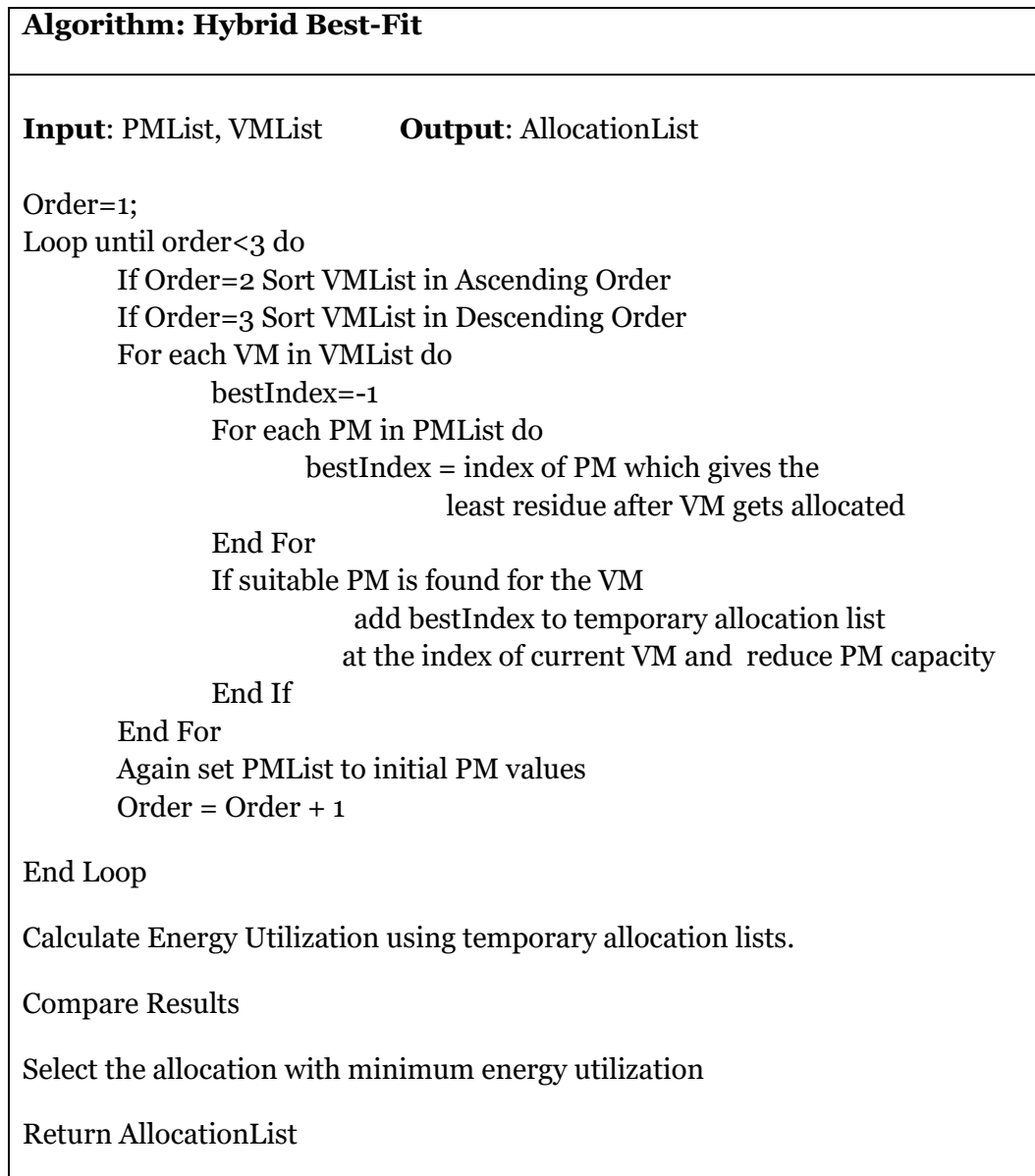
Figure 2. Proposed Hybrid Best-Fit (HBF) heuristic workflow

Algorithm: Hybrid Best-Fit

Input: PMList, VMList   **Output**: AllocationList

Order=1;
Loop until order<3 do
  If Order=2 Sort VMList in Ascending Order
  If Order=3 Sort VMList in Descending Order
  For each VM in VMList do
    bestIndex=-1
    For each PM in PMList do
      bestIndex = index of PM which gives the
        least residue after VM gets allocated
    End For
    If suitable PM is found for the VM
      add bestIndex to temporary allocation list
      at the index of current VM and reduce PM capacity
    End If
  End For
  Again set PMList to initial PM values
  Order = Order + 1

End Loop

Calculate Energy Utilization using temporary allocation lists.

Compare Results

Select the allocation with minimum energy utilization

Return AllocationList

Figure 3. Hybrid Best-Fit Algorithm

Table 1. A comparison of energy consumption by heuristics

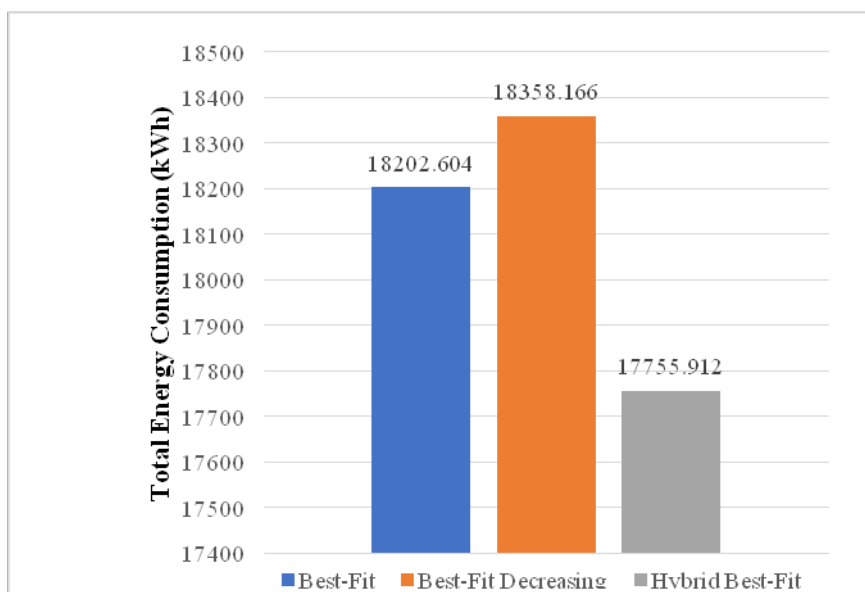| Heuristic	Best-Fit	Best-Fit Decreasing	Hybrid Best-Fit
Total energy consumption (kWh)	18202.604	18358.166	17755.912

Figure 4. Proposed HBF heuristic saves energy compare to the existing heuristics BF and BFD

5. Conclusion and Future work

We discussed energy-efficient resource allocation in cloud data centres using HBF heuristic algorithm. Here one type of resource is considered for allocation. In future, our work is to be extended to support VM placement and migration for VMs with multiple types of resources. Migration techniques allow VMs after initial placement to move from one PM to another PM. This can be done in case of underutilization or overutilization of PMs. In case of overloaded PMs, SLA violations may occur. Hence to avoid this situation, VMs should be migrated to bring the overloaded PM to normal state. In case of underloaded PMs, the VMs on the PM can be migrated to another PM so that the underloaded PM can be moved to sleep mode. Experimental results shows that the proposed HBF is consuming 2.516% and 3.392% less energy compared to Best-Fit and BFD heuristics.

References

[1] S. Jangiti, V.S.S. Sriram, R. Logesh, The role of cloud computing infrastructure elasticity in energy efficient management of datacenters, in: 2017 IEEE Int. Conf. Power, Control. Signals Instrum. Eng., IEEE, 2017: pp. 758–763. doi:10.1109/ICPCSI.2017.8391816.

[2] M. Mishra, A. Sahoo, On Theory of VM Placement : Anomalies in Existing Methodologies and Their Mitigation Using a Novel Vector Based Approach, (n.d.).

[3] Z. Usmani, S. Singh, A Survey of Virtual Machine Placement Techniques in a Cloud Data Center, Phys. Procedia. 78 (2016) 491–498. doi:10.1016/j.procs.2016.02.093.

[4] A. Fatima, N. Javaid, T. Sultana, M.Y. Aalsalem, S. Shabbir, Durr-e-Adan, An Efficient Virtual Machine Placement via Bin Packing in Cloud Data Centers, in: Springer, Cham, 2020: pp. 977–987. doi:10.1007/978-3-030-15032-7_82.

[5] F.F. Moges, S.L. Abebe, Energy-aware VM placement algorithms for the OpenStack Neat consolidation framework, J. Cloud Comput. 8 (2019) 2. doi:10.1186/s13677-019-0126-y.

[6] Z. Li, Y. Li, T. Yuan, S. Chen, S. Jiang, Chemical reaction optimization for virtual machine placement in cloud computing, Appl. Intell. 49 (2019) 220–232. doi:10.1007/s10489-018-1264-5.

[7] Energy and Service Level Agreement Aware Resource Allocation Heuristics for Cloud Data Centers, KSII Trans. Internet Inf. Syst. 12 (2018). doi:10.3837/tiis.2018.11.011.

[8] M.A. Kaaouache, S. Bouamama, An energy-efficient VM placement method for cloud data centers using a hybrid genetic algorithm, J. Syst. Inf. Technol. 20 (2018) 430–445. doi:10.1108/JSIT-10-2017-0089.

[9] G. Gupta, G. Singh Khichar, R. Rathi, R. Singh, Energy-Aware and Cost Effective VM Allocation Strategy Based on Best Fit MultiValued Binpaking Algorithm, in: 2018 8th Int. Conf. Cloud Comput. Data Sci. Eng., IEEE, 2018: pp. 647–652.

[10] doi:10.1109/CONFLUENCE.2018.8442478. G. Batra, H. Singh, I. Gupta, A.K. Singh, Best Fit Sharing and Power Aware (BFSPA) Algorithm for VM placement in cloud environment, in: 2017 3rd Int. Conf. Adv. Comput. Autom., IEEE, 2017: pp. 1–4. doi:10.1109/ICACCAF.2017.8344674.

[11] H. Wang, H. Tianfield, Energy-Aware Dynamic Virtual Machine Consolidation for Cloud Datacenters, IEEE Access. 6 (2018) 15259–15273. doi:10.1109/ACCESS.2018.2813541.

[12] M. Abdel-Basset, L. Abdle-Fatah, A.K. Sangaiah, An improved Lévy based whale optimization algorithm for bandwidth-efficient virtual machine placement in cloud computing environment, Cluster Comput. (2018) 1–16. doi:10.1007/s10586-018-1769-z.

[13] D. Alsadie, Z. Tari, E.J. Alzahrani, A. Alshammari, LIFE-MP: Online Virtual Machine Consolidation with Multiple Resource Usages in Cloud Environments, in: Springer, Cham, 2018: pp. 167–177. doi:10.1007/978-3-030-02925-8_12.

[14] F. Abdessamia, Y. Tai, W.Z. Zhang, M. Shafiq, An Improved Particle Swarm Optimization for Energy-Efficiency Virtual Machine Placement, in: 2017 Int. Conf. Cloud Comput. Res. Innov., IEEE, 2017: pp. 7–13. doi:10.1109/ICCCRI.2017.9.

[15] S. Jangiti, E.S. Ram, V.S.S. Sriram, Aggregated Rank in First-Fit-Decreasing for Green Cloud Computing, in: Cogn. Informatics Soft Comput., Springer, 2019: pp. 545–555.

[16] S. Jangiti, S.S. V S, Scalable and direct vector bin-packing heuristic based on residual resource ratios for virtual machine placement in cloud data centers, Comput. Electr. Eng. (2018). doi:10.1016/j.compeleceng.2018.03.029.

[17] S. Jangiti, E. Sriram, R. Jayaraman, H. Ramprasad, V.S.S. Sriram, Resource ratio based virtual machine placement in heterogeneous cloud data centres, Sadhana - Acad. Proc. Eng. Sci. 44 (2019). doi:10.1007/s12046-019-1215-9.

[18] S. Jangiti, S. V, S.S. V S., Bulk-bin-packing based migration management of reserved virtual machine requests for green cloud computing, EAI Endorsed Trans. Energy Web Online First. (2019). doi:10.4108/eai.11-6-2019.159605.

[19] S. Jangiti, E. Sri Ram, L. Ravi, V.S.S. Sriram, Scalable hybrid and ensemble heuristics for economic virtual resource allocation in cloud and fog cyber-physical systems, J. Intell. Fuzzy Syst. Preprint (2019) 1–11. doi:10.3233/JIFS-179004.

[20] B.S. Apinke, G.A. Adekunle, A.G. Adesola, Energy-Efficient Virtual Machine Placement Algorithm, 14 (2017) 68–75.

[21] S. Mustafa, K. Bilal, S.A. Madani, N. Tziritas, S.U. Khan, L.T. Yang, Performance Evaluation of Energy-Aware Best Fit Decreasing Algorithms for Cloud Environments, Proc. - 2015 IEEE Int. Conf. Data Sci. Data Intensive Syst. 8th IEEE Int. Conf. Cyber, Phys. Soc. Comput. 11th IEEE Int. Conf. Green Comput. Commun. 8th IEEE Inte. (2015) 464–469. doi:10.1109/DSDIS.2015.104.

[22] B. Gohil, A Comparative Analysis of Virtual Machine Placement Techniques in the Cloud Environment, 156 (2016) 12–18.