The Scientific Temper (2024) Vol. 15 (spl): 64-70
E-ISSN: 2231-6396, ISSN: 0976-8653
Doi: 10.58414/SCIENTIFICTEMPER.2024.15.spl.08
https://scientifictemper.com/

RESEARCH ARTICLE

Enhancing cloud efficiency: an intelligent virtual machine selection and migration approach for VM consolidation

O. Devipriya[1*], K. Kungumaraj[2]

Abstract

Cloud-based computing, despite its numerous benefits, frequently exerts a negative influence on the environment. The primary concern lies in the emission of greenhouse gases and the consumption of electricity by cloud data centers, which demands considerable scrutiny. Virtual machine consolidation (VM) is a widely adopted strategy aimed at achieving energy efficiency and maximizing resource utilization. The consolidation of VMs is a fundamental process in the development of a sophisticated cloud resource management system that prioritizes energy efficiency. The underlying premise is that by shifting VMs onto a reduced number of physical machines, it is possible to achieve optimization objectives, increase the utilization of cloud servers, and concurrently decrease energy consumption in cloud data centers. This proposed solution utilizes the best fit decrease (BFD) approach for VM allocation. An enhanced Greedy selection approach is proposed for VM migration, utilizing the Genetic method optimization method.

Keywords: Cloud computing, Virtual machine consolidation, Energy efficient, Optimization, Greedy selection, Genetic algorithm, VM migration.

Introduction

Cloud computing enables the provision of services through shared access, allowing users to request and utilize these services without extensive interaction with service providers. Cloud service providers provide consumers with three main types of services: infrastructure as a service (IaaS), platform as a service (PaaS), and software as a service (SaaS). These services provide help to both direct and indirect web users. The advent of cloud data centers has had a significant influence on the information technology (IT) sector. In contrast, data centers require substantial amounts of power and contribute significantly to carbon emissions in order to operate efficiently, Subashini, S., & Kavitha, V. (2011); Gorelik, E. (2013).

In the realm of cloud computing, infrastructure as a service (IaaS) has gained significant prevalence in recent years. Prominent cloud service providers, such as Rackspace and Amazon EC2, offer virtual services to consumers through internet connectivity across multiple data centers. The significant demand for cloud computing has resulted in the establishment of numerous large-scale cloud computing centers, which in turn has led to substantial power usage. The primary factor contributing to the elevated energy consumption is the suboptimal utilization of cloud resources rather than the substantial power demand associated with extensive hardware infrastructure. Hence, it is imperative, from the standpoint of energy management, to endeavor towards the establishment of adaptive techniques that enable dynamic alterations to be made to each source within cloud data centers, Lin, C.-C., Liu, P., & Wu, J.-J. (2011).

Virtualization technology, which involves the sharing of a physical machine (PM) among many digital machines (VMs), has emerged as a potential solution for enhancing energy efficiency. This technology enables improved resource utilization, hence contributing to increased efficiency. VMs have the capability to be transferred between different physical machines via a process known as live migration. This migration occurs while the virtual machine is actively running and its associated services remain accessible. This

[1]Department of Computer Science, Mother Teresa Women's University, Kodaikanal, Tamilnadu, India.

[2]Department of Computer Science, A.P.A College, Palani, Tamilnadu, India.

***Corresponding Author:** O. Devipriya, Department of Computer Science, Mother Teresa Women's University, Kodaikanal, Tamilnadu, India. , E-Mail: odpdevipriya212@gmail.com

How to cite this article: Devipriya, O., Kungumaraj, K. (2024). Enhancing cloud efficiency: an intelligent virtual machine selection and migration approach for VM consolidation. The Scientific Temper, **15**(spl):64-70.

Doi: 10.58414/SCIENTIFICTEMPER.2024.15.spl.08

Source of support: Nil

Conflict of interest: None.

Received: 12/09/2024 **Accepted:** 20/09/2024 **Published** : 16/10/2024

methodology enables the dynamic configuration of VM) to cater to diverse energy efficiency objectives or load balancing requirements based on real-time resource use, Gorelik, E. (2013), Lin, C.-C., Liu, P., & Wu, J.-J. (2011), Huang, Q., *et al*. (2011), Beloglazov, A., & Buyya, R. (2010), Zhao, Y., & Huang, W. (2009).

Virtual machine (VM) consolidation is a critical approach in cloud data centers that involves transferring VMs to a reduced number of physical machines (PMs) in order to optimize resource utilization. This technique plays a significant role in enhancing energy efficiency inside cloud data centers. In certain instances, an excessive number of virtual machines (VMs) may be consolidated within a single physical machine (PM), resulting in potential degradation of service quality. This is mostly due to the fact that VMs sharing a PM often possess comparable physical resources. Therefore, consolidation strategies pertaining to virtual machines (VMs) must prioritize the establishment of a robust and all-encompassing quality of service (QoS) that is often delineated in service level agreements (SLAs). Additionally, these strategies should aim to facilitate dynamic adjustments in the placement of VMs. Furthermore, it is imperative to consider power usage and VM migration expenses during each phase of VM consolidation, Cao, Z., & Dong, S. (2014); Gao, Y., *et al*. (2014); Yousefipour, A., Rahmani, A. M., & Jahanshahi, M. (2018); Wang, H., & Tianfield, H. (2018).

Related Works

An approach called 'PPRGear' is suggested, which utilizes use-level sampling to quantify unique PPRs. These PPRs are quantified by the number of server-side Java activities executed within a specific time frame, divided by the average active power consumption during that period. The technique is predicated upon the sampling of user proficiency levels. In addition, we introduce a framework that utilizes PPR to allocate and relocate virtual machines among various types of hosts, Ruan, X. *et al*. (2019).

The Krill Herd algorithm, a recently implemented collective intelligence system, has been proposed for the assignment of virtual machines in physical hosts within cloud data centres, Soltanshahi, M., Asemi, R., & Shafiei, N. (2019).

Gossip Contracts (GC) propose a novel multi-agent system aimed at facilitating the creation of decentralized collaborative techniques. The GC framework draws upon the Contract Net Protocols as its foundation while also finding inspiration from the Gossip protocol. In this study, we employed the Genetic Algorithm (GA) to devise a Dynamic Virtual Machine Consolidation (DVMC) strategy, which was primarily inspired by the GA approach. Subsequently, we conducted a comparative analysis between our proposed GC-based DVMC strategy and two other well-established techniques, namely Sercon and the distributed strategy of ecoCloud. One comparative experiment utilizes Google cluster-use traces, which consist of a dataset obtained from a Google data center containing real-time usage data. The GC-based technique has superior effectiveness in mitigating SLA infringements and exhibits comparable or superior performance compared to alternative energy consumption solutions, McDonnell, N., Howley, E., & Duggan, J. (2020).

The researcher devised an enhanced VM assignment model and proposed a better dynamic programming approach specifically tailored to handle a variety of user activities associated with solving the aforementioned optimization challenge. The authors have conducted a comparative analysis of the suggested method with several existing approaches, such as round-robin (RR), min-min, and differential growth, Zhang, P., Zhou, M., & Wang, X. (2020).

The proposed technique for consolidating the VM considers both current and projected resource utilization, as measured by UP-POD and UP-PUD. The utilization of resources can be reliably predicted by the application of a Gray-Markov model, Hsieh, S.-Y. *et al*. (2020).

The author proposed a strategy called Dynamic Consolidation with Minimization of Migrating Thrashing (DCMMT) that prioritizes high-capacity VMs. This strategy aims to significantly reduce migration thrashing and the overall number of migrations in order to maintain service-level agreements (SLAs). The rationale behind this approach is to prioritize migrating VMs rather than initiating migrations, Liu, X. *et al*. (2020).

A mathematical model known as a two-dimensional Markov chain, has been devised to evaluate energy conservation and response effectiveness. The efficacy of our proposed technique for VM assignment in reducing energy consumption and ensuring reaction efficiency is substantiated by numerical analysis and simulation testing, Jin, S., Qie, X., & Hao, S. (2019).

The author presented a proposed energy-efficient strategy (EES) aimed at consolidating cloud virtual machinery in order to decrease energy usage and effectively handle a greater number of high throughput operations. The use of the performance-to-power ratio is employed within our concept to establish the uppermost threshold for the detection of overload. Additionally, the energy efficiency scheduler (EES) considers the overall workload utilization of the data center in order to identify lower levels that might effectively minimize the need for virtual machine migrations, Saadi, Y., & El Kafhali, S. (2020).

This study presented a novel approach for enhancing energy efficiency in virtual machines (VMs) through the implementation of an asynchronous multi-sleep and adaptive task migration solution. The virtual-cluster virtual machines (VMs) are divided into two distinct modules, namely module I and module II. In module I, virtual machines (VMs) exhibit continuous wakefulness, while in Module II, VMs strive to operate autonomously whenever feasible. Virtual machines (VMs) within module I consistently maintain a state of wakefulness. Hence, a queuing model has been created to encapsulate the theoretical framework of the suggested technique, incorporating a partial, asynchronous multiple vacation approach. The matrix-geometric solution approach is employed to construct mathematical expressions for performance metrics, including the average response time and energy-saving rate, Qie, X., Jin, S., & Yue, W. (2019).

Genetic Algorithm Optimization

Genetic algorithms (GA) utilize biological and genetic metaphors to progressively refine a population of beginning individuals into a population of persons of superior quality. Each individual within this population represents a solution to the problem at hand and is comprised of a certain number of genes. The cardinality of a gene refers to the total number of potential values it can possess. Each individual is referred to as a chromosome. The aggregation of chromosomes constitutes a population, Song F., *et al*. (2014).

The operation of a genetic algorithm commences with the generation of a population of individuals through a random process. Over the course of multiple generations, these populations have undergone evolution, resulting in an enhancement of individuals' quality. In each successive generation, the three fundamental operators of a genetic algorithm, namely selection, crossover, and mutation, are applied to every individual. In genetic algorithms, crossover refers to the process of transferring genetic material between two chromosomes, typically including the recombination of genes. On the other hand, mutation involves the stochastic alteration of a randomly selected gene inside a chromosome, resulting in a change of its value. These individuals serve as exemplars of the problem that has to be addressed. The encoding of various positions of individuals can be achieved through the use of bits, characters, and integers, Marotta A., *et al*. (2017).

In this context, persons that best conform to the given criteria are chosen. In this context, a fitness function that the user defines is utilized. The fitness function is employed to evaluate the quality of every chromosome. The remaining individuals are coupled and, through the process of crossover, new offspring are generated by partially swapping their genetic material. The success of the genetic algorithm in problem-solving is influenced by three key parameters, as outlined in reference Marotta, A., *et al*. (2017).

- The process of choosing a fitness function
- The representation of individuals and
- The genetic parameters' values

Optimization Based Virtual Machine Consolidation

The optimization of energy usage can be achieved by dynamic consolidation of VMs, which involves the utilization of live migration techniques for VMs and the activation of sleep mode or shutdown for inactive servers. The allocation of VMs in a VM consolidation framework typically relies on the identification of the usual host through host detection techniques. This paper presents a novel approach for the selection of adaptive VM migration and the development of a heuristic algorithm for the dynamic consolidation of VMs. This approach is based on an analysis of historical data.

Input: Host List (hl) and Virtual Machine List (vml)
Output: Allocation of Virtual Machines (VMs)
Step 1: for each vm in vml do
  Step 1.1: $minPower \leftarrow Max$
  Step 1.2: $allocatedHost \leftarrow NULL$
  Step 1.3: foreach host in hl do
    Step 1.3.1: if host has enough resource for vm then
    Step 1.3.2: $power \leftarrow estimatePower(host, vm)$
    Step 1.3.3: if $power < minPower$ then
      $allocatedHost \leftarrow host$
      $minPower \leftarrow power$
  Step 1.4: if $allocatedHost \neq NULL$ then
    Step 1.4.1: allocate vm to allocateHost
Step 2: return allocation

Figure 1: Steps for VM Allocation

The suggested adaptive VM consolidation technique encompasses various components, namely coding, fitness function, selection, crossover, mutation, and greedy selection. The allocation of VMs is performed using a bin packing problem that incorporates variable bin sizes. In this context, the bin sizes correspond to the physical nodes, while the VMs represent the objects that need to be allocated. The allocation process takes into account the CPU capabilities and power consumption of the nodes. The proposed solution employs the best fit decreasing (BFD) method for VM placement. A proposed technique, known as the Greedy selection-based genetic algorithm, has been developed for the purpose of optimising VM migration. The subsequent processes outline the VM placement approach utilizing the best fit decreasing (BFD) algorithm. The list of hosts for virtual machine placement is derived from the host detection mechanism.

The present study introduced an Adaptive strategy that suggests the utilization of an improved genetic algorithm for the purpose of VM Migration. The proposed approach for Adaptive VM consolidation involves the utilization of a Greedy Search algorithm to identify the most optimized VM through the implementation of host-detecting techniques. The proposed VM consolidation technique incorporates the utilization of a Greedy Selection algorithm in the children coding subsequent to each mutation and crossover operator. The VM migration approach incorporates the utilization of the dynamic threshold mechanism. The maximum limit for VM migration is determined by utilizing the proposed method. The input for this strategy consists of the host list and the selected list of VMs.

- To establish a population, a series of codes were randomly generated.
- The utilization of binary code for the encoding of various operations.

Input: Host list (hl), Virtual Machine List (vml)
Output: VM migration list
Step 1: vml.sortDecreasingUtilization()
Step 2: for each h in hl do
    Step 2.1: $hutil \leftarrow h.util()$
        Step 2.1.1: *Initialize the population for GA.*
        Step 2.1.2: *Binary coding of the tasks assigned in the VMs are carried in this step.*
        Step 2.1.3: *The fitness value of each individual is computed.*
        Step 2.1.4: *Roulette Wheel method is used for the selection of the individual based on the probability, which is reflected by the fitness value of the population.*
        Step 2.1.5: *Two point crossover is calculated with the two individuals.*
        Step 2.1.6: *Mutation operator is applied to improve the local search capability of GA algorithm and to maintain the diversity of the population.*
        Step 2.1.7: *The greedy selection in this study occurred in the children coding after each mutation and crossover operator.*
    Step 2.2: $bestFitUtil \leftarrow MAX$
    Step 2.3: while $hutil > h.Th_{upper}()$ do
        Step 2.3.1: foreach vm in vml do
            If $vm.util() > hutil - h.Th_{upper}()$ then
                $t \leftarrow vm.util() - hutil + h.Th_{upper}()$
                If $t < bestFitUtil$ then
                    $bestFitUtil \leftarrow t$
                    $bestFitVm \leftarrow vm$
            Else
                If $bestFitUtil = MAX$ then
                    $bestFitVm \leftarrow vm$
                Break
        Step 2.3.2: $hutil \leftarrow hutil - bestFitVm.util()$
        Step 2.3.3: migration.List.add(h.getvml())
        Step 2.3.4: vmList.remove(h.getvml())
    Step 3.1: if $hutil < Th_{low}()$ then
        Step 3.1.1: migrationList.add(h.getvml())
        Step 3.1.2: vml.remove(h.getvml())
Step 4: return migrationlist

Figure 2: Steps for VM Migration and consolidation

Table 1a: Two types of host characteristics

Type	*Number of Hhost*	*Storage*	*Number of cores*	*RAM*	*Bandwidth*	*MIPS*
HP ProLiant ML 110G4	400	1GB	2	4096	1GB	1860
HP ProLiant ML 110G4	400	1GB	2	4096	1GB	2660

- The objective of this analysis is to determine the fitness value of all coding within the population.
- To generate a new population, the crossover operation is performed by selecting multiple pairs of codes. It is important to note that the overall number of individuals in the population remains unchanged during this process;
- Repeat the mutation operation using the selection operator.
- The process of selecting greedy selections for all codes inside the recently established population shall be undertaken. Facilitate the reproduction and development of a whole cohort of individuals.
- Continue iterating the process until a state of stable optimal individual fitness is achieved, specifically referring to the optimal threshold.

Result and Discussion

In the research work, the data center used in this work is considered which is also included in CloudSim. The data center has 800 hosts from two server models (400 hosts from each server type) and four types of VMs. The CPU capacity of the VM instances is given in millions of instructions per second (MIPS). The following Table 1a and b give the two types of host characteristics and the VM types used in this experiment. The Energy Consumption (kWh), SLA violations, and performance degradation due to Migration (PDM) are considered as the performance metrics.

Table 1b: Used VM types characteristics

Type of VM	*Number of VMs*	*RAM*	*MIPS*	*Storage*
VM1	1	613	500	2.5
VM2	1	1740	1000	2.5
VM3	1	1740	2000	2.5
VM4	1	2500	2500	2.5

Table 2a depicts the energy consumption (kWh) of the proposed VM consolidation strategy, cultural algorithm (CA), and artificial bee colony (ABC) optimization at a number of hosts = 400 and the number of VMs = 100. From Table 2a, it is clear that the proposed VM consolidation strategy consumes less energy when increasing the number of tasks.

Table 2b gives the SLA violations (in %) of the Proposed VM Consolidation Strategy, CA and ABC at number of hosts = 400 and number of VMs = 100. From Table 2b, it is clear that the proposed VM consolidation strategy consumes

Table 2a: Energy consumption (kWh) of the proposed VM consolidation strategy, CA and ABC at number of hosts = 400 and number of VMs = 100

Number of tasks	Energy consumption (kWh)		
	Proposed VM consolidation strategy	CA	ABC
100	3.94	9.61	11.02
200	4.48	10.32	12.5
300	4.96	11.12	13.7
400	5.51	11.94	14.3
500	5.95	12.32	15.4
600	6.38	13.87	16.3
700	6.99	14.31	17.5
800	7.28	15.52	18.6
900	7.84	16.55	19.1

Table 2b: Service level agreement (SLA) violations (in %) of the proposed VM consolidation strategy, CA and ABC at number of hosts = 400 and number of VMs = 100

Number of tasks	SLA Violation (%)		
	Proposed VM consolidation strategy	CA	ABC
100	0.002	0.234	0.865
200	0.165	0.496	1.103
300	0.316	0.812	2.232
400	0.498	1.624	3.275
500	0.501	2.238	4.357
600	0.563	2.781	5.531
700	0.599	3.794	6.642
800	0.621	4.236	7.653
900	0.659	5.626	8.4751

less energy as well as it reduces the SLA violations than the existing optimization techniques.

Table 3a depicts the energy consumption (in kWh) of the proposed VM consolidation strategy, CA and ABC at number of hosts = 400 and the number of VMs = 200. Table 3b gives the SLA violations (in %) of the Proposed VM Consolidation Strategy, CA and ABC at number of hosts=400 and number of VMs = 200. From Table 3a and b, it is clear that the proposed VM consolidation strategy consumes less energy when increasing the number of tasks, and it reduces the SLA violations also.

Table 4a depicts the energy consumption (in kWh) of the proposed VM consolidation strategy, CA and ABC at a number of hosts = 800 and number of VMs = 100. Table 4b gives the SLA violations (in %) of the proposed VM consolidation strategy, CA and ABC at number of hosts = 800 and number of VMs = 100. From Tables 4a and b, it is clear

Table 3a: Energy consumption (kWh) of the proposed VM consolidation strategy, CA and ABC at number of hosts = 400 and number of VMs = 200

Number of Tasks	Energy consumption (kWh)		
	Proposed VM consolidation strategy	CA	ABC
100	4.54	8.85	13.4
200	4.98	9.51	14.9
300	5.46	10.23	15.5
400	5.93	11.94	16.4
500	6.37	12.28	17.1
600	6.88	13.97	18.3
700	7.43	14.36	19.7
800	7.96	15.85	20.2
900	8.54	16.32	20.9

Table 3b: Service level agreement (SLA) violations (in %) of the proposed VM consolidation strategy, CA and ABC at number of hosts = 400 and number of VMs = 200

Number of tasks	SLA Violation (%)		
	Proposed VM consolidation strategy	CA	ABC
100	0.045	0.561	0.974
200	0.192	1.265	1.814
300	0.424	1.818	2.411
400	0.506	2.314	3.583
500	0.613	3.179	4.549
600	0.741	4.671	5.743
700	0.871	5.963	6.831
800	0.913	6.524	7.845
900	1.148	7.815	8.687

Table 4a: Energy consumption (kWh) of the proposed VM consolidation strategy, CA and ABC at number of hosts = 800 and number of VMs = 100

Number of tasks	Energy consumption (kWh)		
	Proposed VM consolidation strategy	CA	ABC
100	6.36	15.22	16.8
200	7.16	16.61	17.2
300	8.24	17.13	18.6
400	9.71	18.32	19.9
500	10.19	19.42	20.4
600	11.60	20.83	21.3
700	12.41	21.62	22.5
800	13.84	22.54	23.7
900	14.32	23.19	24.8

Table 4b: Service level agreement (SLA) violations (in %) of the proposed VM consolidation strategy, CA and ABC at number of hosts = 800 and number of VMs = 100

Number of tasks	*SLA violation (%)*		
	Proposed VM consolidation strategy	*CA*	*ABC*
100	0.452	1.452	1.863
200	0.631	2.753	2.925
300	0.827	3.513	3.532
400	1.078	4.424	4.772
500	1.392	5.568	5.738
600	1.796	6.852	6.931
700	2.143	7.619	7.622
800	2.635	8.735	8.934
900	2.968	9.753	9.876

Table 5a: Energy consumption (kWh) of the proposed VM consolidation strategy, CA and ABC at number of hosts = 800 and number of VMs = 200

Number of tasks	*Energy consumption (kWh)*		
	Proposed VM consolidation strategy	*CA*	*ABC*
100	7.44	17.2	19.6
200	8.24	18.8	20.3
300	9.12	19.43	21.7
400	10.52	20.19	22.6
500	11.27	21.33	23.5
600	12.42	22.53	24.4
700	13.22	23.34	25.9
800	14.62	24.82	26.7
900	15.14	25.55	27.9

Table 5b: Service level agreement (SLA) violations (in %) of the proposed VM consolidation strategy, CA and ABC at number of hosts = 800 and number of VMs = 200

Number of tasks	*SLA violation (%)*		
	Proposed VM consolidation strategy	*CA*	*ABC*
100	0.671	2.241	2.641
200	0.813	3.651	3.713
300	1.245	4.723	4.313
400	1.866	5.628	5.551
500	2.404	6.782	6.916
600	2.914	7.623	7.852
700	3.331	8.582	8.813
800	3.813	9.524	9.726
900	4.146	10.215	10.964

that the proposed VM consolidation strategy consumes less energy when increasing the number of tasks, and it reduces the SLA violations also.

Table 5a depicts the energy consumption (in kWh) of the proposed VM consolidation strategy, CA and ABC at a number of hosts = 800 and number of VMs = 200. Table 5b gives the SLA violations (in %) of the Proposed VM consolidation strategy, CA and ABC at number of hosts = 800 and number of VMs = 200. From Tables 5a and b, it is clear that the proposed VM consolidation strategy consumes less energy when increasing the number of tasks, and it reduces the SLA violations also.

Conclusion

In cloud computing, users may utilize hundreds of thousands of virtualized resources and it is impossible for everyone to allocate each task manually. Due to commercialization and virtualization, cloud computing left the task scheduling complexity to the virtual machine layer by utilizing resources virtually. Hence to assign the resources to each task efficiently and effectively, scheduling plays an important role in cloud computing. In the cloud platform, dynamic change of VM availabilities makes satisfying the task QoS requirement difficult. To improve task scheduling capacities of VMs in the cloud platform and satisfy task QoS requirement, a task scheduling algorithm based on VM availability awareness was proposed in this study to solve the matching problem between available task processing capacities of VMs and task QoS requirement and realized workload balancing of servers in the cloud platform.

References

Beloglazov, A., & Buyya, R. (2010). Energy efficient allocation of virtual machines in cloud data centers. In 2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing (pp. 557-562). *IEEE*. https://doi.org/10.1109/CCGRID.2010.87

Cao, Z., & Dong, S. (2014). An energy-aware heuristic framework for virtual machine consolidation in cloud computing. *The Journal of Supercomputing*, 69(1), 429-451. https://doi.org/10.1007/s11227-014-1124-y

Gao, Y., et al. (2014). Service level agreement based energy-efficient resource management in cloud data centers. *Computers & Electrical Engineering*, 40(5), 1621-1633. https://doi.org/10.1016/j.compeleceng.2014.05.003

Gorelik, E. (2013). Cloud computing models (Doctoral dissertation, Massachusetts Institute of Technology).

Hsieh, S.-Y., et al. (2020). Utilization-prediction-aware virtual machine consolidation approach for energy-efficient cloud data centers. *Journal of Parallel and Distributed Computing*, 139, 99-109. https://doi.org/10.1016/j.jpdc.2020.07.005

Huang, Q., et al. (2011). Power consumption of virtual machine live migration in clouds. In 2011 Third International Conference on Communications and Mobile Computing (pp. 121-125). *IEEE*. https://doi.org/10.1109/CMC.2011.49

Jin, S., Qie, X., & Hao, S. (2019). Virtual machine allocation strategy in energy-efficient cloud data centres. *International Journal* *of Communication Networks and Distributed Systems*, 22(2), 181-195. https://doi.org/10.1504/IJCND.2019.098765

Lin, C.-C., Liu, P., & Wu, J.-J. (2011). Energy-aware virtual machine dynamic provision and scheduling for cloud computing. *In 2011 IEEE 4th International Conference on Cloud Computing (pp. 1-8). IEEE*. https://doi.org/10.1109/CLOUD.2011.33

Liu, X., et al. (2020). Virtual machine consolidation with minimization of migration thrashing for cloud data centers. *Mathematical Problems in Engineering*, 2020. https://doi.org/10.1155/2020/8874984

Marotta, A., et al. (2017). A fast robust optimization-based heuristic for the deployment of green virtual network functions. *Journal of Network and Computer Applications*, 95, 42-53. https://doi.org/10.1016/j.jnca.2017.07.015

McDonnell, N., Howley, E., & Duggan, J. (2020). Dynamic virtual machine consolidation using a multi-agent system to optimise energy efficiency in cloud computing. *Future Generation Computer Systems*, 108, 288-301. https://doi.org/10.1016/j.future.2020.02.016

Qie, X., Jin, S., & Yue, W. (2019). An energy-efficient strategy for virtual machine allocation over cloud data centers. *Journal of Network and Systems Management*, 27(4), 860-882. https://doi.org/10.1007/s10922-019-09519-1

Ruan, X., et al. (2019). Virtual machine allocation and migration based on performance-to-power ratio in energy-efficient clouds. *Future Generation Computer Systems*, 100, 380-394. https://doi.org/10.1016/j.future.2019.05.036

Saadi, Y., & El Kafhali, S. (2020). Energy-efficient strategy for virtual machine consolidation in cloud environment. *Soft Computing*. https://doi.org/10.1007/s00500-020-04553-0

Soltanshahi, M., Asemi, R., & Shafiei, N. (2019). Energy-aware virtual machines allocation by krill herd algorithm in cloud data centers. *Heliyon*, 5(7), e02066. https://doi.org/10.1016/j.heliyon.2019.e02066

Song, F., et al. (2014). An optimization-based scheme for efficient virtual machine placement. *International Journal of Parallel Programming*, 42(5), 853-872. https://doi.org/10.1007/s10766-014-0297-5

Subashini, S., & Kavitha, V. (2011). A survey on security issues in service delivery models of cloud computing. *Journal of Network and Computer Applications*, 34(1), 1-11. https://doi.org/10.1016/j.jnca.2010.07.001

Wang, H., & Tianfield, H. (2018). Energy-aware dynamic virtual machine consolidation for cloud datacenters. *IEEE Access*, 6, 15259-15273. https://doi.org/10.1109/ACCESS.2018.2793600

Yousefipour, A., Rahmani, A. M., & Jahanshahi, M. (2018). Energy and cost-aware virtual machine consolidation in cloud computing. *Software: Practice and Experience*, 48(10), 1758-1774. https://doi.org/10.1002/spe.2647

Zhang, P., Zhou, M., & Wang, X. (2020). An intelligent optimization method for optimal virtual machine allocation in cloud data centers. *IEEE Transactions on Automation Science and Engineering*. https://doi.org/10.1109/TASE.2020.2993276

Zhao, Y., & Huang, W. (2009). Adaptive distributed load balancing algorithm based on live migration of virtual machines in cloud. *In 2009 Fifth International Joint Conference on INC, IMS and IDC* (pp. 226-231). *IEEE*. https://doi.org/10.1109/IMCOM.2009.104