Nature-Inspired Optimization for Virtual Machine Allocation in Cloud Computing: Current Methods and Future Directions

Xiaoqing YANG*
Henan Technical, College of Construction, Zhengzhou 450000, China

***Abstract*—An expanding range of services is offered by cloud data centers. The execution of application tasks is facilitated by assigning (VMs) Virtual Machines to (PMs) Physical Machines. Speaking of VM allocation in the cloud service center, two key factors are taken into consideration: quality of service (QoS) and energy consumption. The cloud service center aims to optimize these aspects while allocating VMs. On the other hand, cloud users have their priorities and focus on their specific requirements, particularly throughput and reliability. User requirements are considered by the cloud service center, resulting in VM allocation that meets QoS targets and optimizes energy consumption. Cloud service centers must, therefore, find a balance between QoS and energy efficiency while considering the user's requirements. To achieve this, various optimization algorithms and techniques must be employed. The objective is to find the best allocation of VMs to PMs. Due to the NP-hardness of the VM allocation problem, nature-inspired meta-heuristic algorithms have become commonly used to solve it. However, there are no comprehensive and in-depth review papers on this specific area. This paper aims to bridge a knowledge gap by providing an understanding of the significance of metaheuristic methods to address the VM allocation issue effectively. It not only highlights the role played by these algorithms but also examines the existing methods, provides comprehensive comparisons of strategies based on key parameters, and concludes with valuable recommendations for future research.**

***Keywords*—Cloud computing; virtualization; virtual machine allocation; optimization**

I. INTRODUCTION

Cloud computing, characterized by on-demand services utilizing virtualized computing resources and streamlined software and hardware maintenance [1], has significantly shifted organizational paradigms from private infrastructure to cloud-based platforms [2]. However, the rapid growth of cloud data centers has brought about challenges, notably in energy consumption and environmental impact due to the extensive deployment of computing resources [3]. Service provisioning in cloud computing revolves around Service Level Agreements (SLAs), offering a spectrum of services encompassing hardware/software rental, resource management, and workload distribution [4]. The versatility of cloud services, encompassing Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS), and the comprehensive concept of Everything as a Service (XaaS), optimizes IT infrastructure for enhanced service delivery [5]. The pervasive adoption of cloud computing has amplified concerns regarding the substantial energy consumption inherent in data centers, accentuated by underutilized servers leading to inefficient energy utilization [6]. Addressing this challenge necessitates optimizing server utilization in data centers while ensuring seamless service delivery [7]. However, the increasing diversity of resource types in data center architectures poses a challenge in improving resource efficiency, especially in dispersed and heterogeneous environments owned by major cloud providers. The fundamental problem centers on efficiently allocating resources based on cloud user requests while adhering to SLAs.

The convergence of Internet of Things (IoT), fuzzy logic, Machine Learning (ML), Deep Learning (DL), Neural Networks (NNs), and meta-heuristic algorithms shapes efficient cloud resource allocation. The proliferation of IoT has generated vast data streams, requiring sophisticated allocation mechanisms [8, 9]. Fuzzy logic, integrating imprecise or uncertain data, enhances decision-making in allocating resources, considering ambiguous parameters [10, 11]. ML comprising supervised and unsupervised learning paradigms, aids in predicting resource demands and pattern recognition for optimized allocations [12-14]. DL, a subset of ML, with its complex neural networks, enables automatic feature extraction, fostering accurate resource predictions, and allocation decision-making [15, 16]. NNs, mimicking human brain functions, offer robust solutions for dynamic resource allocation challenges by learning from patterns and behaviors in cloud environments [17].

By employing clustering techniques, cloud systems can categorize and group entities with similar attributes or behaviors, allowing for more efficient resource allocation strategies. This approach enables the identification of patterns and similarities among diverse entities, such as VMs or user requests, facilitating the allocation of resources based on common characteristics [18]. Moreover, meta-heuristic algorithms, drawing inspiration from natural phenomena, provide efficient search strategies in complex solution spaces, optimizing cloud resource allocation by addressing scalability, dynamicity, and diverse user requirements [19]. This convergence is pivotal in enhancing the adaptability, accuracy, and efficiency of cloud resource allocation, catering to the burgeoning demands of modern cloud infrastructures while enabling dynamic, scalable, and optimized resource

provisioning, underpinning the evolution and sustainability of cloud computing ecosystems [20].

The primary questions addressed in this study revolve around the optimization of energy consumption and resource allocation complexities within cloud computing. Firstly, how can server utilization be improved in data centers to mitigate energy wastage? Secondly, what effective strategies can manage the diverse resource types and demand patterns inherent in cloud systems? Finally, how can resources be allocated efficiently while meeting the diverse SLA requirements of cloud users? These key inquiries guide the exploration of efficient resource allocation methodologies and user-centric approaches within the realm of cloud computing. This review paper aims to address the gap in existing literature by comprehensively exploring solutions for efficient resource allocation in cloud computing. It assesses methodologies to optimize energy consumption while meeting SLAs, investigates resource allocation complexities, and suggests user-centric strategies. By offering comparative insights and recommendations, this study aims to provide a robust understanding of cloud resource allocation for future research directions.

II. BACKGROUND

Various methods have been developed to optimize the allocation of VMs on physical machines in cloud data centers. Fig. 1 presents an overview of virtual machine allocation strategies. Using multidimensional resources in an unbalanced manner can enhance resource utilization and lead to abnormalities. A balanced use of multidimensional resources refers to the use of resources in each dimension proportional to their total amount. Virtual machines may run out of resources if placed regardless of this characteristic. Placement decisions can take into account various costs, including virtual machines, physical machines, cooling, data centers, and traffic, as shown in Fig. 2. The cost of cloud services is influenced by multiple factors. One aspect focuses on reducing costs for cloud users, specifically, the expense associated with virtual machines. Simultaneously, other factors aim to decrease costs for cloud service providers. Virtual machine allocation involves assigning virtual machines to suitable physical machines. To address the complexity of the NP-hard problems involved in this allocation, meta-heuristic algorithms are employed. Virtualization technology is critical for cloud computing, and the task of assigning virtual machines to physical machines is referred to as the virtual machine allocation problem, a known NP-hard challenge.

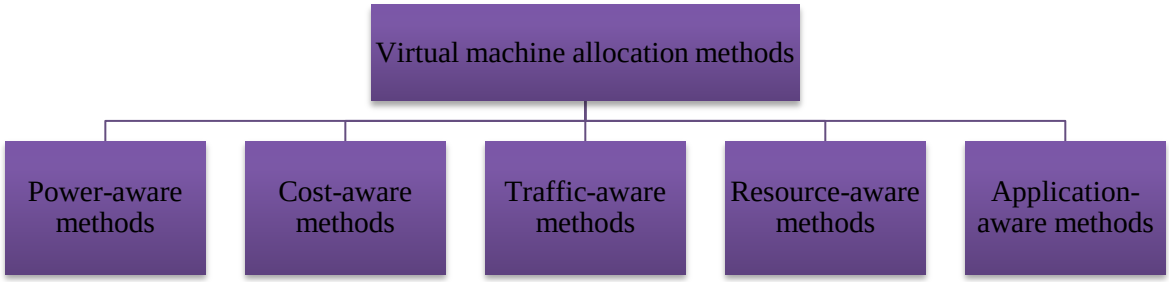

Fig. 1. Taxonomy of virtual machine allocation methods

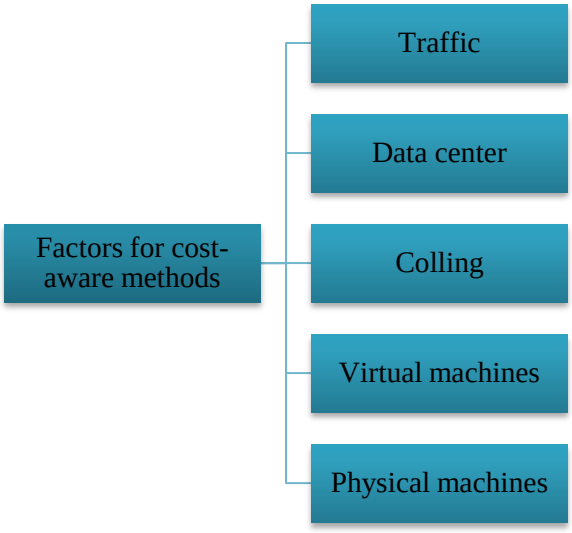

Fig. 2. Considered factors for cost-aware methods.

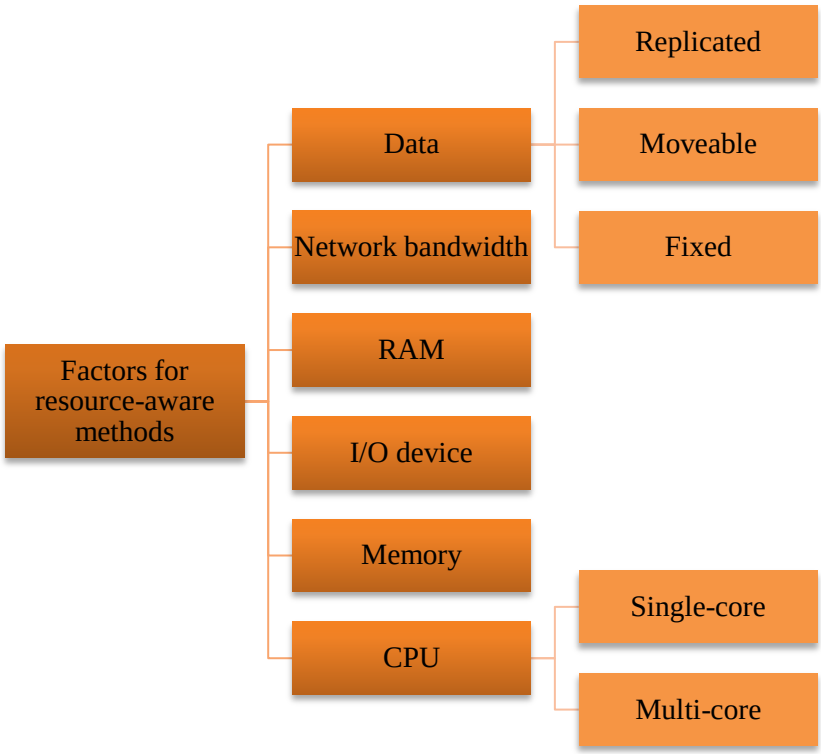

Fig. 3. Factors for resource-aware methods.

By utilizing virtualization technology extensively, data centers can optimize their resource management. The use of virtualization technology in cloud data centers optimizes the use of resources and minimizes operating costs. Cloud computing uses demand-driven resources such as VMs, to simplify the processing of complex duties. In addition, some VM allocation strategies attempt to place virtual machines at a minimum cost by considering various cost parameters. In this regard, as illustrated in Fig. 3, virtual machines or virtual data centers may be assigned different resources. Virtual machines are primarily equipped with a virtual CPU, which may either be single-core or multi-core. Virtual machines have access to crucial virtual resources, including network bandwidth and data. To optimize resource allocation for virtual machine applications, it is beneficial to place interacting virtual machines in close proximity. Additionally, the management of data, whether moveable, fixed, or replicated, depends on factors such as size and security policy.

Fig. 4 depicts the importance of considering traffic-related parameters, specifically the significant role of bandwidth, in improving the performance and efficiency of virtual machine allocation. Most approaches consider traffic between physical machines, traffic between interacting virtual machines, and traffic between virtual machines and their data repositories. Furthermore, as shown in Fig. 5, in the case of multiple-site clouds or geographically distributed clouds, a variety of factors will affect the location of the data center. Virtual machine allocation techniques, as shown in Fig. 5, depend on several factors, such as physical machine power consumption, number of physical machines, switches, and switch ports. Data centers are constructed based on specific topologies, taking into account these considerations.

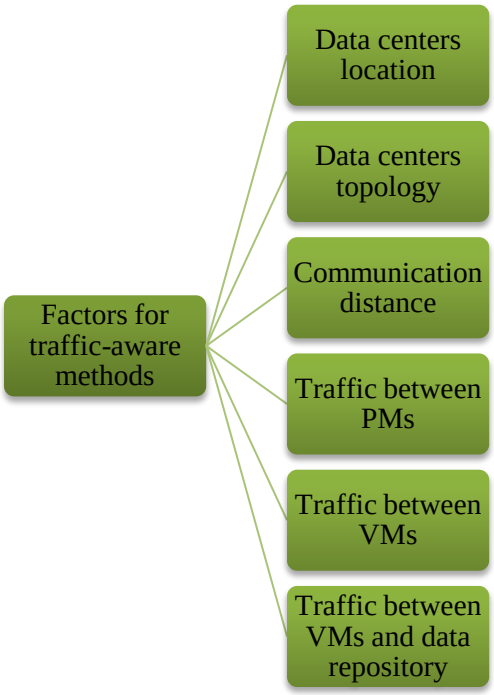

Fig. 4. Factors for traffic-aware methods.

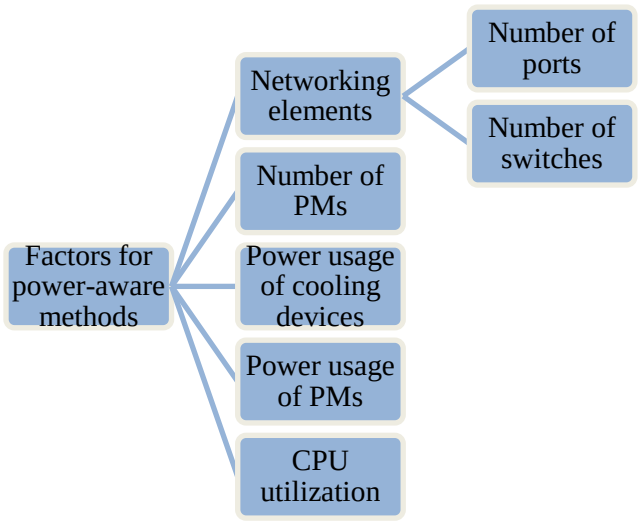

Fig. 5. Factors for power-aware methods.

Fig. 6 categorizes virtual machine allocation methods based on the availability of sites and clouds for virtual machine deployment. Both the single-cloud and multi-cloud models have the capability to utilize one or multiple sites to provide services. In the single-cloud model, services are offered from a single cloud provider's infrastructure, which may be composed of multiple sites or data centers. On the other hand, the multi-cloud model involves multiple cloud providers, each with their own sites or data centers, to deliver services. Various factors, including resources, location, performance, and cost, determine the selection of one or more sites. The aim is to ensure efficient service delivery, high availability, and scalability based on the specific needs of the applications or workloads. On the other hand, Fig. 7 classifies virtual machine allocation strategies in terms of load-aware parameters. These parameters are utilized by virtual machine allocation techniques to forecast and optimize virtual machine utilization.

Fig. 8 demonstrates the categorization of data-aware factors within a cloud data center, highlighting the significance of data characteristics in data-aware virtual machine allocation methods. Virtual machine allocation strategies are classified as dynamic and static. Virtual machine lifetime is the primary factor in static virtual machine allocation. The suitability of a physical machine to host a virtual machine is determined by its performance and longevity. During dynamic virtual machine allocation, the initial virtual machine assignment on PMs within a cloud data center can be modified based on certain factors, such as system load. Dynamic virtual machine assignment strategies can be categorized as reactive or proactive. During reactive virtual machine allocation, the initial placement changes when a particular undesirable state is reached. Modification of the initial placement of virtual machines is often necessitated by various factors such as SLA violations, load balancing, power consumption, and performance. Proactive virtual machine allocation methods proactively adjust the initial placement in advance of reaching a particular state or meeting specific demands. Virtual machine allocation techniques typically fall into one of two categories: allocation within a single cloud or allocation within a federated cloud.

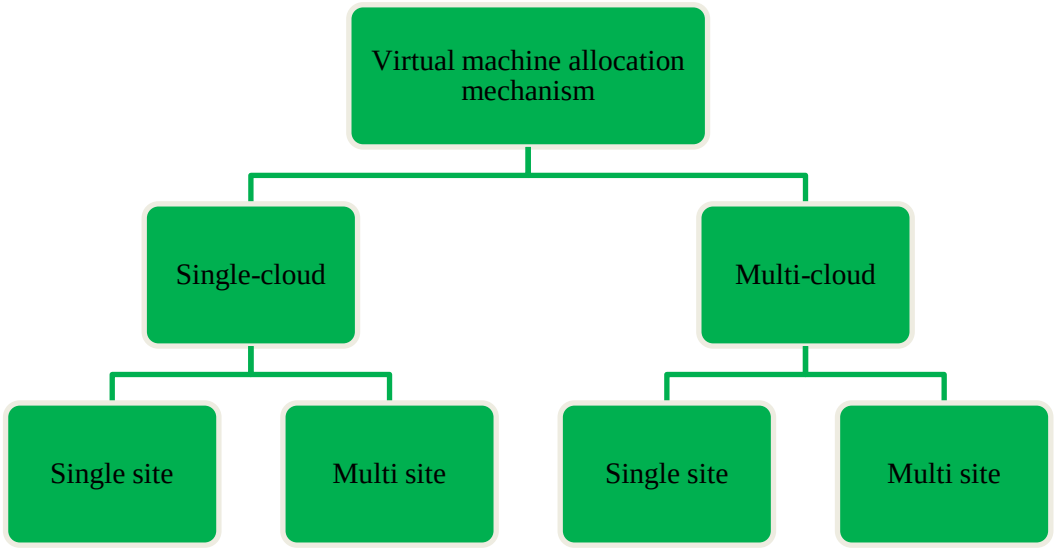

Fig. 6. Taxonomy of methods based on the number of clouds.

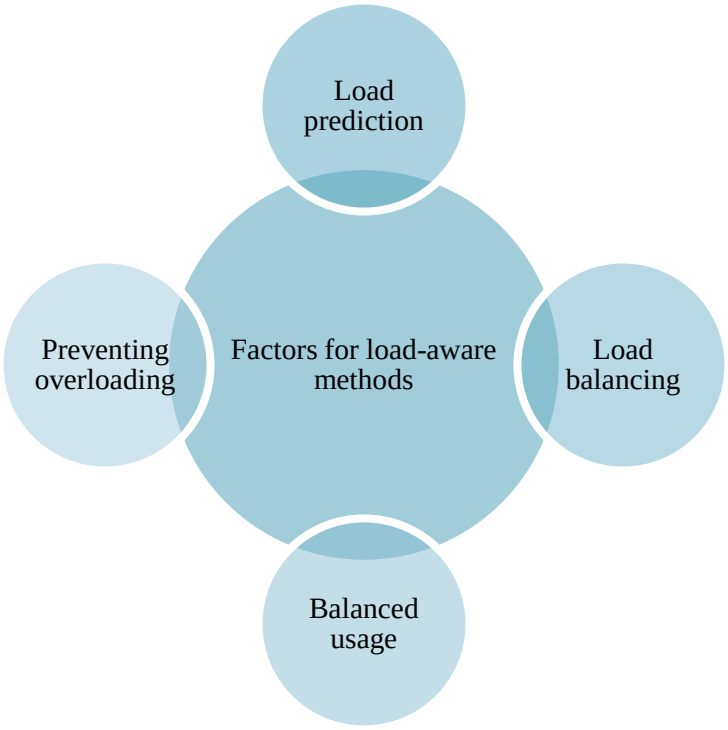

Fig. 7. Factors for load-aware methods.

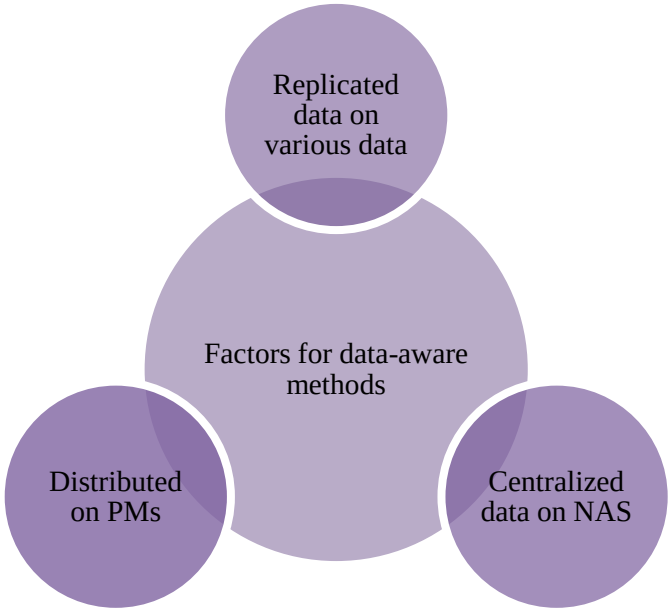

Fig. 8. Factors for data-aware methods.

Traffic in the cloud data center can be categorized into the inter-data center and intra-data center. These factors can be improved by optimizing the placement of virtual machines on physical machines. Additionally, achieving optimal allocation of virtual machines brings about significant enhancements such as increased availability, improved accessibility, higher performance, enhanced load balancing, reduced SLA violations, lowered costs in cloud data centers, decreased power consumption, and optimized resource utilization.

III. Review of the Selected Meta-Heuristic Algorithms

This section offers a detailed assessment of the selected virtual machine allocation methods, emphasizing their essential features, benefits, and drawbacks. We have conducted an analysis of nature-inspired meta-heuristic algorithms across various categories, including SA, PSO, HS, GA, FA, CSA, ABC, BBO, ACO, CSO, GRASP, SCA, PSOA, SFLA, and

TS. We examine the main characteristics and performance of these algorithms, allowing for a comprehensive understanding of their applicability to virtual machine allocation.

A. Ant Colony Optimization Algorithm

The ACO algorithm imitates how ants search for food. It has gained significant importance in addressing the virtual machine allocation problem in cloud computing. ACO employs a population of artificial ants that iteratively build solutions by depositing pheromone trails on a graph representation. These trails guide the ants to explore and exploit potential areas of the solution space. In the context of virtual machine allocation, ACO can effectively optimize resource allocation and load balancing and minimize energy consumption. By using the pheromone trails to represent the desirability of allocating virtual machines to specific hosts, ACO can intelligently guide the allocation process [21]. Due to its ability to find near-optimal solutions, adapt to dynamic environments, and deal with large-scale optimization problems, ACO provides a valuable tool for addressing VM allocation challenges in cloud computing.

Esnault, et al. [22] suggested a decentralized dynamic VM allocation method based on a Peer-to-Peer (P2P) network of PMs. The system utilizes a dynamic structure where neighborhood data is randomly exchanged among the PMs, eliminating the need for global system knowledge. This approach enables regular VM allocation within the local area, ensuring scalability even as the number of PMs increases. The neighborhood-randomized characteristics of this approach contribute to the convergence of the allocation, resulting in a performance that closely resembles a centralized one. Experimental results on the Grid'5000 testbeds demonstrate that this approach achieves global packing efficiency comparable to a centralized system, leading to increased scalability, resource utilization, and overall performance. This technique has the potential to disregard high SLA violations and create more PMs and migration overhead, which is important to keep in mind.

Feller, et al. [23] offered a novel dynamic VM allocation method to enhance resource utilization by reducing the number of running PMs according to the available resources. This is a distributed mechanism to solve two sub-problems: VM allocation optimization and PM status monitoring. The ACO algorithm has been applied to discover the best VM allocation solutions. Finally, the authors have shown that this technique on the real workload effects has improved performance and high scalability, reduced energy consumption and resource utilization, and fewer SLA violations. However, this method ignores high migration time and high migration overhead. Farahnakian, et al. [24] have suggested compatibility and integration of the ACO algorithm with optimal and balanced computing resources integrated with vector algebra. This technique minimizes resource wastage and energy consumption in virtual data centers by using the ACO-based server consolidation mechanism. Furthermore, this technique has low energy consumption, low time complexity, and high resource utilization. But this technique suffers from a high overhead VMs placement and low QoS.

Ferdaus, et al. [25] have proposed distributed system architecture to achieve a dynamic VM allocation method. The proposed method finds an optimal solution for a specified objective function. Energy consumption in data centers has decreased by assigning VMs to fewer active PMs while preserving QoS requirements. The technique incorporates inactive PMs and immigrants into a multi-objective model. Furthermore, the comparison has shown that this method reduced power consumption, increased performance, fewer SLA violations, and high resource utilization. However, this method ignores low scalability and high overhead migration.

Farahnakian, et al. [26] proposed a dynamic load balancing method based on the average load of the cloud data center. This technique has an algorithm for dynamic load balancing and VM allocation to decrease cloud data centers' energy utilization while preserving the desired QoS. In this method, the ACO-based VM allocation method is based on action artificial ants to allocate VMs to decrease the number of active PMs while satisfying the current resource requirements and also uses a dynamic threshold calculation approach to calculate the load of all active PMs and change the higher threshold value according to necessity. The comparison analysis has revealed that the method employing a static threshold scheme has effectively reduced the overall migrations of VM and power usage. The ant colony optimization approach gives a better result in combining dynamic threshold-based VM consolidation. Furthermore, low energy consumption, high QoS, and low SLA violation are some of the advantages of this method.

Matre, et al. [27] have offered a novel migration overhead-aware VM allocation mechanism based on the ACO algorithm for reducing asset usage for large info centers, computational overhead, and power usage. Data center elements and applications are also damaged by dynamic VM consolidation over VM-aware migrations. Therefore, a method for estimating migration overhead is suggested, incorporating pragmatic migration factors. Also, in this technique, to improve the system's scalability for large-scale data centers, a hierarchical, decentralized integration framework has been developed using localization of VM integration functions and minimizing their network impact. Nevertheless, this technique ignores QoS and migration time.

Ashraf and Porres [28] have presented a new workload-aware VM allocation method to reach efficient VM allocation on the basis of the MMAS (Max-Min Ant System) by applying multidimensional time-varying workloads. This method has used the perfect caseload models and multidimensional assets being heuristic variables, and the SLA model has been exploited to guarantee customer satisfaction. Also, this method examines both relations between workloads and the influence of resource-imbalanced usage, which decreases the server counts and resource waste. The proposed method offers several advantages, including resource utilization and high performance, ensuring efficient utilization of available resources in the cloud environment. Additionally, it aims to provide high QoS by optimizing VM allocation and minimizing resource wastage. Another significant advantage is the reduction in energy consumption, contributing to improved

sustainability and cost-effectiveness of cloud computing systems.

Malekloo, et al. [29] presented an energy-aware VM allocation method employing the ACO algorithm to minimize energy consumption in data centers. This method takes into account the energy consumption associated with VM migration as a key factor. As a result, it greatly decreases dynamic PMs and relocations number, leading to savings in energy utilization while ensuring QoS requirement are met through SLAs. The technique effectively decreases the migration rate and minimizes total energy waste in the network. It offers high adaptability, QoS, resource utilization, and low energy consumption. However, this method may face challenges in terms of scalability and overhead migration, which could impact its applicability in larger and more complex environments.

Aryania, et al. [30] proposed a multi-objective approach for VM allocation utilizing double thresholds and the Ant Colony System (ACS). This method focuses on VM consolidation and employs double thresholds to determine the conditions for consolidation. The authors use a mapping relation between VMs and hosts, treating the hosts as food sources, and optimize this mapping relationship through a multi-stage consolidation process based on ACS. By utilizing distributed search and the cooperation of artificial ants, the approach aims to achieve an optimal global mapping relationship between virtual machines and hosts. The performance of the proposed method is assessed by utilizing actual workload data and employing dual thresholds for CPU utilization. These thresholds help determine whether a host is experiencing an overload or underload condition. When a host falls into either of these categories, the process of VM consolidation is initiated. This consolidation process involves implementing specific policies based on the load status of the host, which in turn enables the migration of selected VMs to destination hosts using ACS. This method has some drawbacks, such as high migration time and limited scalability, which may affect its suitability for larger and more complex environments.

Xiao, et al. [31] proposed a new method called ELM_MPACS, which combines the multi-population ACS algorithm with Extreme Learning Machine (ELM) prediction. This method offers a lower complexity compared to other approaches. The ELM_MPACS algorithm operates in two steps. Firstly, it estimates the state of hosts using ELM, which helps determine whether a host is over-loaded or under-loaded. In the second step, if a host is found to be over-loaded, the virtual machine on that host is migrated to a normal host.

On the other hand, if a host is under-loaded, the virtual machine is consolidated onto another under-loaded host with higher utilization. The concentration of pheromones in the algorithm increases the likelihood of a combination being selected in the future. ELM_MPACS uses multiple populations to choose the ideal solution each time, increasing diversity and ensuring convergence. Experimental results have shown several positive outcomes, including improved resource utilization, reduced SLA violations, decreased energy consumption, and shorter migration times. This method provides promising results in terms of resource management, SLA compliance, energy efficiency, and migration efficiency.

B. Biogeography-based Optimization Algorithm

The Biography-based Optimization (BBO) algorithm is a nature-inspired metaheuristic algorithm used for solving optimization problems. It draws inspiration from the behavior of biographical organisms and applies it to various domains, including virtual machine allocation in cloud computing. BBO employs population-based techniques to efficiently explore the solution space, optimize resource utilization, and minimize energy consumption. Its adaptive nature makes it well-suited for dynamic cloud environments, leading to improved performance and resource efficiency [32].

Zheng, et al. [33] introduced a novel method for solving the VM allocation problem using the BBO algorithm. Their approach combines a Multi-objective BBO algorithm with Differential Evolution (MBBO/DE). In this method, each habitat represents a placement solution, where a vector of length n corresponds to n VMs mapped to m servers. Initially, the habitats of MBBO/DE are created using a random function to ensure diversity. Then, all habitats are evaluated using a cosine model to calculate the migration rate for each habitat. The next step involves migrating Similar Individual Vectors (SIVs) through the differential evolution strategy, where habitats in MBBO/DE share their SIVs, resulting in the creation of new habitats. Mutated SIVs are generated using a Gaussian model, producing potentially new optimal solutions. Multiple objectives are measured, and non-dominated sorting is employed to evaluate the effectiveness of each solution in MBBO/DE. Finally, a greedy one-to-one deletion mechanism based on differential evolution is used to select better habitats. If a new habitat outperforms the reserved elite habitat, it replaces the previous habitat; otherwise, the previous habitat remains. This method offers advantages such as low power consumption, reduced migration time, and improved resource utilization. However, it has some drawbacks, including lower Quality of Service (QoS), limited scalability, and higher overhead migration.

Shi, et al. [34] introduced a new multi-objective optimization method that builds upon the classical BBO algorithm to fix the allotment issue of the VM. The method proposes enhancements in the form of enhanced Cosine migration and mutation models aimed at improving the efficiency of attaining optimal solutions. By incorporating these enhancements, the algorithm aligns better with the actual VM allocation scenario. Furthermore, the algorithm focuses on optimizing three key objectives: migration resource overhead, load balancing, and server power consumption. Thus, the proposed improvements in the migration and mutation models are geared towards efficiently achieving these optimization goals in the context of VM allocation.

Zheng, et al. [35] have suggested a new approach for the VM allocation problem based on a multi-objective BBO algorithm in cloud-based environments. In this technique, after the initial parameters of the system are determined, the within-subsystem immigration is based on the ranking of the island within-subsystem, so the elementary population is the number of subsystems multiplied by the number of islands per

subsystem. Then, the subsystem migration is based on the likeness level among the subsystem pairs. The next step concerns the probability of mutation of each island, so if an island is selected for the mutation, the SIV is randomly selected from the island and replaced with a new randomly generated SIV. Finally, a modified non-dominant ranking system is used to select Nelite elites from the optimal answers of each subsystem and the elite of the previous generation. Then, if the repetition doesn't end, a new population is created for the next generation, and the Nelite matrices are replaced by the elitists generated in the last stage. Therefore, reduced power consumption, increased resource utilization, and high performance are the advantages of the method. However, this method suffers from high overhead migration, high time migration, and high SLA.

C. Artificial Bee Colony Algorithm

The Artificial Bee Colony (ABC) algorithm is a population-based metaheuristic optimization technique miming honeybee foraging behavior. It employs a diverse population of artificial bees, including employed bees, onlookers, and scout bees, to explore the solution space. The employed bees exploit food sources based on local information, while onlookers choose food sources based on quality. Scout bees introduce diversity by randomly searching for new food sources. Through iterative search processes, the ABC algorithm effectively balances exploration and exploitation, making it suitable for solving various optimization problems [36].

In their work, Jiang, et al. [37] have introduced an innovative VM consolidation policy that utilizes the ABC algorithm in conjunction with the Data-intensive Energy Evaluation Model. The objective of this policy is to minimize energy consumption costs and improve the SLAV value. By employing the ABC algorithm, the VM migration decisions are optimized, resulting in enhanced VM allocation. Moreover, the authors propose an energy efficiency evaluation model specifically designed for data-intensive tasks within green data centers. The model takes into account two influential factors: the CPU and GPU interest rates, which are observed through the behavior of data-intensive tasks in data centers. The VM allocation policy, based on the ABC utility, is then employed to minimize energy consumption and achieve satisfactory SLAV values. This approach effectively reduces energy consumption, the number of VM migrations, and SLAs while maximizing resource utilization and maintaining a high quality of service. However, it is important to note that this technique overlooks scalability and the overhead involved in the migration process.

Li, et al. [38] have introduced a novel approach called energy-aware dynamic virtual machine consolidation (EC-VMC) for VM migration. This method aims to address the constraints related to potential overloading across multiple sources. It employs a set of algorithms for selecting and placing virtual machines, taking into account the limitations of different sources within a physical machine while considering additional overhead. Additionally, the algorithm integrates and simulates the foraging behaviors of an artificial bee colony, treating the mapping relationship between PMs and VMs as a food source. By utilizing the search mechanism and optimization strategy of the ABC algorithm, the optimal mapping relationship is achieved globally, considering the multi-source constraints between PMs and VMs. Experimental results demonstrate that this method effectively reduces the VM's relocations number and power usage while ensuring large resource utilization and performance. However, it should be noted that this approach faces challenges related to overhead migration and scalability.

D. Chicken Swarm Optimization Algorithm

Chicken Swarm Optimization (CSO) is a swarm intelligence algorithm introduced in 2014 by Meng et al. It is a randomized optimization algorithm inspired by the group behavior of chickens searching for food. The CSO algorithm categorizes the chickens into roosters, hens, and chicks based on their fitness levels. Each type of chicken employs a different searching strategy, and the entire chicken swarm updates itself over multiple generations. One of the key strengths of the CSO algorithm is its ability to prevent getting trapped in local optima and instead find the optimal global solution for a given optimization problem. By mimicking the cooperative foraging behavior of chickens, the algorithm explores the search space effectively [39]. Tian et al. [40] have presented a new algorithm for Virtual Machine Consolidation (VMC) based on the Chicken Swarm Optimization model with deadlock-free migration (DFM-CSO). In this method, the consolidation VM design, which turns the VMC problem into a vector packing optimization problem according to deadlock-free migration, minimizes energy consumption. This algorithm is specified by the "one-step look-ahead with n VMs migration in parallel (OSLA-NVMIP)" method, which validates the VM migration and resets the aim physical host, also records the migration order for each solution placement so that VM transfer can be done according to the migration sequence. Furthermore, the algorithm reduces energy consumption, improves resource utilization, and reduces migration times. However, it exhibits low performance, low scalability, and a high migration overhead.

E. Cuckoo Search Algorithm

The Cuckoo Search Algorithm (CSA) is a metaheuristic optimization algorithm inspired by the behavior of cuckoo birds. It simulates the cuckoos' parasitic breeding manner, where they put their eggs in the nests of different bird species. In the CSA, each solution is represented as a cuckoo egg, and the nests represent potential solutions. The algorithm employs three key steps: (1) Lévy flights, which simulate the random walk of cuckoos, are used to generate new solutions; (2) some eggs are randomly selected for replacement to introduce diversity; and (3) the quality of eggs in the nests is assessed using an objective function, and the nests with higher quality solutions are more likely to survive. By iteratively repeating these steps, the CSA effectively explores the solution space and converges toward optimal or near-optimal solutions for optimization problems [41].

Joshi and Kaur [42] have proposed a new approach to solving the virtual machine consolidation issues based on a cuckoo search to save both power consumption and resource utilization in cloud space. The cuckoo search has been used to solve the problem of consolidation of the VM, which is similar to the bin packing problem. Therefore, in this method, it is

shown that the cuckoo search algorithm is a useful way to solve bin packing problems compared to other heuristics algorithms and can be efficiently used for the VM consolidation approach for both resource utilization and power consumption in the cloud network. This approach enables great source employment, less power usage, and great quality of service, reducing the number of PMs. Nevertheless, this method ignored overhead migration, time migration, scalability, and SLA violations.

Naik, et al. [43] have presented a new approach using a multi-objective Fruit-fly hybridized Cuckoo Search algorithm for virtual machine consolidation in the cloud environment. This algorithm is based on VM migration, which is used to reduce the over-provisioning of a PM by consolidating VMs on an underutilized physical machine. Thus, it relies on VM migration plans, which reduce the migration of PMs, consolidate under-utilized PMs, and select migrating VMs based on threshold significance. Multi-objective criteria should be used instead of single-objective criteria for placing and integrating VMs. The best host is detected from the host list. Compared with other methods, this method uses less power and resources, and it also reduces the relocations number. However, scalability, service quality, and violations of service level agreements were ignored.

F. Firefly Algorithm

The Firefly algorithm (FA), developed by Yang in 2008, was inspired by the flashing behavior of fireflies and uses swarm intelligence. Fireflies are also integrated to attract one another based on brightness. Therefore, light is considered a factor for absorption for flashing fireflies, and less brightness leads to a lighter situation. With increasing distance, attractiveness decreases directly with brightness. The goal function could be equated to the light to get the optimal solution [44].

Perumal and Murugaiyan [45] have proposed a novel for server consolidation and virtual machine placement problems that use a firefly colony and fuzzy firefly colony algorithms. The purpose of this method is to reduce the number of PMs utilized and solve the problem of VM disposition to achieve the right method with lower energy usage and lower waste of resources. Therefore, in this approach, both mentioned goals are considered simultaneously in the procedure of figuring out the ideal arrangement for setting up the virtual machine. One of the benefits of using fuzzy firefly colonies to solve virtual machine placement problems is that, unlike other algorithms, this algorithm works quickly and reduces randomization when searching for the optimal solution so that it will perform well. The fuzzy firefly colony method is provided with a rule of fuzzy probability to handle the behavior of the firefly probe with uncertainty. This technique reduces power consumption, maximizes resource utilization, and enhances performance. However, it ignores the number of migrations, SLA, migration time, and scalability.

John and Bindu [46] have proposed two new methods, the first being exploratory energy and temperature-based integrated temperature (HET-VC) and the second, virtual machine-based firefly and temperature-based integration (FET-VC). These methods are used to find the best solution in the problem space in improving the integration problems of virtual machines. Besides, in the proposed method, the current position and the next position after the migration of virtual machines will be updated at regular intervals. This method tries to decrease the number of transmissions and SLA violations, as well as decrease the energy consumption of servers. Besides, these two algorithms use high-performance servers that use the least CPU and RAM, as well as low temperatures, to integrate virtual machines. It suffers from high system overhead, long migration times, and limited scalability.

G. Genetic Algorithm

Genetic Algorithm (GA) was introduced in 1975 as a population-based optimization method with an evolutionary process. GA has been widely used to optimize VM integration parameters and also to solve optimization problems. In the GA algorithm, each chromosome represents a possible solution, and these solutions are combined by a string of genes. A fitness function is also used to check if the chromosome for the environment is right, and then crossover and mutation operations are used to produce offspring for the new population. The fitness function is used to assess the quality of each offspring, and this action is repeated until sufficient offspring are manufactured. Quick convergence and high processing time are the disadvantages of this algorithm [47].

Joseph, et al. [48] has proposed a novel technique to allocate virtual machines using the Family Gene approach. In this method, the host list and VM list are used for optimal mapping, and then the whole process is distributed among different families running in parallel in this module using the FGA module. Also, by using a self-adjusting mutation operator, an attempt has been made to reduce the rate of early convergence in this approach. Simple mutations are made to construct families. The resulting chromosomes, which are slightly different from each other, are placed in a family. Each family is processed "k" times. In the absence of a better individual being found by then, the family will be destroyed, and the next family will be taken in their place. In this way, the quality of each individual is determined by the amount of fitness value associated with that individual. Physical resource utilization is the purpose of the proposed method. In order to accomplish this, each individual's performance is evaluated so that if the individual fails, the impossible solution becomes feasible. This method performs well regarding energy consumption, resource utilization, and VM migrations. Nevertheless, it endures large upward and limited versatility.

Wu and Ishikawa [49] have presented an Energy-aware method for dynamic VM consolidation to reduce the energy consumed in heterogeneous cloud environments. In this method, the migration cost of the virtual machine and the amount of energy savings through the dynamic integration of the virtual machine in heterogeneous cloud data centers are investigated. Thus, a general assessment is defined by two opposing goals, namely, the cost of migration and energy saving. For this purpose, in this method, a merge score function has been arranged for total assessment based on the immigration cost estimation method and a high-limit estimation method for maximum saving power. Therefore, the IGGA algorithm, a kind of genetic grouping algorithm, is designed to enhance the merging score by a greedy heuristic

and an exchange action. The offered method offers better scalability, migration cost, QoS, power consumption, and resource utilization. But it ignores migration overhead, migration time, and SLA violation.

Wu, et al. [50] have presented a new method based on the evolutionary computing Adaptive Genetic Algorithm (A-GA) for the VM consolidation approach. This method has good performance in issues related to virtual machine consolidation in the areas of dynamism in resource utilization, VM migration number, failure reduction, SLA breach, and energy efficiency. Therefore, the suggested system can be used to manage large-scale cloud resources, reduce energy consumption, and increase QoS. Furthermore, the Minimum Migration Time (MMT) by VM selection policy based on VM placement or allocation has performed better than the Maximum Correlation (MC) and Random Selection (RS) method for energy utilization and QoS in minimal failure time and SLA violation. Furthermore, less power usage, large source employment, less relocation duration, less SLA violation, and high QoS are some of the advantages of this algorithm. However, high migration overhead and low scalability are some of the weaknesses of this algorithm.

Theja and Babu [51] have proposed an improved energy-saving asset allotment on the basis of a genetic method taking the power usage in cooler systems and IT items into account. They have focused on the problem of resource allocation and have evaluated various important issues in the cloud network, namely power and energy usage, VM's relocations number, and SLA violations. Also, the primary importance of considering various system parameters such as CPU, RAM, and network bandwidth in the decision-making process in their work has been evaluated. Most importantly, a comprehensive multi-objective policy has been used as a solution to the problem of resource allocation based on genetic algorithms to simultaneously reduce the power consumption of IT equipment and CRAC units.

Arianyan, et al. [52] have proposed a new approach with mixed-integer nonlinear programming (MINLP) formula for virtual machine integration problems in cloud computing that uses the genetic algorithm (GA) named energy and cost-aware VM consolidation or (ECVMC). They designed a mathematical model to reduce power consumption and costs using effective VM consolidation. In this method, by decreasing the number of active servers, the use of resources is maximized and minimizes total power consumption. Besides, this method decreases the placement process cost by discovering the optimal solution. As a result, low cost, low power consumption, and high resource utilization are some of the benefits of this method.

H. Greedy Random Adoptive Search Procedure Algorithm

The Greedy Random Adaptive Search Procedure (GRASP) algorithm was presented in 1989 to solve combinatorial problems by Feo and Resende. The GRASP is a meta-heuristic exploration based on a structural exploration that can create various initial solutions. This is a repetitive random optimization method in which each iteration involves two steps: one construction step and one local search improvement step. The construction step is an iterative greedy and adaptive process, and the second phase is a method of improving the local search for the initial solution, both of which are repeated for the maximum number of GRASP iterations. In the greedy construction stage, the list of solutions is made on the basis of greedy performance by randomly demoing the resolution habitat. In the second stage, a local search is performed to detect the best present solution from the formerly created solution list [53].

Ilager, et al. [54] have suggested a new method for a dynamic consolidation framework for the comprehensive controlling of cloud sources through enhancing cooling and computing systems. Via the Energy and Thermal-Aware Scheduling (ETAS) algorithm, they have controlled the trade among aggressive consolidation and the scattered distribution of VMs, which affects energy and hotspots. Also, the ETAS algorithm is designed to handle the trade-off between expense and time efficiency and could be adjusted as needed. Besides, based on the needs of the system, this algorithm is customizable to manage computation time and solution quality. Extensive experiments have been performed using real-world traces with precise thermal and power samples on this algorithm. Furthermore, this technique has advantages such as high QoS, less power usage, large source employment, and less relocation duration. However, this algorithm ignored scalability and performance.

I. Harmony Search Algorithm

The Harmony Search (HS) algorithm is a metaheuristic optimization technique inspired by the improvisation process of musicians in a musical ensemble. It mimics the harmony creation process, where musicians adjust their musical notes to achieve pleasing melodies. In the HS algorithm, a population of solution vectors represents musical harmony, and each element of the vector corresponds to a decision variable. Initially, random solutions are generated, and a fitness function evaluates their quality. Through iterations, new solution vectors are created by considering three main operators: pitch adjustment, harmony memory consideration, and randomization. These operators guide the search process towards better solutions. By continuously refining the harmony vectors based on their fitness values, the HS algorithm aims to find the optimal or near-optimal solution for the given optimization problem. The algorithm has demonstrated effectiveness in solving various complex optimization problems [55].

Fathi and Khanli [56] have suggested a new method based on the HS algorithm for the active allotment of VMs, which has been proven to be effective in power management systems. To solve the problem of virtual machine allocation, this algorithm has been used to detect an optimal solution. It defines a multi-objective function that takes into account both the number of silent PMs and the number of migrations and data center power consumption in each state. Also, some of the benefits of this method result in faster results, such as no need for basic parameters and no need to extract data. This method dramatically reduces energy utilization, resulting in fewer active PMs while maintaining QoS service quality standards. Furthermore, this algorithm has low SLA violations, low VM migration counts, high QoS, and low migration times. However, it ignores scalability and resource utilization.

Kim, et al. [57] have proposed a new method using a grouping agent approach and a meta-heuristic harmonic search algorithm for effectively solving the virtual machine allocation problem. In this algorithm, to solve VMC, the population number should be increased in proportion to increasing the size of the problem to certify search variety. In this case, the cost of searching for a solution may increase, so knowing that the number of VMs in virtualization is greater than the number of PMs, VMCs are targeted in such environments. Reflecting on this feature, the length of the solution will be reduced if the PM-based solution agent is classified instead of the VM-based redesign. Therefore, as the size of the problem increases, population growth becomes more linear and efficient search results. In this method, VMs have a unique workload pattern because they are provided in real-time using network resources. When migrating a virtual machine, depending on the VM time-series pattern as well as the number of resources allocated, the PM could be stabled by maintaining a good PM-VM map. Therefore, some advantages of this algorithm are low migration cost, low power consumption, high QoS, and high performance. However, this method suffers from high SLA violations, high overload migration, and low scalability.

J. Sine-Cosine Algorithm

The sine-cosine Algorithm (SCA) is an optimization method based on a population that performs the optimization function by applying a set of random solutions. These solutions are computed by a target function repeatedly in each iteration. In this method, the random set is repeatedly measured by a target function and improved by using a set of rules that is the core of this optimization method. The likelihood of global optimization increases with the right number of random solutions. In the optimization algorithms with enough random solutions and optimization steps (iterations), the possibility of discovering the global optimum increases [58].

Jayasena, et al. [59] have proposed a multi-objective approach based on the Multi-objective Sine Cosine Algorithm (MOSCA) for virtual machine allocation problems. This method is presented using a Multi-objective Evolutionary Algorithm based on Decomposition (MOEAD) and a Non-Dominated Sorting Genetic Algorithm (NSGAII). This approach generally focuses on evaluating and comparing multi-objective algorithms to find the optimal solution, as well as developing the MOSCA to figure out the resolution for the proposed VMC model. In this method, three conflicting goals discussed for the high energy efficiency of VMC are to reduce power consumption, achieve better SLAVs, and maximize MTBHS time. Therefore, the advantages of this algorithm are low power consumption, high resource utilization, low SLA violation, and high QoS. However, this method endures less versatility, high upward systems, and high migration time.

K. Penguin Search Optimization Algorithm

Penguins Search Optimization Algorithm (PSOA), introduced by Gheraibia and Moussaoui [60], is a meta-heuristic algorithm based on a common penguin hunting strategy. The algorithm is inspired by the penguin's search strategy, as they can combine their attempts and synchronize their dives to optimize global energy within the collective looking out and nutrition strategy. Each of them is represented by its place as well as the number of fish consumed. The distribution of penguins is determined by the number of prior fish in the area. Penguins are divided into groups, and they search in different situations. If the result of the number of dives is successful, Penguins return to the land and share important points such as places and a set of available food resources.

Jayasena, et al. [59] have presented a new method, according to the PSOA, for consolidating multiple virtual machines in the distribution of cloud-based hosting surroundings. This method uses the PSOA algorithm to make an economic VM consolidation and also supports scheduling for multiple VMs with different applications. The system can allocate VM resources for applications, reducing the number of VMs required. Therefore, the current strategy, by automatically planning and implementing several programs simultaneously with the specified features, provides the desired thinking in a very dedicated hosting space. Therefore, this algorithm, using placement rules, can lead to the Optimization of various multi-objective problems. Finally, this algorithm provides low power consumption and high QoS and reduces the number of PMs. However, this method ignored scalability and SLA violations, as well as resource utilization.

L. Particle Swarm Optimization Algorithm

The Particle Swarm Optimization (PSO) algorithm was introduced in 1995 by Kenney and Eberhart to find suitable solutions to continuous space optimization problems. The PSO algorithm uses a set of potentially random solutions (particles) to discover proper solutions to optimization problems. For each particle, the speed required to move in the search space is allocated. In each iteration, the speed of each particle is adjusted according to its best position and the position of the best particle in the total population. In this algorithm, each particle tracks its coordinates in the problem space with the optimal solution (fitness) obtained because each particle has internal memory. This algorithm has other benefits, such as combining local search methods with global search methods and trying to balance exploration and exploitation [61].

Dashti and Rahmani [62] have proposed a new method with a hierarchical architecture to meet the needs of producers and consumers, as well as a new service for scheduling consumer tasks in the PaaS layer. They have improved the allocation of dynamic resources and gained more benefits in the Personal Data Center by using the PSO algorithm. Therefore, PSO is used to ensure the quality of user services and reduce energy efficiency in redistributing migratory virtual machines in the overloaded host. In this method, low-load hosts and their power are consolidated to save energy. Also, there are balanced overload hosts used to ensure the quality of service during response times and deadlines. Therefore, this method offers less power usage, great QoS, and high source employment but ignores SLA violations and migration time.

Li, et al. [63] have presented a new approach based on several assets and energy-saving relocation and VMs' integration method under dynamic load and have arranged a double-threshold algorithm with multi-resource utilization for the VMs migration. One of the common problems of traditional heuristic algorithms in the field of virtual machine

consolidation is falling into local optima, which is prevented by using the Modified Particle Swarm Optimization (MPSO) method. Therefore, this method can reduce energy utilization with the QoS guarantee. Finally, the advantages of this algorithm are less power usage, great source employment, great QoS, and a reduced number of PMs. However, high system overhead and high migration time are some of the disadvantages of this algorithm.

M. Simulated Annealing Algorithm

The Simulated Annealing (SA) algorithm mimics the annealing process in metallurgy, in which materials are heated and slowly cooled to reduce defects and improve structure. In SA, the search process starts with an initial solution and iteratively explores the solution space by randomly making changes. These changes can be either accepted or rejected based on a probability determined by a cooling schedule. Initially, the algorithm accepts more changes, resembling a high-temperature phase, allowing for exploration [64]. As the temperature decreases, the algorithm becomes more selective, favoring changes that lead to improved solutions. This transition from exploration to exploitation helps the algorithm escape local optima and converge towards global optima. The cooling schedule controls the balance between exploration and exploitation. SA has been widely used to solve combinatorial optimization problems and has shown effectiveness in finding near-optimal solutions, even in complex search spaces [65].

Marotta and Avallone [66] have proposed a new model for solving consolidation problems using the SA algorithm, which solves these problems by evaluating the attractiveness of possible VM migrations. In this method, the attractiveness of virtual machine migration is considered through the use of resources and the energy efficiency of physical servers. Therefore, to reduce energy waste, the consolidation of the virtual machine in live VM migration is used. Increasing the overall cost efficiency by decreasing the number of active nodes is one of the main goals of this method. Therefore, the advantages of this method are low energy consumption, high resource utilization, and high performance. However, this method ignores the scalability count of migration and time migration.

Rajabzadeh and Haghighat [67] have proposed a new method based on an energy-aware framework for the consolidation of VM to optimize energy consumption and SLA violation reduction. In this approach, the whole process of allocating and managing virtual machines is divided into smaller parts, and each of these small parts is improved by new algorithms or existing algorithms. First, the hosts are determined by the critical mode. In the first stage, the host overload status is classified once with the possibility of violating the SLA and again without considering it. Then, in the second stage, using this algorithm, virtual machines are selected to migrate from important hosts. Therefore, using the SA algorithm, according to the list of selected virtual machines for migration, new hosts are selected as the migration destination of virtual machines. Eventually, the low-load hosts are identified and selected by the low-load host selection algorithm, and then all the virtual machines deployed on these hosts migrate, and the host goes to sleep off. All steps in this method are done in the distributed mode, except for the VM's disposition, which is done in the centralized model. Finally, the advantages of this method are low energy consumption and high resource utilization, low SLA violation and low migration time, and high performance. However, high overload and low scalability are some of the weaknesses of this algorithm.

Telenyk, et al. [68] have proposed a new optimization method using the SA to solve dynamic virtual machine integration problems, an extension of the bin-packing problem. In this system, to provide a new configuration, the optimized objective function of the proposed Simulated Annealing algorithm is used. This approach uses temperature as a control parameter when exploring to optimally map the allocation of virtual machines for the objective function. At each stage of the algorithm evolution, using the search function, the virtual machine allocation map becomes a new neighborhood state. The acceptance indicator of each new VM allocation map is compared to the current allocation map indicator to decide whether it is accepted or not. Therefore, the advantages of this method are low SLA violation, high resource utilization, and low energy consumption. However, high system overload and this method suffer from low scalability, low QoS, and high migration time.

Telenyk, et al. [69] suggested a new approach to dynamic VM allocation based on the SA algorithm so that it can calculate the cost of migration by a consolidation program. In this algorithm, the overhead cost resulting from the live migration of the virtual machine is calculated using an estimated model. The goals of this algorithm are to minimize the total energy consumption and minimize the total migration overhead. A static threshold-based method can be used to identify underloaded hosts in this approach. In this method, by selecting a host as the source PM, all its host virtual machines are selected for migrated to the out. Lastly, the SA is used to select the destination host. Thus, this technique has reduced energy consumption, living migration costs, and SLA violations, as well as the overall migration overhead. This method, however, neglected migration times, quality of service, and scalability.

N. Shuffled Frog Leaping Algorithm

The Shuffled Frog Leaping Algorithm (SFLA) was proposed by Eusuff and Lansey as an intelligent optimization algorithm that mimics natural organisms' behavior. It adopts a population-based approach and cooperative search paradigm to tackle discrete optimization problems. The SFLA incorporates a shuffling approach, facilitating the exchange of information between local search and global Optimization [70]. In this algorithm, frogs in the population symbolize tasks, and the positions of the frogs denote the mapping between VMs and their respective tasks.

Luo, et al. [71] suggested a new scheme based on energy-aware resource allocation for dynamic consolidation of VMs. This approach focuses on the infrastructure in which custom VMs run on the unsuitable servers of a data center as a service-oriented model. To properly manage resources, by achieving dynamic consolidation of virtual machine host resources and changing the mode of idle or low-consumption hosts to energy-saving mode, it is possible to use more energy resources and adhere to SLAs. Thus, a new hybrid intelligent algorithm using

the Modified SFLA based on Extreme Optimization (EO) called (MSFLA-EO) is used to quickly and efficiently complete the dynamic allocation of VMs. This method increases the SFLA frog's leaping visibility and improves local search capability. Finally, low energy consumption, high performance, reduced migration cost, and high resource utilization are some of the advantages of this algorithm.

O. Tabu Search Algorithm

Tabu Search (TS) is a metaheuristic algorithm developed by Glover to address mixed optimization problems like the bin packing one. TS utilize local searching to avoid being trapped in local optima and continue the search until the desired results are satisfied. The algorithm incorporates random selection to enhance the efficiency of searching and employs a "Taboo List" to restrict search movements and avoid cycles. The Taboo list serves as a short-term memory, keeping track of recently explored solutions [72]. Nasim and Kassler [73] introduced a novel approach that addresses the trade-off between resource conflict protection and the additional energy costs associated with increased server loads while addressing uncertainties in VM resource demand. This method utilizes a TS-based method coupled with greedy heuristics to explore local issues and repetitive resistance. The proposed method demonstrates the capability to achieve near-optimal solutions through robust discoveries and accurately address time-constrained online data optimization.

Moreover, the approach can be customized to accommodate varying resistance levels required by data center operators. The benefits of this algorithm include reduced power consumption, low resource utilization, and minimal VM migrations. However, it should be noted that this method has some drawbacks, such as lower performance and limited scalability.

IV. DISCUSSION

Meta-heuristic algorithms play a crucial role in the allocation of VMs in cloud environments. This problem involves efficiently allocating VMs to physical servers in cloud infrastructure, aiming to optimize resource utilization, energy consumption, and overall system performance. Cloud environments are complex and changing, making it difficult to find the best solutions. Nature-inspired meta-heuristic algorithms draw inspiration from natural phenomena, biological systems, and evolutionary processes to solve complex optimization problems.

Table I outlines the key features, advantages, and limitations of the discussed algorithms. These algorithms offer unique advantages for tackling the VM allocation problem in cloud environments. Each algorithm has unique traits and demonstrates gradual advancements. For example, GA demonstrates significant progress in its capacity to tackle intricate challenges, demonstrating gradual improvements in exploring various arrangements for effective VM placement. FA reveals gradual improvements in achieving a trade-off between exploration and exploitation, leading to faster convergence even in dynamic contexts despite initial difficulties. In addition, ACO and PSO algorithms demonstrate improvements in their capacity to handle dynamic environments, gradually addressing the scaling issues found in their previous generations. The incremental solutions emphasize the progressive path of meta-heuristic approaches, depicting their potential to adapt and improve in dealing with the complexities of VM allocation in the constantly changing field of cloud computing.

- Global search capability: Meta-heuristic algorithms excel in exploring a vast solution space, allowing them to identify promising solutions even in highly complex and large-scale cloud environments. These algorithms, such as genetic algorithms, PSO, and ACO, utilize search strategies inspired by natural systems to navigate the search space and find good solutions efficiently.
- Robustness and adaptability: Cloud environments are dynamic, with varying workloads and resource demands. Nature-inspired algorithms possess inherent robustness and adaptability, enabling them to handle the dynamic nature of the virtual machine consolidation problem. They can quickly adapt to changes in the cloud environment, such as VM migrations, workload fluctuations, and server failures, and reoptimize the VM-to-server allocation.
- Parallelism and scalability: Nature-inspired algorithms are inherently parallelizable, allowing them to leverage the distributed nature of cloud environments. Parallelism makes it possible for algorithms to manage large-scale cloud systems with multiple VMs and servers, enhancing scalability and speeding up the optimization process.
- Exploration-exploitation trade-off: Balancing exploration and exploitation is necessary for solving the virtual machine allocation problem. Nature-inspired algorithms inherently possess exploration-exploitation mechanisms, ensuring a balance between exploring the search space for new configurations and exploiting promising solutions to optimize VM placement.

By leveraging the strengths of nature-inspired meta-heuristic algorithms, cloud providers can achieve efficient and optimized VM allocation, leading to improved resource utilization, reduced energy consumption, and enhanced overall system performance. These algorithms offer powerful tools for addressing the complexities of virtual machine allocation in cloud environments and contribute to advancing cloud computing technology.

TABLE I. COMPARISON OF META-HEURISTIC ALGORITHMS FOR VM ALLOCATION IN CLOUD COMPUTING

Algorithm	Key Features	Advantages	Limitations
Ant Colony Optimization (ACO)	Pheromone trails Inspiration from ant foraging behavior	Good exploration robustness to dynamic environments	Slow convergence Scalability issues
Biogeography-Based Optimization (BBO)	Migration and evolution-inspired algorithm	Consideration of migration rates, habitat suitability	Parameter tuning required Scalability issues
Artificial Bee Colony (ABC)	Inspired by the bee foraging behavior	Fast convergence Simplicity	May get trapped in local optima Sensitivity to parameters
Chicken Swarm Optimization (CSO)	Inspired by the chicken foraging behavior	Simplicity Good exploration	Convergence speed may be slow Performance variation
Cuckoo Search Algorithm (CS)	Inspired by cuckoo bird behavior	Fast convergence Able to escape local optima	Parameter sensitivity Scalability issues
Firefly Algorithm (FA)	Inspired by firefly flashing behavior	Exploration-exploitation balance Fast convergence	Slow convergence Difficulty with dynamic environments
Genetic Algorithm (GA)	Genetic operators: selection, crossover, mutation	Good exploration Able to handle complex problems	Selection of suitable operators Parameter tuning
Greedy Random Adoptive Search Procedure (GRASP)	A combination of greedy and random search	Simplicity Fast convergence	May get trapped in local optima
Harmony Search Algorithm (HS)	Music-inspired algorithm with improvisation and memory considerations	Good exploration Robustness to noisy environments	Parameter tuning required Slow convergence
Sine-Cosine Algorithm (SCA)	Inspired by sine and cosine functions	Simplicity Fast convergence	Sensitivity to parameters Difficulty with complex problems
Penguin Search Optimization Algorithm (PSOA)	Inspired by the hunting behavior of penguins	Good exploration Robustness to dynamic environments	Limited scalability Parameter sensitivity
Particle Swarm Optimization (PSO)	Particle movement and social interaction-based algorithm	Fast convergence Good exploration	Difficulty in balancing exploration and exploitation
Simulated Annealing (SA)	Inspired by the annealing process in metallurgy	Good exploration Able to escape local optima	Parameter tuning required Slower convergence
Shuffled Frog Leaping Algorithm (SFLA)	Simulates the frog leaping behavior	Good exploration Able to escape local optima	Slow convergence Scalability issues
Tabu Search Algorithm (TS)	Uses tabu list to avoid revisiting recently visited solutions	Good exploration Able to escape local optima	May get stuck in suboptimal solutions Parameter tuning

V. OPEN ISSUES

- Resource Utilization: One open issue is optimizing the utilization of computing resources in cloud data centers. Dynamic workload patterns pose a challenge in allocating virtual machines to maximize resource utilization and ensure efficient task execution. Efficient allocation and placement algorithms are needed to adjust resource allocation based on workload variations dynamically.
- Energy Efficiency: Another open issue is the improvement of energy efficiency in VM allocation and placement. In cloud data centers, energy consumption is a major concern, and efficient allocation of VMs can help reduce overall power usage. Developing algorithms and techniques that consider energy consumption metrics and prioritize energy-efficient VM placement is crucial for achieving sustainable and green cloud computing environments.
- Quality of Service (QoS): Ensuring QoS requirements for applications running in cloud environments is a critical challenge. VM allocation and placement algorithms should consider factors such as response time, latency, throughput, and reliability to meet the performance expectations of users. Balancing resource allocation for various applications and providing adequate QoS guarantees is an ongoing concern in cloud computing.
- Security and Privacy: VM allocation and placement involve sensitive data and require protection against security threats. Ensuring secure communication, data integrity, and privacy preservation during the allocation process is a significant challenge. Robust security mechanisms and privacy-preserving techniques need to be integrated into VM allocation algorithms to mitigate risks and protect the confidentiality and integrity of user data.
- Scalability: As cloud computing environments continue to grow in size and complexity, scalability becomes an open issue in VM allocation and placement. Efficient allocation is crucial when handling a large number of VMs, hosts, and data center resources. Developing scalable algorithms that can handle the increasing scale of cloud environments is essential for smooth and efficient VM allocation and placement.
- Multi-objective Optimization: VM allocation and placement involve conflicting objectives, including enlarging asset usage, minimizing power usage, and meeting QoS requirements. Multi-objective optimization techniques that can balance these objectives and provide trade-off solutions are necessary. Developing algorithms that can handle multiple optimization objectives and consider the preferences and constraints of different stakeholders is an open research area in VM allocation and placement.

- Dynamic and Real-time Allocation: Real-time allocation of VMs to adapt to dynamic workload changes and user demands is an ongoing challenge. VM allocation and placement algorithms should be able to handle sudden workload spikes, failures, or changes in resource availability. Designing dynamic and adaptive allocation strategies that can efficiently respond to real-time changes in the cloud environment is crucial for maintaining performance and responsiveness.

VI. CONCLUSION

The VM migration process necessitates the allocation of VMs in cloud computing. Its objective is to identify the optimal PM for each VM. With an increase in the number of VMs deployed in data centers, it becomes harder to control the utilization rate of host PMs. This study emphasized the importance of meta-heuristic algorithms to develop effective approaches to VM allocation. A description and evaluation of the algorithms in each group are presented based on a number of relevant criteria, including resource utilization, number of PMs and VMs, migration time, energy consumption, migration overhead, and SLA violation. The results of the research indicate that no meta-heuristic algorithm has been developed to allocate VMs capable of meeting all the requirements. The review has also identified several challenges and limitations associated with existing methods. These include scalability issues, high computational overhead, and the need for improved adaptation to dynamic environments. Furthermore, the lack of standardization and benchmarking frameworks for evaluating the performance of different algorithms remains a critical concern.

REFERENCES

[1] B. Pourghebleh, A. A. Anvigh, A. R. Ramtin, and B. Mohammadi, "The importance of nature-inspired meta-heuristic algorithms for solving virtual machine consolidation problem in cloud environments," Cluster Computing, pp. 1-24, 2021.

[2] K. Prasanna Kumar and K. Kousalya, "Amelioration of task scheduling in cloud computing using crow search algorithm," Neural Computing and Applications, vol. 32, no. 10, pp. 5901-5907, 2020.

[3] K. N. Qureshi, G. Jeon, and F. Piccialli, "Anomaly detection and trust authority in artificial intelligence and cloud computing," Computer Networks, vol. 184, p. 107647, 2021.

[4] O. Ali, A. Shrestha, J. Soar, and S. F. Wamba, "Cloud computing-enabled healthcare opportunities, issues, and applications: A systematic review," International Journal of Information Management, vol. 43, pp. 146-158, 2018.

[5] V. Hayyolalam, B. Pourghebleh, M. R. Chehrehzad, and A. A. Pourhaji Kazem, "Single - objective service composition methods in cloud manufacturing systems: Recent techniques, classification, and future trends," Concurrency and Computation: Practice and Experience, vol. 34, no. 5, p. e6698, 2022.

[6] A. A. Khan, M. Zakarya, R. Buyya, R. Khan, M. Khan, and O. Rana, "An energy and performance aware consolidation technique for containerized datacenters," IEEE Transactions on Cloud Computing, vol. 9, no. 4, pp. 1305-1322, 2019.

[7] A. Yousafzai et al., "Cloud resource allocation schemes: review, taxonomy, and opportunities," Knowledge and information systems, vol. 50, pp. 347-381, 2017.

[8] A. Peivandizadeh and B. Molavi, "Compatible authentication and key agreement protocol for low power and lossy network in IoT environment," Available at SSRN 4194715, 2022.

[9] B. Pourghebleh and V. Hayyolalam, "A comprehensive and systematic review of the load balancing mechanisms in the Internet of Things," Cluster Computing, pp. 1-21, 2019.

[10] M. Khodayari, J. Razmi, and R. Babazadeh, "An integrated fuzzy analytical network process for prioritisation of new technology-based firms in Iran," International Journal of Industrial and Systems Engineering, vol. 32, no. 4, pp. 424-442, 2019.

[11] M. Mohseni, F. Amirghafouri, and B. Pourghebleh, "CEDAR: A cluster-based energy-aware data aggregation routing protocol in the internet of things using capuchin search algorithm and fuzzy logic," Peer-to-Peer Networking and Applications, pp. 1-21, 2022.

[12] M. Momeni, D.-C. Wu, A. Razban, and J. Chen, "Data-driven Demand Control Ventilation Using Machine Learning CO2 Occupancy Detection Method," 2020.

[13] B. M. Jafari, M. Zhao, and A. Jafari, "Rumi: An Intelligent Agent Enhancing Learning Management Systems Using Machine Learning Techniques," Journal of Software Engineering and Applications, vol. 15, no. 9, pp. 325-343, 2022.

[14] S. R. Abdul Samad et al., "Analysis of the Performance Impact of Fine-Tuned Machine Learning Model for Phishing URL Detection," Electronics, vol. 12, no. 7, p. 1642, 2023.

[15] S. Vairachilai, A. Bostani, A. Mehbodniya, J. L. Webber, O. Hemakesavulu, and P. Vijayakumar, "Body Sensor 5 G Networks Utilising Deep Learning Architectures for Emotion Detection Based On EEG Signal Processing," Optik, p. 170469, 2022.

[16] W. Anupong et al., "Deep learning algorithms were used to generate photovoltaic renewable energy in saline water analysis via an oxidation process," Water Reuse, vol. 13, no. 1, pp. 68-81, 2023.

[17] S. Aghakhani, A. Larijani, F. Sadeghi, D. Martín, and A. A. Shahrakht, "A Novel Hybrid Artificial Bee Colony-Based Deep Convolutional Neural Network to Improve the Detection Performance of Backscatter Communication Systems," Electronics, vol. 12, no. 10, p. 2263, 2023.

[18] M. Bagheri, "Clustering Individual Entities Based on Common Features," 2021.

[19] D.-C. Wu, M. Momeni, A. Razban, and J. Chen, "Optimizing demand-controlled ventilation with thermal comfort and CO2 concentrations using long short-term memory and genetic algorithm," Building and Environment, vol. 243, p. 110676, 2023.

[20] S. P. Rajput et al., "Using machine learning architecture to optimize and model the treatment process for saline water level analysis," Journal of Water Reuse and Desalination, 2022.

[21] M. Dorigo, M. Birattari, and T. Stutzle, "Ant colony optimization," IEEE computational intelligence magazine, vol. 1, no. 4, pp. 28-39, 2006.

[22] A. Esnault, E. Feller, and C. Morin, "Energy-aware distributed ant colony based virtual machine consolidation in IaaS Clouds bibliographic study," Informatics Mathematics (INRIA), pp. 1-13, 2012.

[23] E. Feller, C. Morin, and A. Esnault, "A case for fully decentralized dynamic VM consolidation in clouds," in 4th IEEE International Conference on Cloud Computing Technology and Science Proceedings, 2012: IEEE, pp. 26-33.

[24] F. Farahnakian et al., "Energy-aware dynamic VM consolidation in cloud data centers using ant colony system," in 2014 IEEE 7th International Conference on Cloud Computing, 2014: IEEE, pp. 104-111.

[25] M. H. Ferdaus, M. Murshed, R. N. Calheiros, and R. Buyya, "Virtual machine consolidation in cloud data centers using ACO metaheuristic," in Euro-Par 2014 Parallel Processing: 20th International Conference, Porto, Portugal, August 25-29, 2014. Proceedings 20, 2014: Springer, pp. 306-317.

[26] F. Farahnakian et al., "Using ant colony system to consolidate VMs for green cloud computing," IEEE Transactions on Services Computing, vol. 8, no. 2, pp. 187-198, 2014.

[27] P. Matre, S. Silakari, and U. Chourasia, "Ant colony optimization (ACO) based dynamic VM consolidation for energy efficient cloud computing," International Journal of Computer Science and Information Security, vol. 14, no. 8, p. 345, 2016.

[28] A. Ashraf and I. Porres, "Multi-objective dynamic virtual machine consolidation in the cloud using ant colony system," International Journal of Parallel, Emergent and Distributed Systems, vol. 33, no. 1, pp. 103-120, 2018.

[29] M.-H. Malekloo, N. Kara, and M. El Barachi, "An energy efficient and SLA compliant approach for resource allocation and consolidation in cloud computing environments," Sustainable Computing: Informatics and Systems, vol. 17, pp. 9-24, 2018.

[30] A. Aryania, H. S. Aghdasi, and L. M. Khanli, "Energy-aware virtual machine consolidation algorithm based on ant colony system," Journal of Grid Computing, vol. 16, no. 3, pp. 477-491, 2018.

[31] H. Xiao, Z. Hu, and K. Li, "Multi-objective VM consolidation based on thresholds and ant colony system in cloud computing," IEEE Access, vol. 7, pp. 53441-53453, 2019.

[32] D. Simon, "Biogeography-based optimization," IEEE transactions on evolutionary computation, vol. 12, no. 6, pp. 702-713, 2008.

[33] Q. Zheng, J. Li, B. Dong, R. Li, N. Shah, and F. Tian, "Multi-objective optimization algorithm based on bbo for virtual machine consolidation problem," in 2015 IEEE 21st International Conference on Parallel and Distributed Systems (ICPADS), 2015: IEEE, pp. 414-421.

[34] K. Shi, H. Yu, F. Luo, and G. Fan, "Multi-Objective Biogeography-Based Method to Optimize Virtual Machine Consolidation," in Proceedings of the International Conference on Software Engineering and Knowledge Engineering, 2016, pp. 225-230, doi: https://doi.org/10.18293/SEKE2016-151.

[35] Q. Zheng et al., "Virtual machine consolidated placement based on multi-objective biogeography-based optimization," Future Generation Computer Systems, vol. 54, pp. 95-122, 2016.

[36] D. Karaboga and B. Akay, "A comparative study of artificial bee colony algorithm," Applied mathematics and computation, vol. 214, no. 1, pp. 108-132, 2009.

[37] J. Jiang, Y. Feng, J. Zhao, and K. Li, "DataABC: A fast ABC based energy-efficient live VM consolidation policy with data-intensive energy evaluation model," Future generation computer systems, vol. 74, pp. 132-141, 2017.

[38] Z. Li, C. Yan, L. Yu, and X. Yu, "Energy-aware and multi-resource overload probability constraint-based virtual machine dynamic consolidation method," Future Generation Computer Systems, vol. 80, pp. 139-156, 2018.

[39] X. Meng, Y. Liu, X. Gao, and H. Zhang, "A new bio-inspired algorithm: chicken swarm optimization," in Advances in Swarm Intelligence: 5th International Conference, ICSI 2014, Hefei, China, October 17-20, 2014, Proceedings, Part I 5, 2014: Springer, pp. 86-94.

[40] F. Tian, R. Zhang, J. Lewandowski, K.-M. Chao, L. Li, and B. Dong, "Deadlock-free migration for virtual machine consolidation using chicken swarm optimization algorithm," Journal of Intelligent & Fuzzy Systems, vol. 32, no. 2, pp. 1389-1400, 2017.

[41] Q. Yang, H. Huang, J. Zhang, H. Gao, and P. Liu, "A collaborative cuckoo search algorithm with modified operation mode," Engineering Applications of Artificial Intelligence, vol. 121, p. 106006, 2023.

[42] S. Joshi and S. Kaur, "Cuckoo search approach for virtual machine consolidation in cloud data centre," in International Conference on Computing, Communication & Automation, 2015: IEEE, pp. 683-686.

[43] B. B. Naik, D. Singh, A. B. Samaddar, and S. Jung, "Developing a cloud computing data center virtual machine consolidation based on multi-objective hybrid fruit-fly cuckoo search algorithm," in 2018 IEEE 5G World Forum (5GWF), 2018: IEEE, pp. 512-515.

[44] X.-S. Yang and X. He, "Firefly algorithm: recent advances and applications," International journal of swarm intelligence, vol. 1, no. 1, pp. 36-50, 2013.

[45] B. Perumal and A. Murugaiyan, "A firefly colony and its fuzzy approach for server consolidation and virtual machine placement in cloud datacenters," Advances in Fuzzy Systems, vol. 2016, 2016.

[46] N. P. John and V. Bindu, "Energy-Efficient Hybrid Firefly–Crow Optimization Algorithm for VM Consolidation," in Intelligent Computing and Communication: Proceedings of 3rd ICICC 2019, Bangalore 3, 2020: Springer, pp. 413-427.

[47] B. Alhijawi and A. Awajan, "Genetic algorithms: Theory, genetic operators, solutions, and applications," Evolutionary Intelligence, pp. 1-12, 2023.

[48] C. T. Joseph, K. Chandrasekaran, and R. Cyriac, "A novel family genetic approach for virtual machine allocation," Procedia Computer Science, vol. 46, pp. 558-565, 2015.

[49] Q. Wu and F. Ishikawa, "Heterogeneous virtual machine consolidation using an improved grouping genetic algorithm," in IEEE 17th International Conference on High Performance Computing and Communications, 2015: IEEE, pp. 397-404.

[50] Q. Wu, F. Ishikawa, Q. Zhu, and Y. Xia, "Energy and migration cost-aware dynamic virtual machine consolidation in heterogeneous cloud datacenters," IEEE transactions on Services Computing, vol. 12, no. 4, pp. 550-563, 2016.

[51] P. R. Theja and S. K. Babu, "Evolutionary computing based on QoS oriented energy efficient VM consolidation scheme for large scale cloud data centers," Cybernetics and Information Technologies, vol. 16, no. 2, pp. 97-112, 2016.

[52] E. Arianyan, H. Taheri, and S. Sharifian, "Multi Target Dynamic VM Consolidation in Cloud Data Centers Using Genetic Algorithm," Journal of Information Science & Engineering, vol. 32, no. 6, 2016.

[53] M. G. Resende and C. C. Ribeiro, "Greedy randomized adaptive search procedures: Advances, hybridizations, and applications," Handbook of metaheuristics, pp. 283-319, 2010.

[54] S. Ilager, K. Ramamohanarao, and R. Buyya, "ETAS: Energy and thermal - aware dynamic virtual machine consolidation in cloud data center with proactive hotspot mitigation," Concurrency and Computation: Practice and Experience, vol. 31, no. 17, p. e5221, 2019.

[55] M. T. Abdulkhaleq et al., "Harmony search: Current studies and uses on healthcare systems," Artificial Intelligence in Medicine, p. 102348, 2022.

[56] M. H. Fathi and L. M. Khanli, "Consolidating VMs in green cloud computing using harmony search algorithm," in Proceedings of the 2018 International Conference on Internet and e-Business, 2018, pp. 146-151.

[57] M. Kim, J. Hong, and W. Kim, "An Efficient Representation Using Harmony Search for Solving the Virtual Machine Consolidation," Sustainability, vol. 11, no. 21, p. 6030, 2019.

[58] S. Mirjalili, "SCA: a sine cosine algorithm for solving optimization problems," Knowledge-based systems, vol. 96, pp. 120-133, 2016.

[59] K. Jayasena, L. Li, M. Abd Elaziz, S. Xiong, and J. Xiang, "Optimizing the energy efficient VM consolidation by a multi-objective algorithm," in 2018 IEEE 22nd International Conference on Computer Supported Cooperative Work in Design ((CSCWD)), 2018: IEEE, pp. 81-86.

[60] Y. Gheraibia and A. Moussaoui, "Penguins search optimization algorithm (PeSOA)," in Recent Trends in Applied Artificial Intelligence: 26th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2013, Amsterdam, The Netherlands, June 17-21, 2013. Proceedings 26, 2013: Springer, pp. 222-231.

[61] T. M. Shami, A. A. El-Saleh, M. Alswaitti, Q. Al-Tashi, M. A. Summakieh, and S. Mirjalili, "Particle swarm optimization: A comprehensive survey," IEEE Access, 2022.

[62] S. E. Dashti and A. M. Rahmani, "Dynamic VMs placement for energy efficiency by PSO in cloud computing," Journal of Experimental & Theoretical Artificial Intelligence, vol. 28, no. 1-2, pp. 97-112, 2016.

[63] H. Li, G. Zhu, C. Cui, H. Tang, Y. Dou, and C. He, "Energy-efficient migration and consolidation algorithm of virtual machines in data centers for cloud computing," Computing, vol. 98, no. 3, pp. 303-317, 2016.

[64] S. Mahmoudinazlou, A. Alizadeh, J. Noble, and S. Eslamdoust, "An improved hybrid ICA-SA metaheuristic for order acceptance and scheduling with time windows and sequence-dependent setup times," Neural Computing and Applications, pp. 1-19, 2023.

[65] K. Amine, "Multiobjective simulated annealing: Principles and algorithm variants," Advances in Operations Research, vol. 2019, 2019.

[66] A. Marotta and S. Avallone, "A simulated annealing based approach for power efficient virtual machines consolidation," in 2015 IEEE 8th international conference on cloud computing, 2015: IEEE, pp. 445-452.

[67] M. Rajabzadeh and A. T. Haghighat, "Energy-aware framework with Markov chain-based parallel simulated annealing algorithm for dynamic management of virtual machines in cloud data centers," The Journal of Supercomputing, vol. 73, no. 5, pp. 2001-2017, 2017.

[68] S. Telenyk, E. Zharikov, and O. Rolik, "Consolidation of Virtual Machines Using Stochastic Local Search," in Advances in Intelligent Systems and Computing II: Selected Papers from the International Conference on Computer Science and Information Technologies, CSIT 2017, September 5-8 Lviv, Ukraine, 2018: Springer, pp. 523-537.

[69] S. Telenyk, E. Zharikov, and O. Rolik, "Consolidation of virtual machines using simulated annealing algorithm," in 2017 12th International Scientific and Technical Conference on Computer Sciences and Information Technologies (CSIT), 2017, vol. 1: IEEE, pp. 117-121.

[70] B. B. Maaroof et al., "Current Studies and Applications of Shuffled Frog Leaping Algorithm: A Review," Archives of Computational Methods in Engineering, pp. 1-16, 2022.

[71] J.-p. Luo, X. Li, and M.-r. Chen, "Hybrid shuffled frog leaping algorithm for energy-efficient dynamic consolidation of virtual machines in cloud data centers," Expert Systems with Applications, vol. 41, no. 13, pp. 5804-5816, 2014.

[72] V. K. Prajapati, M. Jain, and L. Chouhan, "Tabu search algorithm (TSA): A comprehensive survey," in 2020 3rd International Conference on Emerging Technologies in Computer Engineering: Machine Learning and Internet of Things (ICETCE), 2020: IEEE, pp. 1-8.

[73] R. Nasim and A. J. Kassler, "A robust Tabu Search heuristic for VM consolidation under demand uncertainty in virtualized datacenters," in 2017 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID), 2017: IEEE, pp. 170-180.