International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2024/224431 Volume 11, Issue 1, January – February (2024)

RESEARCH ARTICLE

Optimizing Virtual Machines Placement in a Heterogeneous Cloud Data Center System

Aristide Ndayikengurukiye
Intelligent Processing and Security of Systems Team, Faculty of Sciences, Mohammed V University, Rabat, Morocco.
lemure.dede@gmail.com

Abderrahmane Ez-Zahout
Intelligent Processing and Security of Systems Team, Faculty of Sciences, Mohammed V University, Rabat, Morocco.
abderrahmane.ezzahout@um5.ac.ma

Fouzia Omary
Intelligent Processing and Security of Systems Team, Faculty of Sciences, Mohammed V University, Rabat, Morocco.
omary@fsr.ac.ma

Received: 20 July 2023 / Revised: 02 October 2023 / Accepted: 25 January 2024 / Published: 26 February 2024

Abstract – In a cloud computing environment, good resource management remains a major challenge for its good operation. Implementing virtual machine placement (VMP) on physical machines helps to achieve various objectives, such as resource allocation, load balancing, energy consumption, and quality of service. VMP (virtual machine placement) in the cloud is critical, so it's important to audit its implementation. It must take into account the resources of the physical server, including CPU, RAM, and storage. In this paper, a metaheuristic algorithm based on the Grey Wolf Optimization (GWO) method is used to optimize the placement of virtual machines in a cloud environment, effectively minimizing the number of active virtual machines used to host virtual servers. Experimental results demonstrate the effectiveness of the proposed method, called Grey Wolf Optimization for Virtual Machine Placement (GWOVMP). The method reduces power consumption by 20.99 and resource wastage by 1.80 compared with existing algorithms.

Index Terms – Cloud Computing, GWO Algorithm, Metaheuristics Algorithm, Optimization, Virtual Machine Placement, Data Center, Power Consumption.

1. INTRODUCTION

Nowadays, Cloud Computing (CC) has developed rapidly and has become an indispensable technology in the field of information technology. This has led to the emergence of ever larger modern data centers with a high number of physical devices, which causes a remarkable increase in energy consumption and huge cooling expenses. In these data centers, three types of physical equipment consume electricity, including servers, cooling systems, and network equipment. To get a better idea of the benefits a user can get, CC offers four main deployment modes: public cloud, private cloud, hybrid cloud, and community cloud. The various CC resources are provided in three main services: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS).

In the public cloud model, the various resources are managed by the service provider that owns the cloud, and its resources are sold to the public according to its own needs. In this kind of cloud, part of the resources can be rented and managed by the end users. Examples of some of the best-known cloud providers are Amazon, Google, and Microsoft [1]. For the private cloud model, the ownership of this type would be an entity or an organization that wants to use certain applications that contain very sensitive information. If it is a community cloud, the provider is a set of organizations that come together for a common purpose [2]. In this case, several companies for example can decide to federate their efforts by building and managing a cloud together. Through virtualization, these servers are offered to customers in the form of VMs, which allows a high level of flexibility for the proper management of resources. These data centers and above all facilitates the execution of workloads in an elastic way; certain techniques such as hardware consolidation are therefore combined to maximize energy efficiency. In a DC (data center), the virtualization technique also allows different applications to be run on VMs or containers. This technology also allows efficient pooling of physical resources, such as CPU, RAM,

RESEARCH ARTICLE

and I/O interfaces, while allowing physical servers to host multiple VMs.

Although cloud computing has several advantages, the biggest challenges of cloud computing are the costs associated with energy consumption in data centers as well as carbon dioxide emissions. To overcome these difficulties, the advantages of virtualization are used to minimize the number of active PMs (Physical Machines) and to switch off inactive PMs. This requires the implementation of a strategy to better manage the placement of virtual machines (VMPs). To minimize the total power of a DC, it is therefore necessary to restrict the consumption of the PMs. In addition, there are other challenges such as security, which some studies have focused on [3]. Researchers have been interested in the fact that they can exploit the large storage and processing capacities that cloud computing offers [4]. VMP in the cloud environment remains a topic that attracts the attention of various researchers. The optimization of VMP in data centers can improve the efficiency of the energy quantity and the optimal use of resources while maximizing the quality of services (QoS) offered.

The optimization of VMP problems in a cloud environment is considered an NP-hard problem for which it is very rare to converge on an optimal solution. Solving such a problem can be expensive in terms of computation time when common methods such as graph theory are used. Therefore, it is preferable to use metaheuristic and heuristic techniques. Metaheuristics suffer from high execution time but provide optimal solutions. Heuristic algorithms have a moderate execution time, but not with a good quality of solutions [5]. Several algorithms and objectives have been implemented to solve the VMP problem, including the fuzzy logic algorithm [6].

In this paper, an algorithm called GWOVMP is adapted from the GWO method to optimize the VMP in a heterogeneous DC. To compare GWOVMP with existing algorithms, we perform similarities using the CloudSIM tool. The main contributions of this article are:

- Formulation of mathematical models of the problem of placing VMs in the cloud;
- A metaheuristic algorithm based on the grey wolf hierarchy called GWOVMP is proposed to solve this VMP problem in a cloud environment;
- We perform a simulation using CloudSIM and evaluate the proposed algorithm against those in the state of the art.

The remainder of this document is organized as follows. Section 2 presents a review of the literature and presents the various related works. Section 3 presents the problem formulation and the GWO for virtual machine placement. The experimental evaluation of the GWOVMP algorithm and the analysis of the results of our work are found in Section 4. Finally, Section 5 presents the conclusion and future work.

2. LITERATURE REVIEW

2.1. Virtual Machine Placement

Generally speaking, the placement problem is an old and classic one. It consists of filling boxes with different objects while using the minimum number of boxes. In the case of DCs, the placement of a virtual machine in the optimal assignment of VMs in CC consists of assigning VMs in a minimum number of PMs of a DC. This results in VMs being considered as objects and PMs as boxes.

The objective of optimal placement in the DC is to distribute loads across all physical servers while minimizing factors such as power consumption, increasing QoS, meeting service-level agreement (SLA), and reducing resource wastage. In a cloud-based DC, virtualization is one of the key technologies to rely on.

The benefits of virtualization, such as consolidation, migration, and load balancing of different VMs across MPs, increase DC efficiency and reduce operating costs. Figure 1 illustrates the simplified architecture of this virtualization technique.

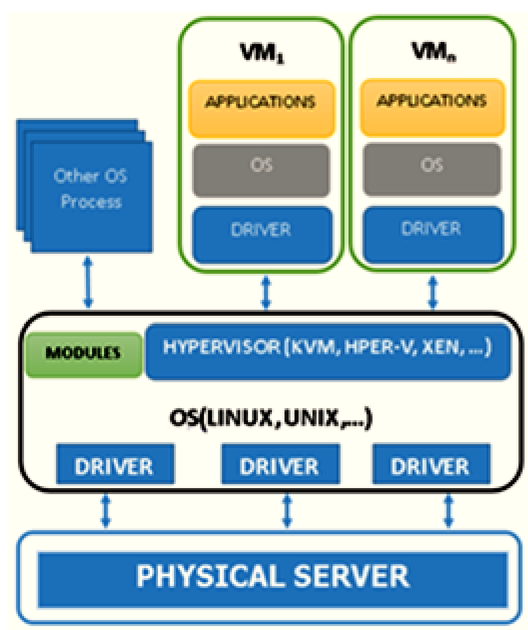

Figure 1 Virtualization Architecture

2.2. Grey Wolf Optimization

In the last two decades, metaheuristic optimization has become very popular. Some metaheuristic algorithms are well-known and used in different application areas [7]. Algorithms such as Ant Colony Optimization (ACO), Genetic Algorithm (GA), and Particle Swarm Optimization (PSO) have been used in various research including the VMP problem. The Grey Wolf Optimizer is one of the life-based metaheuristic algorithms that harkens back to the hunting techniques of the grey wolf and is inspired by the grey wolf's leadership chain. These grey wolves live in packs ranging from 5 to 12 on average and there are four types of these grey wolves such as α, β, δ, and ω. The first level is composed of the alphas which are male or female grey wolves that are stronger and guide the others in appropriate areas.

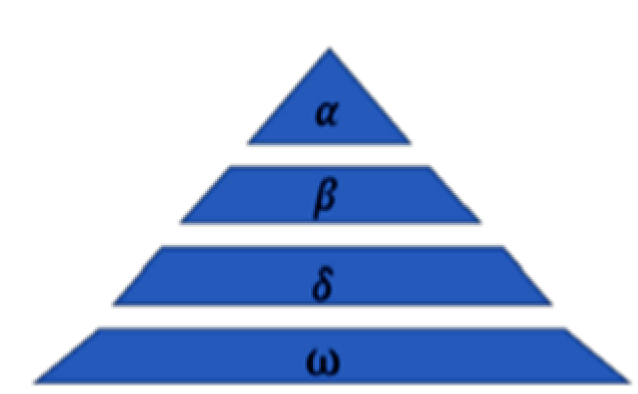

Figure 2 Social Hierarchy of Grey Wolves [8]

Figure 2 shows the social hierarchy of grey wolves. The equations (1) and (2) below present the mathematical modeling of this algorithm which is based on the encirclement of the prey by these grey wolves [9]:

$$\vec{D} = \vec{C}.\vec{X_p}(t) - \vec{X}(t) \qquad (1)$$

$$\vec{X}(t+1) = \vec{X_p}(t) - \vec{A}.\vec{D} \qquad (2)$$

In Equations (2) and (5), the current iteration is represented by t, the coefficients are represented in equations (3) and (4) by and $\vec{C}$, $\vec{X}$ is the vector that indicates the position of the grey wolf while the vector indicates the position of the prey. The following equations how the vectors $\vec{A}$, $\vec{C}$ and $\vec{a}$ shall be calculated[10]:

$$\vec{A} = 2\vec{a}.\vec{r_1} - \vec{a} \qquad (3)$$

$$\vec{C} = 2\vec{a}.\vec{r_2} \qquad (4)$$

$$\vec{a} = 2 - t\frac{2}{NumOfIt} \qquad (5)$$

The vectors $\vec{r}1$ and $\vec{r}2$ are random in the interval [0, 1], and during the exploration and exploitation phase, the vector decreases linearly from 2 to 0 during these iterations [11]. The best solution is represented first by the alphas, then the betas, and finally the deltas.

The omegas are considered undesirable solutions and are therefore not taken into account. The Figure 3[8] shows the different positions of the wolves during the hunt and can be defined mathematically by the equations below[12], [13]:

$$\vec{X}(t+1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} \qquad (6)$$

Where $\vec{X}_1 + \vec{X}_2 + \vec{X}_3$ are defined in Equations (7)-(9) below[13]:

$$\vec{X_1} = \vec{X_\alpha} - \vec{A_\alpha}.\vec{D_1} \qquad (7)$$

$$\vec{X_2} = \vec{X_\beta} - \vec{A_\beta}.\vec{D_2} \qquad (8)$$

$$\vec{X_3} = \vec{X_\delta} - \vec{A_\delta}.\vec{D_3} \qquad (9)$$

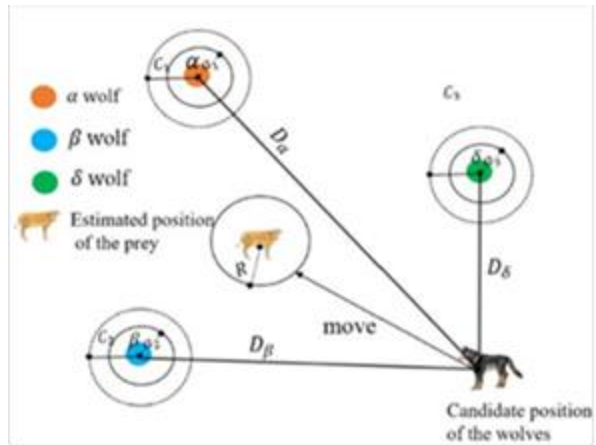

Figure 3 Update on the Position of Grey Wolves during the Hunt [14]

In this case, $\vec{X}_1 + \vec{X}_2 + \vec{X}_3$ are the best solutions to an iteration $\vec{A_\alpha}$, $\vec{A_\delta}$ which are defined in Equation (3), and are defined in Equations (10) -(12) as follow [15]:

$$\vec{D_\alpha} = |\vec{C_1}.\vec{X_\alpha} - \vec{X}| \qquad (10)$$

$$\vec{D_\beta} = |\vec{C_2}.\vec{X_\beta} - \vec{X}| \qquad (11)$$

$$\vec{D_\delta} = |\vec{C_3}.\vec{X_\delta} - \vec{X}| \qquad (12)$$

2.3. Related Works

In [16] they proposed an algorithm that performs VMP in a heterogeneous and multidimensional cloud environment in a random way. To optimize energy and resource utilization, they used GRVMP (Greedy Randomized Virtual Machine Placement) which is inspired by the two-choice power model and places VMs on the most energy-efficient PMs. They evaluated the performance of their algorithms using synthetic and real production scenarios with performance matrices. Their results show a significant reduction in active PMs as well as the total resource wastage of their algorithm compared to the methods of the previous study.

The authors have implemented smart techniques to deal with the VMP optimization problem [17]. Their objective is to be able to decrease the number of active physical servers as well as the energy consumption. They adapted two algorithms from the Grey Wolf Optimization model to a VMP problem. The proposed algorithms have been evaluated in a CloudSIM environment composed of homogeneous and heterogeneous servers. Compared to existing techniques, their algorithms minimize the number of active PMs used to host VMs.

For optimal placement in a Cloud Data Center (CDC), in [18] the authors proposed an algorithm based on the hybrid discrete whale and the multi-objective optimization algorithm. The multi-verse optimizer with chaotic functions is also used to optimize placement in a cloud environment. The main objective is to minimize the energy consumption that is consumed in the CDCs by reducing the active physical machines, as well as to reduce the waste of resources by using the optimal placement of VMs on the MPs. The applied method avoids the increase in the migration rate of VMs on PMs. The results they obtained show the performance of the proposed algorithm compared to other state-of-the-art algorithms.

In [19], [20], the authors provided a technique based on a particle swarm algorithm to improve task scheduling in a Cloud system. In the proposed technique, the choice of an appropriate objective function allowed them to perform dynamic VMP on the PM, balance the workload of VMs, reduce the time of all tasks while maximizing the utilization of all resources, and increase productivity. The proposed solution provides optimal results for task scheduling, proper allocation of tasks in VMs placement on the PM, and on the process of outsourcing VMs, the time was improved by 0.02.

In [20], the authors developed two algorithms for optimal resource management in the cloud. The first algorithm used unsupervised planning with the K-means algorithm and the second is the KNN algorithm for supervised learning. This also applies to the systematic analysis of VM and PM models, which allowed them to establish a rule for hybrid and dynamic deployment of cloud data center resources.

The authors of [5] proposed a technique to solve the problem related to the optimization of VMP in the cloud taking into account power and traffic. The proposed method minimizes CDC energy consumption, resource wastage, and network consumption. The method was compared with state-of-the-art algorithms. The results obtained show a 29% reduction in network consumption, an 18% reduction in overall energy consumption, and a 68% reduction in resource wastage. The proposed approach jointly minimizes energy consumption, optimal network utilization, and resource wastage in a heterogeneous and multidimensional cloud system. Their results were obtained after comparison with state-of-the-art fat tree technology.

In [21], a GWO algorithm was proposed based on the reliability capacity of resources for adaptive load balancing. The method they used first finds the inactive or occupied nodes and then calculates the threshold and suitability function for each node. They used CloudSim for simulation and their results improved the cost and response time of the proposed method lower than compared to other techniques.

The authors in [22] proposed an improved genetic algorithm (l-GA) based on the virtual hierarchy architecture model (VHAM) to solve the problem of VM placement. They used the finite element analysis (FEA) application to conduct the experiments. Their results show the efficient reduction of the number of MPs used to host VMs and thus reduce the power consumption in data centers while ensuring the availability of placement. I-GA performs better than other algorithms even in small instances.

In [23], the authors have formalized the VMP problem into an intelligent optimization while building on the different specifications of a data center. The objective of their work is to perform a good placement of VMs on PMs without affecting performance. With the experiments they conducted, the method used is a metaheuristic Particle Swarm Optimization Algorithm that gave efficiency on a significant decrease of energy of the data center and optimal balancing between different resources.

Table 1 provides a comparative study of some of the work carried out on VMP. It shows the different parameters used, the simulation environment (SE), and the different algorithm names implemented.

Table 1 Comparative Study of the State-of-the-Art

	Parameters									
work	Resource utilization				Resource wastage	SLA	Energy	Others	Algorithm	SE
	CPU	RAM	Storage	Bandwidth						
[15]	×	×			×		×		GRVMP	Amazon EC2

RESEARCH ARTICLE

[17]	×	×					×		GWO-VMP	CloudSIM
[18]	×	×	×	×	×	×	×			Amazon EC2
[19]	×	×			×		×			MATLAB
[20]	×				×		×		EHML	CloudSIM Amazon Cloud datacenter
[5]	×				×		×	×		Java
[21]					×		×	×		CloudSIM
[22]	×	×			×		×		I-GA	CloudSIM
[23]	×	×					×		RPMOTA	
This Work	×	×	×		×		×		GWOVMP	CloudSIM

3. PROBLEM FORMULATION AND GWO FOR VIRTUAL MACHINE PLACEMENT (GWOVMP)

3.1. Problem Formulation

In this article, the basic objective is to implement an optimal placement scheme for virtual machines. The main goal of the VMP is to be able to modify the original allocation of virtual machines to physical nodes to minimize energy costs.

In this work, the goal is to minimize this energy consumption in cloud computing according to the available resources. Let us note the matching function f. This virtual machine placement problem can be denoted as f: V→P [5]. The minimum number of active servers for an optimal VMP solution can be formulated as shown in equation (13):

$$f(S) = \min \sum_{i=1}^{n} Y_i \qquad (13)$$

In Cloud Computing, DCs are composed of heterogeneous physical machines. In this paper, a number m of PMs and a number n of VMs are considered. P denotes the PM set where i is the i^{th} PM, i belongs to $[1, \dots, m]$, and $P_i \in P$. Similarly, V denotes the set of heterogeneous VMs, where V_j denotes the j^{th} VM, j belongs to $[1, \dots, n]$, and $V_j \in V$.

To represent the capacity of i^{th}PM represented by the P_i r-dimension vector $P_i = \{P_i^1, P_i^2, \dots, P_i^r\}$, which P_i^k represents the capacity of the resource k^{th} of i^{th} PM, $\forall k = \{1,2, \dots, r\}$.

Represent the capacity of i^{th} VM represented by the V_j r-dimensional vector $V_j = \{R_i^1, R_i^2, \dots, R_i^r\}$, where R_i^krepresents the capacity of the resource k^{th} of J^{th} VM, $\forall k = \{1,2, \dots, r\}$. To express the relationship between VM and PM, this function is represented by the binary matrix $X_{m\times n}$ as shown in the equation (14):

$$x_{ij} = \begin{cases} 1 & \text{if } V_j \text{ is assigned to } P_i \\ 0 & \text{Otherwise } , \forall i \in P et \; \forall j \in V \end{cases} \qquad (14)$$

To designate the state of the PM, the matrix y_n, and P_i are used and is active if $y_i = 1$ and inactive $y_i = 0$. This is shown in equation (15):

$$y_{ij} = \begin{cases} 1 & \text{if } \sum_{i=1}^{n} P_i \geq 1 \;\; \forall i \in P_i \\ 0 & \text{Otherwise} \end{cases} \qquad (15)$$

In this work, processor, memory, and storage resources are considered. For the capabilities of computing power are represented by P^{cpu} and that of memory by P^{ram} . Similarly, the processor and memory are represented by V^{cpu} and V^{ram} which leads us to the following constraints [24]:

$$\sum_{i}^{M} V^{cpu} \,.\, x_{ij} \leq P^{cpu} . y_i \;\; \forall i \in P \qquad (16)$$

$$\sum_{i}^{N} V^{ram} \,.\, x_{ij} \leq P^{ram} . y_i \;\; \forall i \in P \qquad (17)$$

In equations (16) and (17) we specify that the CPU and memory should not exceed, which specifies the capacity constraints. Equation (18), states that the total storage space

RESEARCH ARTICLE

requirements for VMs on PMs must not exceed the storage space of the PM it is hosting.

$$\sum_i^N V^{sto} \,.\, x_{ij} \leq P^{sto} . y_i \ \forall i \in P \qquad (18)$$

Several works show that the energy consumption of DCs in CC is linear with a linear load [24], [25], [26], [27] . Shutting down idle servers will reduce the cost of energy consumption. Energy consumption in the cloud is represented by equation (19):

$$P^{cpu} = \begin{cases} P_{iddle} + \left(P_{full} - P_{iddle}\right) \times U_i^{cpu} & if\ U_i^{cpu} > 0 \\ 0, & otherwise \end{cases} \qquad (19)$$

Where P_{iddle} is the power of energy consumption by the PM when it is in an idle state, P_{full} is the power that is consumed in a fully utilized CPU, and ${P^{cpu}}_i \in [0, 1]$. The equation (20) shows that U^{cpu} is the processor (CPU) rate of the PM P_i as a function of the VM demand V_i.

$$U_i^{cpu} = \frac{\sum_{j=1}^{n} x_{ij} \times R_i^{cpu}}{P_i^{cpu}} \qquad (20)$$

To calculate the waste of different resources of a PM of dimension r, we generalize the equation (21) so that it is applied to resources [28]:

$$W_i = \frac{|R_i^{cpu} - R_i^{ram}|}{|P_i^{cpu} + P_i^{ram}|} + \varepsilon \qquad (21)$$

Where W_i represents the waste of different PM resources. R^{cpu} and R^{ram} represent the normalized processor and memory resources of the physical machine, respectively; the values of P^{cpu} and P^{ram} describe the normalized use of the machine. ϵ is the small real number equal to 0.0001. This model aims to best include the resources of all PMs and allows for the balance between multiple resources to be achieved. The mathematical equation (22) indicates the total waste of resources W^{tot} of all PMs in a CDC [29]:

$$W^{tot}{}_i = \sum_{i=1}^{m} y_i \times \frac{\sum_{k=1}^{r} |R_i^{cpu} - R_i^{ram}|}{\sum_{k=1}^{r} P_k^r} + \varepsilon \qquad (22)$$

3.2. GWO for Virtual Machine Placement

The effective reduction of power consumption in a cloud-based data center is usually achieved by dramatically reducing the number of active physical machines. In this case, an algorithm adapted from the GWO algorithm is used to solve the VMP problem to obtain an optimal solution that allocates VMs to a minimum number of active physical servers.

The best solution that has a minimum number of active PMs is chosen as S. In the initial state, the optimal placement of the VMs in the MPs is not known and the best solution that has been generated, designated SI, is considered first. Indeed, to match the m VMs to the n PMs, a random distribution of the m VMs to the n PMs and each VM is subjected to a single PM. There is therefore m^n possible solutions.

To search for prey, wolves continuously update their positions. After the solution step which is randomly distributed for each wolf that joins the pack. GWOVMP generates a new solution while updating the existing solution of each wolf to find the most optimal distribution of VMs on the MP. For each iteration t, the wolf positions are updated based on the best solutions generated by α, β, and δ.

To minimize the number of active physical machines and to minimize power consumption in a cloud-based system, the solution must be improved to reduce the number of active running VMs under all conditions. In general, the GWO algorithm updates the path of the best solution set based on equation (6), where one by one some VM locations are updated. It can be seen that some locations are updated outside the [1, ..., n] boundary. In this case, the algorithm relies on the binary model to recalculate the allocation of these Virtual Machines while ensuring that the physical servers are within the limit:

$$W_{ij}(t+1) = \begin{cases} 1 & if\ sigmoid\ (\frac{\vec{x}_1 + \vec{x}_2 + \vec{x}_3}{3}) \\ 0, & otherwise \end{cases} \qquad (23)$$

In general, without taking into account the case treated in this paper, we have in Equation (23) the rand which represents a random number extracted from the uniform distribution that ∈ [0, 1]. x_{ij} represents the binary position that is updated at iteration t and in the determined dimension, the sigmoid function is determined by the equation (24):

$$sigmoid\ (a) = \frac{1}{1 + e^{-10(x-0.5)}} \qquad (24)$$

Input: VM, PM

Output: Placement of VMs on PMs

Step 1: Initialization of a population of *n* grey wolves positions randomly. Set parameters a via Equation (4) and (5). *n* is defined as the number of wolves considered as a search factor. Determine the total number of iterations. Set *it= 1* as the initial iteration.

Step 2: Let *n* Wolves build and save the top tree solutions α, β and δ.

Step 3: Update the solutions of α, β and δ with Equations (16) and (17).

Step 4: Calculate the different fitness values for all solutions and identify the tree best solution for the current iteration

Step 5: Terminate the algorithm. Check the current number of iterations; the algorithm terminates if the number reached exceeds the maximum number of iterations. Otherwise, increment and return to step (3).

International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2024/224431 Volume 11, Issue 1, January – February (2024)

Step 6: Return α.

Algorithm 1 GWOVMP

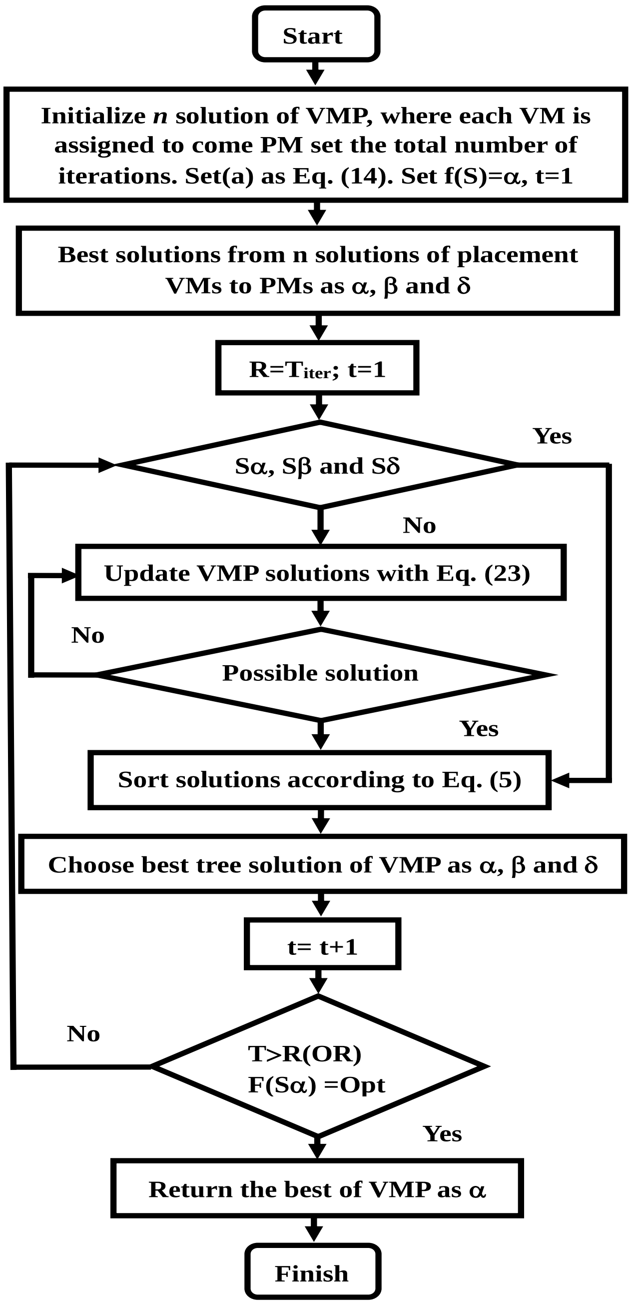

Figure 4 GWOVMP Flowchart

Algorithm 1 describes the GWOVMP based on an adjustment of the various basic operations of the Gray Wolf optimization algorithm, and the corresponding flowchart is shown in Figure 4.

In case a VM has not been allocated during an update, deletion, or duplicate overload, an operation is used to reallocate it. The constraint of the VMs is evaluated by Equations (16) and (17) after an update of the wolf positions. In case the value of x_{ij} corresponding to V_j has not been allocated to any PM during the update, the VM is reallocated to a non-overloaded PM with sufficient resources to meet the demand.

As mentioned, GWO is based on a master strategy in which wolves update their positions each time to attack the prey based on the positions of the top three wolves. In addition, the CDC's energy consumption needs to be considerably reduced by limiting the number of active physical servers. This can be achieved by improving the solution on active servers while reducing their number.

In this case, it can be seen that the exploration phase carried out by the wolves can sometimes be done in the opposite direction involving the binary search space. It is on the generalities of this algorithm that if each bit of the values α, β and δ goes to one, the value that corresponds to xij to be submitted to P_i is updated.

To balance and maximize the use of memory and CPU resources, VMs are reallocated to PMs that give the smallest absolute dissimilarity that is estimated between the use of CPU and memory resources after the virtual machine is used. The difference between the different CPU and memory resource usage after adding a virtual machine to the PM is calculated after adding the different resource requirements of the virtual machine to the resources used by the physical server.

Table 2 lists the various symbols used in this study. Each symbol has a particular meaning and has been carefully selected to contribute to the clarity and understanding of the various equations. This provides a handy visual reference for readers to quickly interpret the information contained in the work.

Table 2 Symbols and Descriptions

Symbol	Descriptions
P	Set of PMs
V	Set of VMs
P^{cpu}	CPU capacities of the PM
P^{ram}	CPU capacities of the PM
V^{cpu}	CPU capacities of the VM
V^{ram}	CPU capacities of the VM
P^{sto}	Storage capacities of the PM
V^{sto}	Storage capacities of the VM
P_{iddle}	Power that is consumed in an idle state

RESEARCH ARTICLE

P_{full}	Power that is consumed in an idle state
U_i^{cpu}	Normalized CPU utilization
R_i^{cpu}	Normalized CPU resources of the PM
R_i^{ram}	Normalized RAM resources of the PM
W_{tot}	Total waste of resources of all PMs in a CDC
x_{ij}	Assigning V to P or not
y_{ij}	Status of the machine (Active if y=1, Otherwise)

4. PERFORMANCE EVALUATION AND DISCUSSION

In this part of the paper, experimental tests were carried out to determine the performance of the proposed algorithm, called GWOVMP. This algorithm was implemented in Java like all the other algorithms compared. The simulator that was used in this work is CloudSIM and the toolkit version 5.0 of the CloudSIM simulator was used. It supports many IaaS featuressuch as on-demand resource provisioning and power consumption solutions [30], [31]. It also allows us to evaluate the performance of new applications and cloud strategies before they are developed in the real environment. The various tests and experiments were carried out on a computer equipped with an AMD Ryzen 5800U processor with 4.4 GHz and 16GB of RAM. The operating system used was Windows 11.

To evaluate the performance of this method, it is compared with algorithms such as First fit decreasing FFD[32], Modified best fit decreasing (MBFD)[33], Random first fit (RFF)[34] and Best Fit Decreasing (BFD)[35]. To test the effectiveness of the GWOVMP algorithm, instances of various sizes ranging from 100 to 4000 were created. Each VM is equipped with a 16-core processor and 32 GB RAM. A CPU of 1 to 4 cores is required for each VM, and RAM of 1 to 8 GB is randomly generated. In cases where the CPU is one of the most important resources, the ratio between the different memory and CPU requirements is almost 10:9.

The associated parameters with the GWOVMP algorithm are a = 2, which decreases linearly with each iteration, and the random vectors r1 and r2 are included in [0,1]. This gives this algorithm the advantage of defining a few parameters. In the case of the other algorithms, the parameters are set according to the basic literature. It is taken into account the 100% achievability of the resource usage and its implementation uses 100 iterations which stops early on the fifth iteration.

4.1. Resource Utilisation Performance

Table 3 shows the results obtained after comparing some existing algorithms with the proposed model. In this case, 100 to 2000 VMs are used to see the evolution of the results and to be able to conclude if a small number of evolving is applied to all the methods used for comparison. In the state of the art, FFD produces the worst result of all algorithms for all instances. GWOVMP performs well in many cases, especially as the number of active VMs increases. The BFD algorithm is the second-best performer. The RFF algorithm gives the worst results in all cases, which also shows that this algorithm does not fit a heterogeneous environment

Table 3 Number of Active Physical Machines

No	VM	BFD	RFF	MBFD	FFD	GWOVMP
1	100	16	16	16	18	16
2	200	32	33	33	37	32
3	300	48	46	46.03	53	48
4	400	63	64	64.4	74	63
5	500	78	77	77.63	91	78
6	600	94	95	95.77	111	99
7	1000	158	159	161.23	184	153.26
8	2000	316	317	319.64	366	278.08

The GWOVMP algorithm, which is the proposed and implemented solution, generates better results than all other solutions, except in the case of using a small number of VMs. As the problem size increases, the GWO result gives a satisfactory result.

Figure 5 shows the active number of physical servers that are used to support the different sets of virtual machines for the different algorithms used. The optimal PM solution used for the GWOVMP case is observed when a significant number of VMs are used.

RESEARCH ARTICLE

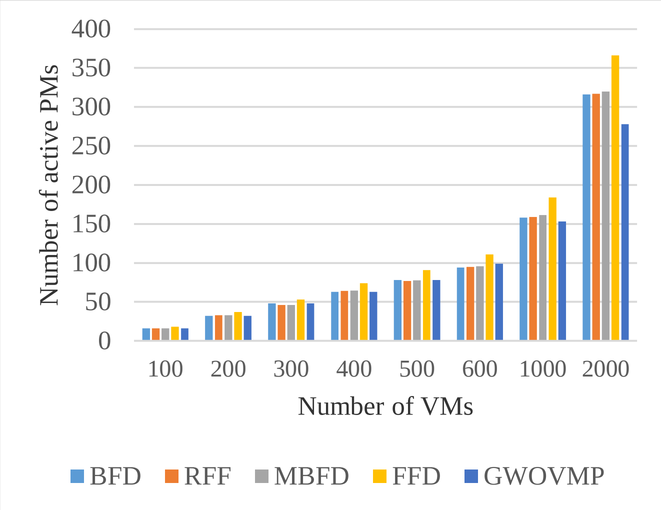

Figure 5 Number of Active PMs

Referring to Figure 6, the proposed GWOVMP model is the most efficient in terms of CPU utilization. This model provides optimal CPU utilization for the different algorithms, which becomes very noticeable when using a large number of virtual machines. Although the FFD algorithm is the least efficient, it is worth noting that it gives an optimal solution when allocating a very small number of VMs on PMs, compared with the BFD, RFF, and MBFD algorithms.

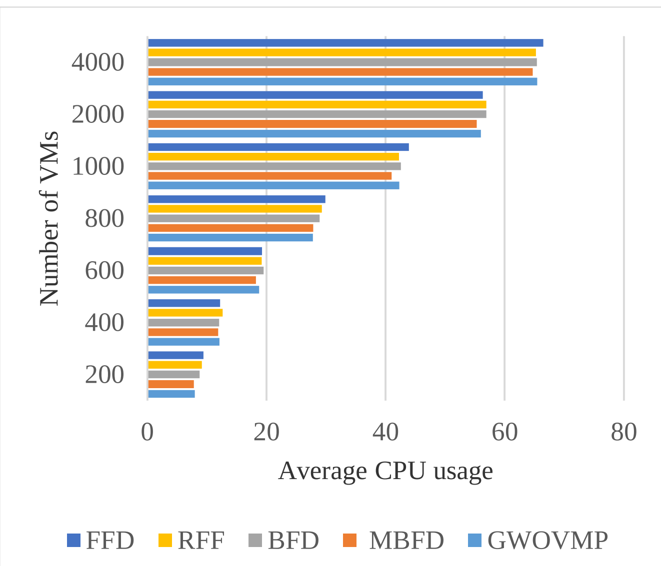

Figure 6 Average CPU Utilization

Figure 7 shows the average RAM usage of active servers in the allocation obtained by BFD, FFD, RFF, MBFD, and GWOVMP for the different tests. FFD and RFF achieved the highest memory usage. For FFD, low memory utilization is achieved by using a very small number of VMs up to 1000 VMs. MBFD gives better optimal solutions for small to medium numbers of VMs. The difference in this algorithm is observed when the number of VMs increases considerably. In this case, it becomes the second algorithm to offer a more optimal solution. In the case of the proposed GWOVMP algorithm, memory usage remains within reasonable limits in all possible cases.

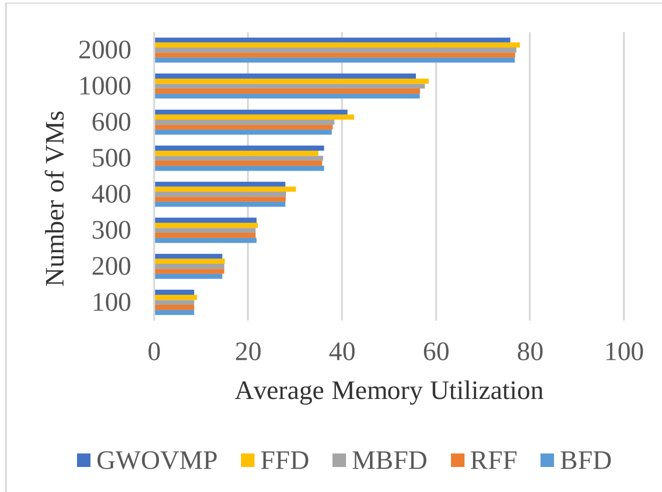

Figure 7 Average RAM Utilization

Figure 8 shows that the FFD method consistently achieves exceptionally high storage utilization in multiple scenarios, closely followed by the MBFD algorithm. However, it is GWOVMP, the proposed solution that stands out, particularly when faced with a substantial increase in the number of active virtual machines.

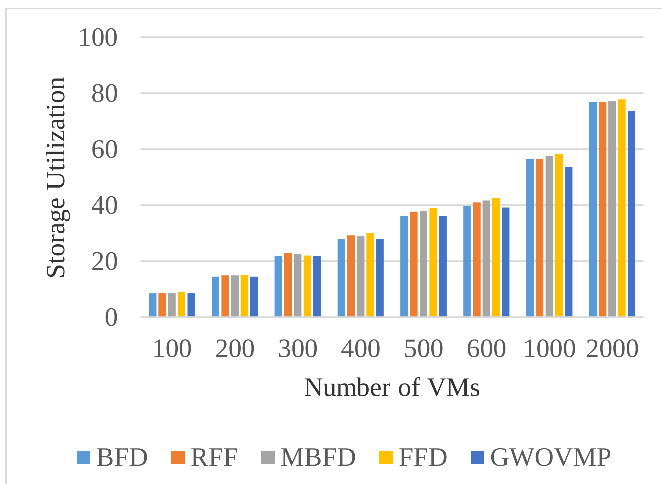

Figure 8 Storage Utilization

Remarkably, GWOVMP boasts an average improvement rate of 1.8% over all the methods studied. This underlines its efficiency and superiority in optimizing storage utilization, even in the face of increased operational demands. These results not only highlight the shortcomings of state-of-the-art

RESEARCH ARTICLE

methods such as FFD but also underscore GWOVMP's potential as a transformative solution to storage efficiency challenges in the cloud.

4.2. Resources Wastages

Figure 9 reveals the results on wasted resource utilization rates in a data center. We can see that the FFD algorithm consistently shows a high rate of resource wastage in multiple scenarios, closely followed by the MBFD algorithm.

However, it is the GWOVMP algorithm that stands out, particularly when confronted with a substantial increase in the number of active virtual machines. GWOVMP achieves an average improvement rate of 1.6 over all the methods studied. These results underline its effectiveness in mitigating resource wastage, making it a better solution than the methods compared.

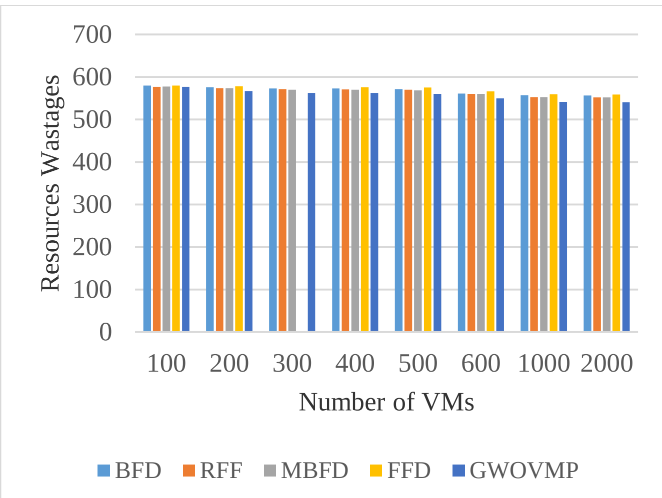

Figure 9 Resources Wastages

4.3. Energy Consummation Performance

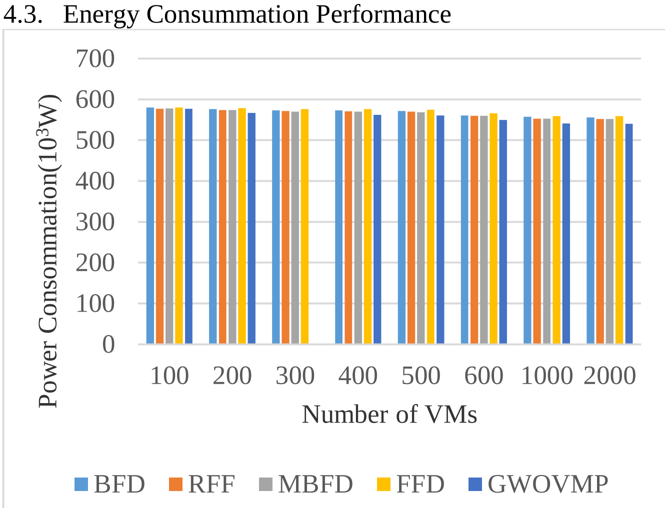

Figure 10 Power Consummation

This part of the article, reveals key results on the energy consumption of various algorithms that have been used to optimize the cloud data center environment.

Figure 10 shows a notable trend in which the energy consumption of the BFD, RFF, MBFD, FFD, and GWOVMP algorithms increases in direct correlation with the growing number of active virtual machines. Intriguingly, GWOVMP stands out from these leading methods by demonstrating a remarkable 20.99% reduction in energy consumption within the cloud infrastructure.

They highlight the maximum energy consumption, which peaks at 6580 kWh for BFD, and the minimum energy consumption, notably achieved by RFF with 6588 kWh. MBFD and FFD record energy consumptions of 65983 kWh and 6899 kWh, respectively, for an equivalent number of active user and PM requests. These statistics provide valuable quantitative evidence of GWOVMP's efficiency compared to its peers.

Collectively, these results not only highlight the escalating demand for energy in cloud computing but also underscore GWOVMP's potential as a more sustainable, energy-efficient solution for cloud infrastructures.

4.4. Discussion and Limitations

The choice of CloudSim as the simulation environment was dictated by its performance, as observed in state-of-the-art work. It enabled us to automate the simulation life cycle while reducing processing time. In this simulation environment, the proposed model was configured in a library according to the proposed configuration parameters.

The results obtained in this study allow us to state that the GWOVMP algorithm is better than the other algorithms for solving the VMP optimization problem. Like most of the work that has been used to compare the proposed algorithm, the work carried out does not take into account the network and certain evaluation metrics (e.g. migration, QoS, risk-aware resource allocation). To remedy this, this method needs to be improved by involving many more evaluation parameters. This work focused on the problem of VM placement to reduce energy, CPU, RAM, and storage usage. The other metric taken into account is resource wastage. Despite GWOVMP's performance, it needs to be improved to give much better results on storage utilization and resource wastage, which gives a low rate of improvement. To solve this problem, we'll have to improve this method while adding certain parameters, such as network bandwidth. Virtual machine migration will also need to be taken into account to strengthen the model.

But also, it would be preferable to implement and compare the method that has been proposed in this work in other

RESEARCH ARTICLE

simulators used in some research works and compare the results.

5. CONCLUSION

In a cloud data center, the physical machines as well as the cooling system consume a lot of energy. The significant reduction in the number of physical machines turned on will impact significantly the energy consumption in these cloud-based environments. The optimal placement of virtual machines in a cloud computing environment is considered one of the most difficult tasks. A lot of research has been conducted to solve this type of optimization problem. Metaheuristics have become a very important solution for this type of difficult optimization problem. The GWO-inspired algorithm that has been proposed in this work is one of the nature-inspired methods. In this paper, we have implemented a metaheuristic approach called GWOVMP to provide an optimal or near-optimal solution to the VMP problem in a cloud environment. The objective of GWOVMP is to be able to decrease the large power consumption in a cloud environment while minimizing the active MPs and minimizing the waste of CPU and memory resources. The results of this work show that the proposed technique gives the optimal use of PM resources by placing the VM on a suitable PM while reducing the power consumption compared to other algorithms.

In future work, we will try to use the GWOVMP algorithm to achieve other VM placement goals in a cloud environment. It will be compared with other algorithms such as SSA (Salp Swarm Algorithm), and PSO (particle swarm optimization). Furthermore, these algorithms will be coupled with different machine-learning mechanisms and applied to the case of containers in cloud computing.

REFERENCES

[1] S. Abohamama and E. Hamouda, "A hybrid energy–Aware virtual machine placement algorithm for cloud environments," Expert Syst Appl, vol. 150, p. 113306, 2020, doi: 10.1016/j.eswa.2020.113306.

[2] S. P. M. Ziyath and S. Senthilkumar, "MHO: meta heuristic optimization applied task scheduling with load balancing technique for cloud infrastructure services," J Ambient Intell Humaniz Comput, vol. 12, no. 6, pp. 6629–6638, Jun. 2021, doi: 10.1007/s12652-020-02282-7.

[3] A. Ndayikengurukiye, A. Ez-Zahout, and F. Omary, "A Big Data-Driven Model for Secured Systems: A Blockchain-Based Architecture for Cloud Computing and IoT Security," in Advances on Smart and Soft Computing, F. Saeed, T. Al-Hadhrami, E. Mohammed, and M. Al-Sarem, Eds., Singapore: Springer Singapore, 2022, pp. 409–418.

[4] A. Ndayikengurukiye, A. Ez-Zahout, A. Aboubakr, Y. Charkaoui, and O. Fouzia, "Resource Optimisation in Cloud Computing: Comparative Study of Algorithms Applied to Recommendations in a Big Data Analysis Architecture," Journal of Automation, Mobile Robotics and Intelligent Systems, vol. 2021, no. 4, pp. 65–75, 2021, doi: 10.14313/JAMRIS/4-2021/28.

[5] S. Omer, S. Azizi, M. Shojafar, and R. Tafazolli, "A priority, power and traffic-aware virtual machine placement of IoT applications in cloud data centers," Journal of Systems Architecture, vol. 115, May 2021, doi: 10.1016/j.sysarc.2021.101996.

[6] U. Chourasia, R. Gandhi, P. Vishwavidyalaya, and D. Sanjay, "Optimal Load Balancing in Cloud using ANFIS and MGWO based Polynomial Neural Network", doi: 10.21203/rs.3.rs-529618/v1.

[7] M. S. Kumar and G. R. Karri, "EEOA: Cost and Energy Efficient Task Scheduling in a Cloud-Fog Framework," Sensors, vol. 23, no. 5, Mar. 2023, doi: 10.3390/s23052445.

[8] Y. Li, X. Lin, and J. Liu, "An improved gray wolf optimization algorithm to solve engineering problems," Sustainability (Switzerland), vol. 13, no. 6, Mar. 2021, doi: 10.3390/su13063208.

[9] J. Too, A. R. Abdullah, N. M. Saad, N. M. Ali, and W. Tee, "A new competitive binary grey wolf optimizer to solve the feature selection problem in EMG signals classification," Computers, vol. 7, no. 4, Dec. 2018, doi: 10.3390/computers7040058.

[10] E. G. Dada, S. B. Joseph, D. O. Oyewola, A. A. Fadele, H. Chiroma, and S. M. Abdulhamid, "Application of Grey Wolf Optimization Algorithm: Recent Trends, Issues, and Possible Horizons," Gazi University Journal of Science, vol. 35, no. 2, pp. 485–504, Jun. 2022, doi: 10.35378/gujs.820885.

[11] G. Shial, S. Sahoo, and S. Panigrahi, "An Enhanced GWO Algorithm with Improved Explorative Search Capability for Global Optimization and Data Clustering," Applied Artificial Intelligence, vol. 37, no. 1, 2023, doi: 10.1080/08839514.2023.2166232.

[12] N. M. Sallam, A. I. Saleh, H. Arafat Ali, and M. M. Abdelsalam, "An Efficient Strategy for Blood Diseases Detection Based on Grey Wolf Optimization as Feature Selection and Machine Learning Techniques," Applied Sciences (Switzerland), vol. 12, no. 21, Nov. 2022, doi: 10.3390/app122110760.

[13] Y. Hou, H. Gao, Z. Wang, and C. Du, "Improved Grey Wolf Optimization Algorithm and Application," Sensors, vol. 22, no. 10, May 2022, doi: 10.3390/s22103810.

[14] H. M. R. Al-Khafaji, "Improving Quality Indicators of the Cloud-Based IoT Networks Using an Improved Form of Seagull Optimization Algorithm," Future Internet, vol. 14, no. 10, Oct. 2022, doi: 10.3390/fi14100281.

[15] P. Hu, J. S. Pan, and S. C. Chu, "Improved Binary Grey Wolf Optimizer and Its application for feature selection," Knowl Based Syst, vol. 195, May 2020, doi: 10.1016/j.knosys.2020.105746.

[16] S. Azizi, M. Shojafar, J. Abawajy, and R. Buyya, "GRVMP: A Greedy Randomized Algorithm for Virtual Machine Placement in Cloud Data Centers," IEEE Syst J, vol. 15, no. 2, pp. 2571–2582, Jun. 2021, doi: 10.1109/JSYST.2020.3002721.

[17] A. Al-Moalmi, J. Luo, A. Salah, and K. Li, "Optimal virtual machine placement based on grey wolf optimization," Electronics (Switzerland), vol. 8, no. 3, Mar. 2019, doi: 10.3390/electronics8030283.

[18] S. Gharehpasha, M. Masdari, and A. Jafarian, "Virtual machine placement in cloud data centers using a hybrid multi-verse optimization algorithm," Artif Intell Rev, vol. 54, no. 3, pp. 2221–2257, Mar. 2021, doi: 10.1007/s10462-020-09903-9.

[19] A. Yousefipour, A. M. Rahmani, and M. Jahanshahi, "Improving the load balancing and dynamic placement of virtual machines in cloud computing using particle swarm optimization algorithm," International Journal of Engineering Transactions C: Aspects, vol. 34, no. 6, pp. 1419–1429, Jun. 2021, doi: 10.5829/ije.2021.34.06c.05.

[20] B. Liang, D. Wu, P. Wu, and Y. Su, "An energy-aware resource deployment algorithm for cloud data centers based on dynamic hybrid machine learning," Knowl Based Syst, vol. 222, Jun. 2021, doi: 10.1016/j.knosys.2021.107020.

[21] S. S. Sefati, M. Mousavinasab, and R. Zareh Farkhady, "Load balancing in cloud computing environment using the Grey wolf optimization algorithm based on the reliability: performance evaluation," Journal of Supercomputing, vol. 78, no. 1, pp. 18–42, Jan. 2022, doi: 10.1007/s11227-021-03810-8.

[22] J. Lu, W. Zhao, H. Zhu, J. Li, Z. Cheng, and G. Xiao, "Optimal machine placement based on improved genetic algorithm in cloud

RESEARCH ARTICLE

computing," Journal of Supercomputing, vol. 78, no. 3, pp. 3448–3476, Feb. 2022, doi: 10.1007/s11227-021-03953-8.
[23] J. Luo, W. Song, and L. Yin, "Reliable Virtual Machine Placement Based on Multi-Objective Optimization with Traffic-Aware Algorithm in Industrial Cloud," IEEE Access, vol. 6, pp. 23043–23052, Mar. 2018, doi: 10.1109/ACCESS.2018.2816983.
[24] A. Ndayikengurukiye, A. Ez-Zahout, and F. Omary, "An overview of the different methods for optimizing the virtual resources placement in the Cloud Computing," J Phys Conf Ser, vol. 1743, p. 012030, Jan. 2021, doi: 10.1088/1742-6596/1743/1/012030.
[25] H. Li, L. Wen, Y. Liu, and Y. Shen, "More than Meets One Core: An Energy-Aware Cost Optimization in Dynamic Multi-Core Processor Server Consolidation for Cloud Data Center," Electronics (Switzerland), vol. 11, no. 20, Oct. 2022, doi: 10.3390/electronics11203377.
[26] J. Wang, H. Gu, J. Yu, Y. Song, X. He, and Y. Song, "Research on virtual machine consolidation strategy based on combined prediction and energy-aware in cloud computing platform," Journal of Cloud Computing, vol. 11, no. 1, Dec. 2022, doi: 10.1186/s13677-022-00309-2.
[27] S. Nabavi, L. Wen, S. S. Gill, and M. Xu, "Seagull optimization algorithm based multi-objective VM placement in edge-cloud data centers," Internet of Things and Cyber-Physical Systems, vol. 3, pp. 28–36, 2023, doi: 10.1016/j.iotcps.2023.01.002.
[28] H. Talebian et al., "Optimizing virtual machine placement in IaaS data centers: taxonomy, review and open issues," Cluster Comput, vol. 23, no. 2, pp. 837–878, Jun. 2020, doi: 10.1007/s10586-019-02954-w.
[29] M. Ghetas, "A multi-objective Monarch Butterfly Algorithm for virtual machine placement in cloud computing," Neural Comput Appl, vol. 33, no. 17, pp. 11011–11025, Sep. 2021, doi: 10.1007/s00521-020-05559-2.
[30] T. B. Hewage, S. Ilager, M. A. Rodriguez, and R. Buyya, "CloudSim express: A novel framework for rapid low code simulation of cloud computing environments," Softw Pract Exp, 2023, doi: 10.1002/spe.3290.
[31] I. Behera and S. Sobhanayak, "Task scheduling optimization in heterogeneous cloud computing environments: A hybrid GA-GWO approach," J Parallel Distrib Comput, vol. 183, Jan. 2024, doi: 10.1016/j.jpdc.2023.104766.
[32] M. A. Khan, A. Paplinski, A. M. Khan, M. Murshed, and R. Buyya, "Dynamic virtual machine consolidation algorithms for energy-efficient cloud resource management: A review," in Sustainable Cloud and Energy Services: Principles and Practice, Springer International Publishing, 2017, pp. 135–165. doi: 10.1007/978-3-319-62238-5_6.
[33] A. Beloglazov, J. Abawajy, and R. Buyya, "Energy-aware resource allocation heuristics for efficient management of data centers for Cloud computing," Future Generation Computer Systems, vol. 28, no. 5, pp. 755–768, May 2012, doi: 10.1016/j.future.2011.04.017.
[34] M. Masdari, S. S. Nabavi, and V. Ahmadi, "An overview of virtual machine placement schemes in cloud computing," Journal of Network and Computer Applications, vol. 66. Academic Press, pp. 106–127, May 01, 2016. doi: 10.1016/j.jnca.2016.01.011.
[35] D. Zhu, "Max–min Bin Packing Algorithm and its application in nano-particles filling," Chaos Solitons Fractals, vol. 89, pp. 83–90, Aug. 2016, doi: 10.1016/j.chaos.2015.09.024.

Authors

Aristide Ndayikengurukiye graduated with an Msc in Computer Engineering from the University of Burundi in 2017. PhD Student in Intelligent Processing Systems & Security Team (IPSS), Computer Science Department Faculty of Science of Rabat, Mohammed V University, Morocco. His research interests include Cloud computing optimization, artificial intelligence, IoT, and Big Data. He can be contacted at e-mail: lemure.dede@gmail.com.

Abderrahmane Ez-Zahout is currently an assistant professor of computer science at the Department of Computer Science/faculty of Sciences at Mohammed V University. His research is in the fields of computer sciences, digital systems, big data, and computer vision. Recently, he has worked on intelligent systems. He has served as an invited reviewer for many journals. Besides, he is also involved in NGOs, and student associations, and managing non-profit foundations. He can be contacted at email: abderrahmane.ezzahout@um5.ac.ma.

Fouzia Omary is the leader of IPSS team research and is currently a full professor of computer science at the Department of Computer Science/ENS, Mohammed V University. She is a full-time researcher at Mohammed V University and she is the chair of the international conference on cybersecurity and blockchain and also the leader of the national network of blockchain and cryptocurrency. She can be contacted at email: omary@fsr.ac.ma.

How to cite this article:

Aristide Ndayikengurukiye, Abderrahmane Ez-Zahout, Fouzia Omary, "Optimizing Virtual Machines Placement in a Heterogeneous Cloud Data Center System", International Journal of Computer Networks and Applications (IJCNA), 11(1), PP: 1-12, 2024, DOI: 10.22247/ijcna/2024/224431.