International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

# A Novel of Calculating the Optimal Number of Virtual Servers to Improve Resources Usage in Cloud Computing

**Abdualmajed A. G. Al-Khulaidi*[1], Ahmed A. Al-Shalabi [2], Adel A. Nasser [3], Mohammed Sarhan Al-Duais [4], Mansoor N. Ali [5], Nesmah A. AL-Khulaidi [6]**

**Submitted**: 02/12/2023 **Revised**: 15/01/2024 **Accepted**: 25/01/2024

**Abstract:** Recently, there has been a major revolution in cloud computing, where most companies have become dependent on cloud computing because of its importance in saving resources and reducing expenses. The research dealt with the problems of optimal allocation of resources within the cloud environment. Experiments have shown that the optimal number of virtual servers to be placed on the physical server to be presented to the user are the advantages of using algorithms to place virtual machines on computing servers online to save money, make maximum use of resources, and avoid system overload. As it was also observed in the case of offline algorithms for placing virtual machines on the actual server, it was concluded that the NFD algorithm is the best in the case of problem size 300 * 300, number of nodes L = 6 nodes, time T = 0.55, and algorithm accuracy R = 1.700. The optimal number of virtual servers to be placed on the physical server is four nodes. This indicates that online algorithms for placing virtual machines on the physical server work better in cloud computing. Research has demonstrated that the advantages of having the ideal number of virtual servers installed on the physical server that the user sees.

***Keywords****: Cloud Computing, Optimal number, Resources*

## 1. Introduction

Cloud computing is considered to be one of the most important technologies that companies rely on, regardless of their size. Therefore, cloud computing is an urgent requirement. Thus, the development and allocation of resources in cloud computing are considered important [1]. When discussing this, we must look for problems and find appropriate solutions. There are many problems within the cloud; however, the number of nodes required to perform tasks in cloud virtualization was chosen based on the resource consumption. The problem of optimal resource allocation to these nodes is of great importance, and the dynamic allocation of resources is important within the cloud environment because it plays a role in scheduling incoming tasks on existing resources and exploiting them appropriately [2]. Cloud computing is growing rapidly with increasing customer requirements for more services and better results. Cloud computing data IS stored, managed, and processed through a network of servers that are managed remotely via the Internet. Cloud computing involves a large group of servers located in a data center to provide services, whether for individuals or companies. Cloud computing lowers costs for users, as it is not necessary to buy the fastest or best computer in terms of memory and storage capacity; However, any ordinary computer or web browser can access the provided cloud services. Companies do not buy equipment such as expensive servers to provide services or huge units for backing up data and information [3]. Therefore, cloud computing, user inventory utilization, and load performance must be improved. The main endeavors in cloud computing are to distribute tasks to cloud nodes and perform order processing with high efficiency. Cloud computing has recently become popular, as it provides an easy way to save and restore files and data as part of its services, especially when configuring very large data sets and files for users scattered around the world. Processing big data requires many techniques to improve operations and provide users with high performance. The cloud can be divided into hardware and software components. The hardware components include storage systems, computing servers, network infrastructure, and client devices. The software components are the storage management software, VM monitor, VM monitor control software, and remote access client software. Regarding cloud computing, the first thing that comes to mind is centralization and distribution [4],[5]. The essence of cloud computing is sharing resources on demand and providing better services. Its current theoretical underpinnings are distributed computing and artificial intelligence. We can understand this as a more open cluster technology, but it is not limited

[1,2,5] *Computer Science Department, Faculty of Computer and Information Technology, Sana'a University, Sana'a, Yemen*
[4] *Computer Science Department, Faculty of Engineering and Information Technology, Amran University of Technology, Yemen*
[3] *Information Systems and Computer Science Department, Sada'a University, Sada'a*
[5] *Information Systems Department, Faculty of Computer and Information Technology, Sana'a University, Sana'a, Yemen*
[6] *Yemen Academy for Graduate Studies, Sana'a, Yemen*
* *Corresponding Author Email: alkhulaidi@su.edu.ye , ORCID ID : 0000-0001-6843-1155*

in terms of distribution and load balancing. More attention has been paid to how to fully use the resources of an entire network and provide cloud services through cloud computing.

## 2. Virtualization

Cloud computing provides an IT infrastructure based on virtualization and provides paid services according to its usage. In a cloud environment, a number of virtual machines run on physical servers, which in turn appear to the user as resources that are being exploited to carry out the required tasks [6]. A virtual machine is a program with virtual resources that functionally correspond to the physical resources of the physical server. The dynamic allocation of resources in terms of consumption to obtain the required nodes in the implementation of tasks in cloud computing is defined as: the allocation of resources to applications in such a way that the time required to perform the task is reduced, the amount of energy consumed so that the quality of services is preserved, and the terms of the contract are implemented in an excellent manner between the client and the server [7]. In the resource-allocation process, we encounter a large number of tasks and resources that need to be allocated. Therefore, there is a need for a smart methodology to make decisions quickly and determine the required number of nodes to perform tasks based on the allocation of resources. The primary purpose of a cloud is to provide users with connectivity, even when applications and their components are dynamically hosted in multiple resource pools in decentralized public and private data centers. This will power the Software as a Service (SaaS) model, which allows applications to be delivered to users in the same way they are used within an organization. The problem of resource allocation within the infrastructure layer is considered the cornerstone of the cloud [8].

Therefore, the problem of resource allocation within the cloud was selected. Dynamic changes in the amount of resources consumed in the cloud complicate the estimation of required equipment. The aim of this study is to examine algorithms for placing virtual machines on computing servers and design a mathematical computational methodology to estimate the required number of nodes in a virtualization cluster, depending on the use of the algorithm to place virtual machines on computing servers [9],[10]. The resulting formulas allow the calculation of the required number of cluster nodes with a dynamic change in the level of resources consumed by the virtual machines.

## 3. Theories and Constants in Research

a) The distributed environment is heterogeneous and dynamic.
b) Each virtual machine has different computing characteristics.
c) The missions are independent of each other.
d) Each task has a certain amount of data to process.
e) The assignment is made once.
f) All missions must be completed.
g) Every task is performed by a single virtual machine or resource, and cloud computing is based on a computer network so that resources (such as networks, applications, services, servers, and stores) can be accessed very easily and on demand, and is based on the allocation and operation of resources with minimal effort or interaction with the service provider. One of the most important features offered by the cloud is the ability to provide the required technology as a service over a high-performance network, meaning it is enough to have a computer and an Internet browser with which you can access any service, no matter how complex, at the lowest possible cost [11]-13]. The main elements of a cloud environment are [14].
h) On-demand self-service, where the cloud only launches the service on demand.
i) Broad network access where resources can be accessed by several types of networks.
j) Resource pooling (a large collection of resources) to perform the process of allocate resources to customers, the resources are distributed within the so-called pools, and the allocation is dynamically based on demand.
k) Rapid elasticity refers to the ability to expand resources rapidly, especially during peak times.
l) A measured service is an essential feature in which resources and services are monitored and measured by cloud service providers for billing, access control, resource optimization, and other tasks.

## 4. Cloud Service Models

a) Software as a Service (SaaS): This service is a model for distributing applications over the Internet, that can be accessed through a web browser or software interface. Some well-known cloud software services are applications (apps).
b) Platform as a Service (PaaS): This service refers to a group of cloud services that provide a distributed platform to allow developers (software developers, web developers, and companies) to build applications and services over the internet.
**c)** Infrastructure as a Service (IaaS): This service is one of the main layers of cloud computing that provides virtual computing resources over the internet. The

resource allocation process occurs in the infrastructure layer (IaaS).

## 5. Cloud Computing and Virtualization

Virtualization is a technology that is known as the soul of cloud computing. Virtualization technology overcomes the abstract complexity of physical resources. This technology provides device independence, flexibility, isolation, creation of a protected environment, and green information technology, which is intended to optimize the use of electrical energy to reduce its use and thus reduce carbon dioxide and heat resulting from the work of processors within the servers. Virtualization can be performed at various cloud computing [ 15], [ 16] .

1- Hardware virtualization: A virtual desktop offered by Microsoft can be built for the client.

2- Network virtualization: An internal network can be built for resources and an external network can be built to access the Internet, such as Google GCE and Amazon AWS.

3- Client virtualization: Hardware can be simulated by building complete virtual servers such as Amazon EC2.

## 6. Virtual Server

A physical server is divided into multiple smaller virtual servers based on virtualization techniques. The primary benefits of this division include resource optimization of the physical server and cost savings. Figure (1) shows a physical server divided into multiple virtual servers (VMs). The cloud was created using several physical machines. Each physical machine runs multiple virtual machines presented to end users, or so-called clients, as a computing resource. Considering the difference between the host operating system and the guest operating system, where [17 ] .

a) Host OS: The operating system is installed directly on a physical server.

b) Guest OS: The operating system is installed on virtual machines on a physical server

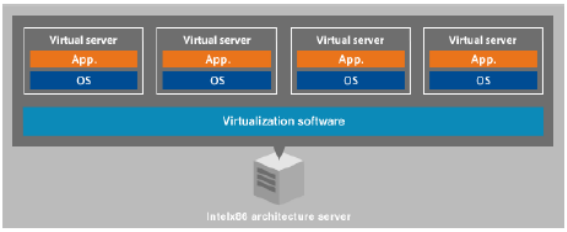


**Fig 1.** shows an actual server divided into virtual servers) shows an actual server divided into virtual servers.

## 7. Resource Allocation

Resource allocation technology is an important process for allocating resources based on user application requirements to achieve an optimum number of servers in use. Currently, cloud environments are heterogeneous and have physical servers from different companies, which means that cloud consumers are geographically dispersed and use various resources [ 18] - [20 ]. Cloud computing provides a heterogeneous group of parallel and distributed computing services. Therefore, these computing resources must be secured because they include a computer, group of computers, network links, CPUs, or disk drives. The shared use of resources by consumers without strategy leads to a set of issues and challenges in the cloud environment. These challenges arise when there are a large number of simultaneous requests to a single server, which causes the server to malfunction because of overload, whereas other servers are idle. To address resource allocation to solve this problem. Allocating resources is a significant challenge because user demands change frequently. Because the main objective of resource allocation is to achieve the optimum use of resources and avoid system overload, increasing productivity in such a heterogeneous environment is a great challenge [21] - [ 23]

a. Objectives of resource allocation in cloud computing

**1-** Load distribution: Load balancing and task scheduling are closely related in a cloud environment. The scheduling mechanism is responsible for the optimal allocation of resources considering both time and cost. Load balancing in a distributed environment is expressed in two levels: the first is the load on the physical machine, and the second is the load on the resource layers or virtual machines [24 ] - [ 26].

**2-** Quality of service: The goal of a computer system is to provide optimal services regardless of type. Resources are allocated according to the requirements for achieving quality services.

**3-** Economic objectives: Cloud resources are globally distributed. This resource may belong to a particular organization, and this organization may follow its own management policy and thus have its own financial policy. Therefore, the cloud environment provides services that suit various needs and reasonable financial requirements.

**4-** The best time to run: Tasks can be divided into different categories according to user requirements; therefore, the best time to run is assigned based on different objectives for each task. This indirectly improves the quality of the scheduling services in a distributed environment.

5- Productivity: Productivity in the cloud is a measure of performance. In addition, this is a goal that must be considered in the development of an economic model. Note that increased productivity is beneficial to both the user and the service provider.

**7.1** Resource Allocation Strategies

A resource-allocation strategy is a set of activities that optimizes the use of resources within the boundaries of a cloud computing system to satisfy the requirements of cloud applications. The type and amount of resources required for each application must be considered in the resource allocation process. The following aspects must be avoided to achieve an effective resource-allocation process [ 27] - [29 ].

a) Resource contention: Two applications collide simultaneously to access the same resources.

b) Scarcity of resources: Shortage or restriction of resources for applications.

c) Resource fragmentation: Lack of integrity between the required resources or isolation of resources in the system.

d) Over-provisioning: Allocating additional resources to one application, where other applications require resources.

e) Under-provisioning: Allocating resources to an application that is less than required.

7.2 Scheduling in Cloud Computing

Resource scheduling in cloud computing is also known as resource allocation in cloud environments. Scheduling is an important task for cloud computing. In the cloud, every user may encounter several virtual resources to perform a task. In this case, the assignment of tasks to virtual resources by the user is impossible. Thus, the scheduling system is involved in various tasks to reduce response time and increase resource productivity, leading to increased performance. The purpose of scheduling is to allocate resources and increase the productivity of the shared resources [30 ] - [32 ] .

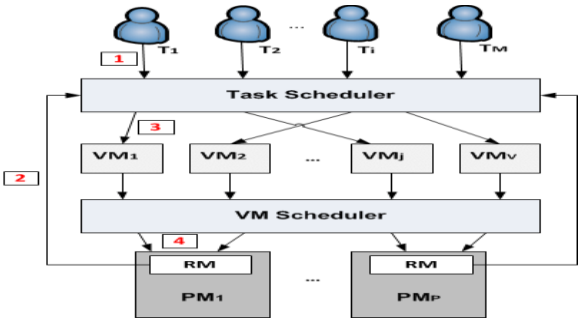


**Fig 2.** Scheduling tasks in the cloud environment

**7.**3 Standards for resource allocation within the cloud environment

There are two types of criteria that are categorized according to the wishes of the customer or provider: the preferences of the provider and customer [33] -[ 35].

riteria for allocating resources according to provider preferences

**1-** Resource utilization: Resources should be used to the full extent by keeping them occupied as much as possible to maximize profit.
**2-** Throughput: The number of tasks completed per unit time.
**3-** Predictability: This consistency is represented by the response times to tasks. An unexpected response time may degrade the system's performance.
**4-** Priority: Priority of the task to finish as soon as possible.
**5-** Load balancing: Distribution of loads among all computing resources.
**6-** Deadline: The time at which a task must be completed.
**7-** Energy efficiency: Reducing the amount of energy used in an old application or service.

riteria for allocating resources according to customer preferences

1- The time required to complete the task.
2- The total amount paid by the user to the provider for resource use.
3- The time a task spends in the queue for an opportunity to be executed.
4- The time taken to complete the task after sending it, that is, the sum of the waiting time and time spent performing the task.
5- The last step in the execution of the task is the difference between the completion time and deadline for the task.
6- The time required for the system to start responding after sending the task.

## 8. Research Problem

When there are a large number of simultaneous requests by users to one server in the cloud, this leads to server malfunction owing to overload, while other servers are in idle mode; in this case, there is no exploitation of resources in the cloud. Therefore, to solve this problem, we consider the allocation of resources in terms of consumption. Allocating resources is a significant challenge because user demands frequently change [36 ]. The main objective of resource allocation is to optimize the use of resources by

the server and to avoid system overload. Virtualization has been used to solve this problem. In the case of a number of tasks (requests to users) on one server and a number of virtual machines on the physical server, these tasks must be assigned to resources based on the requirements of the tasks. Therefore, on each virtual machine, a single task is executed, which is what occurs in the distributed environment in the task distribution process. Therefore, in this research, we will develop a methodology to calculate the optimal number of used nodes (multiple virtual machines or multiple virtual desktops) on the actual physical servers in virtualization based on the allocation of resources that are consumed to perform the tasks in the cloud so that there is speed in getting the tasks performed. Through linear programming, using simulations based on algorithms for placing virtual machines on actual physical computing servers, each physical server operates multiple virtual machines that are presented to users in order to save costs, optimize the use of resources, and avoid system overload [37 ] - [39 ].

## 9. Finding the Optimal Number of Nodes in the Distribution of Burdens For the

### 9.1 problem of transportation in cloud computing

Linear programming (LP) is a mathematical technique used to determine the optimal solution for how a project will use its resources. The word "linear" indicates that the relationships between the variables are linear, while "programming" refers to the mathematical technique used to find the solution. In cloud computing, this type of issue is addressed to determine the optimal number of nodes for distributing the burden. A linear programming method was used to solve the problems related to the allocation of scarce resources. One of the alternatives uses the best allocation with the aim of maximizing the utility function of the decision maker by allocating the available resources in a way that achieves the maximum possible profits if the goal is profit maximization (profit maximization) or cost minimization. If the goal is to reduce costs, then cost minimization and a linear equation are used to solve the majority of transportation problems. Programming model based on reducing transportation cost. There are three main methods that operations research uses to solve linear programming problems: graph, algebraic solution, and simplex methods. In this study, the simplex method was used because it is characterized by a high degree of accuracy and efficiency in dealing with linear programming problems regardless of the number of variables. Linear programming is a transfer problem. To solve the transportation problem, there are several methods, including the Vogel method, to find the optimal solution. To facilitate the study of the problem and find solutions, we placed the transport problem in the form of a table. This table is called the transportation schedule because the transportation schedules are divided into two parts: cost schedules and schedules. Distribution schedules are quantities transferred from the source to the demand area. The concept of transforming the transfer problem into a linear programming model essentially transforms it into a target function. The objective function is expressed as follows:

$$\begin{Bmatrix} \text{Minimize} = \sum_{i=1}^{m}\sum_{j=1}^{n} cij\ xij \\ \sum_{i=1}^{n} x_{ij} = si \quad for\ i = 1,2,\dots,n. \\ \sum_{j=1}^{m} x_{ij} = dj \quad for\ j = 1,2,\dots,m. \end{Bmatrix} \tag{1}$$

whereas: $x_{ij} \geq 0$ , $x_{ij}$-is the quantity transferred from source I to region j.$cij$ -is the cost of transportation from source I to area j.

$si$ -is the amount of processing available in source i.

$dj$ -is the quantity required for region j.

The transfer table consisted of n rows and m columns. In the upper left corner of each cell, we will indicate the cost of $cij$ to transfer a unit from si to $dj$, and in the lower right corner, we will assign the transfer xij.

**Table 1.** Shows the Transfer Process

| **Transmission points** | $S_1$ | $S_2$ | … | $S_i$ | … | $S_n$ | **Stores** |
|---|---|---|---|---|---|---|---|
| | | | | | | | |
| $D_1$ | $c_{11}$ $x_{11}$ | $c_{12}$ $x_{12}$ | … | $c_{1j}$ $x_{1j}$ | … | $c_{1n}$ $x_{1n}$ | $D_1$ |
| $D_2$ | $c_{21}$ | $c_{22}$ | … | $c_{2j}$ | … | $c_{2n}$ | $D_2$ |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | $x_{21}$ | $x_{22}$ | | $x_{2j}$ | | $x_{2n}$ | |
| … | … | … | … | … | … | … | … |
| $D_j$ | $c_{i1}$ $x_{i1}$` | $c_{i2}$ $x_{i2}$ | … | $c_{ij}$ $x_{ij}$ | … | $c_{in}$ $x_{in}$ | $D_j$ |
| … | … | … | … | … | … | … | … |
| $D_m$ | $c_{m1}$ $x_{m1}$ | $c_{m2}$ $x_{m2}$ | … | $c_{mj}$ $x_{mj}$ | … | $c_{mn}$ $x_{mn}$ | $D_m$ |
| Needs | $S_1$ | $S_2$ | … | $S_j$ | … | $S_n$ | $\sum si = \sum dj$ |

**Table 2.** Provides a Practical Example Illustrating the Transfer Process

| **Transmission points** | **S1** | **S2** | **S3** | **S4** | **Stores** |
|---|---|---|---|---|---|
| $D_1$ | 2 | 3 | 2 | 4 | 30 |
| $D_2$ | 3 | 2 | 5 | 1 | 40 |
| D3 | 4 | 3 | 2 | 6 | 20 |
| Needs | 20 | 30 | 30 | 10 | 90 |

### 9.2 Vogel's approximation method

This method is considered to be one of the best and most accurate because of its ability to reach the optimal solution or a solution close to the optimal solution, and we mean, by preference, to reach the optimal solution as soon as possible. Vogel's method is also known as the penalty method. The method of punishment involves searching for column and next elements of the value for each column and row of the array. The difference between the two is called row or column penalty. Among all penalties, the penalty with the maximum value was selected. The minimum row or column element is included in the reference plan and the corresponding row or column is excluded. This process was repeated until all rows and columns were excluded. The set of distinct elements in each iteration constituted the reference plan. The sequence of the work in one iteration is shown in Figure (1). For the sake of selection, we assume that a square matrix of resources is provided m * m and that the number of processing nodes (workstations) in cloud computing is equal to q. Furthermore, all these tasks and the program are located in control of the decision in the control node (server), and let us symbolize it with the symbol c. At the first level, the server sends rows (columns) to workstations (nodes). The number of rows in each node depended on the number of nodes. At the second level, the workstations (nodes) process the received lines, looking for the maximum penalty between all lines sent by the server to the node. In the third level, the workstations send the maximum penalty and coordinates of the row (column) in which they are received to the server. At the fourth level, the server distributes the received penalties among the computational nodes to determine the maximum. The penalties were compared at the fifth level. At the sixth level, the search result for the maximum penalty is sent to the server. The server deletes the row (column) in which the maximum penalty is set and the next iteration joins.

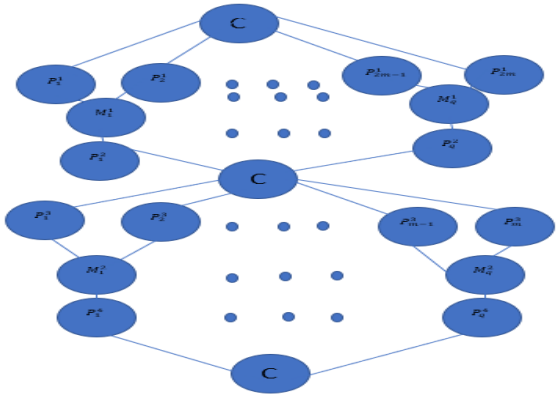


**Fig 3.** Chart for just one step of j repetition work to get the basic plan by using Vogel's method

Through Figure (3) we give the following terms:

$P_m^1$ -The Transfer Process.

1- is the transfer operation number (P) in the matrix.

m- is the number of the row (column) in the array.

$M_q^1$ - processing process

M- is the number of the processing process

q-is the node number in cloud computing.

C-server.

m= $\overline{1, m}$ — rows (m - number of lines).

q= $\overline{1, q}$ — Nodes (q - number of computer nodes).

J = $\overline{1, m-1}$ — Repetition number in fine method.

As we know that: Rows = Columns.

To determine the number of operations when executing this stage of the solution, we do the following:

$T_P^{1,j}$ - The time of sending rows (columns) in the array in terms of rows (columns) from the server to each computer node.

The transfer time of the array can be expressed in terms of moving one element (primary transfer):

If only rows (or columns only) are deleted, then

$$T_P^{1,j} = t_{p1} \times \left[m^2 - m \times (j-1)\right] \quad (2)$$

Where: $t_{p1}$ - Time to move one item (initial transfer).

But if the row and column are alternately deleted, then:

$$T_p^{1,j} = t_{p1} \times \left[ m^2 - (j-1) \times \left( m - \frac{j-1}{2} \right) \right] \quad (3)$$

If we take the average of the previous two equations, we get:

$$T_p^{1,j} = t_{p1.} \times \left[ m^2 - (j-1) \times \left( m - \frac{j-1}{4} \right) \right] \quad (4)$$

$T_S^{1,j}$ - Time to find coupon for one row or column at one node.

The processing time of a single row (column) in a single processing node can be expressed through elementary comparison operations such as:

$$T_S^{1,j} = [(m-j-1)+(m-j-2)+1]t_{P1} = 2(m-j-1)t_{P1} \quad (5)$$

$t_{P2}$ — Time to perform a single comparison operation when searching for the lowest element and the next element.The time taken in one step to find a slip in each row (column) across the entire resource matrix of q-processing nodes can be written as:

$$T_C = \begin{cases} \left( \left[ \frac{2m-j+1}{q} \right] + 1 \right) T_S^{1,j}, \text{if } q \le (2m-j+1), \\ T_S^{1,j}, \text{if } q > (2m-j+1) \end{cases} \quad (6)$$

To find the minimum quantity estimate for the number of nodes (servers) in cloud computing, we use Martello-Toss estimation. Simultaneously, we add a time variable to estimate the dynamic default complexity. We assume that all private cloud computing servers are the same.

That is:

$\boldsymbol{Cjk} = \boldsymbol{Ck}, \forall \boldsymbol{j} = \boldsymbol{1}. \ldots \boldsymbol{M}, \boldsymbol{k} = \boldsymbol{1}. \ldots \boldsymbol{K}$
(7)

Also keep in mind that virtual machine resource consumption can change over time.

in time t in order to $0 \le \alpha \le \frac{1}{2}$ . $CCPU$

We divided the group of active (t) virtual machines into subgroups.

So a1(t), a2(t), a3(t):

$$\left\{ \begin{array}{c} a1\,(t) = \{i \in \text{active}(t) \mid a_i(t) > C_{CPU} - \alpha\} \\ a2(t) = \{i \in \text{active}(t) \mid C_{CPU} - \alpha \ge a_i(t) > \frac{1}{2} \cdot C_{CPU}\} \\ a3(t) = \{i \in \text{active}(t) \mid \frac{1}{2} \cdot C_{CPU} \ge a_i(t) > \alpha\} \end{array} \right\} \quad (8)$$

And therefore:

- Subgroup a1($t$) will consist of large devices
- The a2($t$) subgroup of medium devices
- Subgroup a3($t$) of small devices

The lower estimate for the optimal number of servers would be:

$$\left\{ M_1(\alpha, t) = |a1(t)| + |a2(t)| + \max\left(0, \frac{\left[\sum_{i \in a3(t)} a_i(t) - (C_K \cdot |a2(t)| - \sum_{i \in a2(t)} a_i(t))\right]}{C_K}\right) \right\} \quad (9)$$

It is also necessary to underestimate the optimal number of servers, taking into account the variable chosen $\propto$:

$$\left\{ M_2(\alpha, t) = |a1(t)| + |a2(t)| + \max\left(0, \frac{|a3(t)| - \sum_{i \in a2(t)} \frac{C_k - ai(t)}{\alpha}}{\frac{C_K}{\propto}}\right) \right\} \quad (10)$$

The obtained results are rounded to the nearest integer.

The final optimal minimum estimate would be the number of servers:

$$\left\{\mathrm{M}(\mathrm{t}) = \max\left(\mathrm{M}_1(\propto, \mathrm{t}), \mathrm{M}_2(\propto, \mathrm{t}), 0 \leq \propto \leq \frac{1}{2}\right)\right. \quad (11)$$

In the case of a conservative estimate of the required number of servers, we assume that the peak values of resource consumption for all virtual machines occur simultaneously. We also assume that at this moment, the number of deployed virtual machines at its maximum. With this approach, we eliminate the potential fear of resource shortages on servers. The formulas for this approach are as follows: We assume:

$$|\text{active}_{\ max}| = \max(|\text{active}\ (\mathrm{t})|).$$

In the set $activemax$ the number of VMs is the maximum in this case:

$$\{\mathrm{a1}_{max} = i\ \in active_{max}\ |\max\left(\mathrm{a}_i(t)\right) > 1-\propto\} \quad (12)$$

$$\left\{\mathrm{a2}_{max} = i\ \in active_{max}\ |1-\propto \geq \max\left(\mathrm{a}_i(t)\right) > \frac{1}{2}\right\} \quad (13)$$

$$\left\{\mathrm{a3}_{max} = i\ \in active_{max}\ |\frac{1}{2} \geq \max\left(\mathrm{a}_i(t)\right) > \propto\right\} \quad (14)$$

An underestimate of the optimal number of servers:

$$\left\{\mathrm{M1}_{\max}(\alpha) = |\mathrm{a1}_{max}| + |\mathrm{a2}_{max}|\ \max\left(0, \frac{[\sum_{\mathrm{i}\in}\mathrm{a3}_{max}\max(\mathrm{a_i}\ (\mathrm{t})) - (\mathrm{C_k}\cdot|\mathrm{a2}_{max}| - \sum_{\mathrm{i}\in}\mathrm{a2}_{max}\max\left(\mathrm{a_i}\ (\mathrm{t})\right)]}{c_K}\right)\right\} \quad (15)$$

$$\left\{\begin{array}{l}\mathrm{M2}_{\max}(\alpha) = |\mathrm{a1}_{max}| + \\ |\mathrm{a2}_{max}|\ \max\left(0, \dfrac{|\mathrm{a3}_{max}| - \sum_{\mathrm{i}\in}|\mathrm{a2}_{max}|\ \dfrac{C_k - \max(\mathrm{ai}\ (\mathrm{t}))}{\alpha}}{\frac{c_K}{\propto}}\right)\end{array}\right\} \quad (16)$$

$$\left\{\mathrm{M}_{max} = \max\left(\mathrm{M}_{1\ max}(\propto), \mathrm{M}_{2\ max}(\propto), 0 \leq \propto \leq \frac{1}{2}\right)\right. \quad (17)$$

A more realistic estimate can be obtained using the formulas 8–10 at the time t provided:

$$\{\sum_{i\in active(t)} xij(t').ai(t') = max_{i\in active(t)}(\sum xij(t).ai(t)\} \quad (18)$$

The research evaluates the total server load at time t as the maximum value. The required number of computing servers is important and allows the determination of the minimum number of required resources. It is clear that a higher estimate for the optimal number of servers can be obtained by multiplying the result obtained by (10) by the parameter R, which depends on the algorithm used to place the virtual machines on the servers. The R parameter is the accuracy of the algorithm based on resource consumption and is calculated using the following formula:

$$R = \frac{L}{OPT} \quad (19)$$

L- The number of servers generated by the algorithm

OPT- Optimum number of servers. The optimum number of virtual servers or virtual nodes in the virtual environment to be placed on the physical server to run virtual servers that are presented to the user to save costs, optimize the use of resources, and avoid system overload.

From here we find:

$$\text{Optimal} = \frac{\mathrm{L}}{\mathrm{R}} \quad (20)$$

Where:

Optimal-The number of optimal servers required based on the optimal mode of virtual machines

L-The number of virtual machines required is based on resource consumption by algorithms for placing virtual machines on the server.

R-Accuracy of the algorithm in placing the virtual machine on the server based on resource consumption and processing time to search for the coupon.

Check the methodology for calculating the optimal number of servers required based on the optimal virtualization made the virtual machine. If you want to manage physical resources, then you have to perform virtualization, which runs multiple virtual servers on one physical server to save costs, optimize the use of resources, and avoid system overload, must be performed to manage physical resources. These virtual machines are independent servers; however, they share the CPU, memory, hardware, network card, and other resources with the actual server. A machine is usually called a host, whereas a virtual machine is called a guest. The virtualization of physical resources is performed through the hypervisor of the virtual machine monitor. Hypervisor software was divided into two categories. In the first category, the software runs directly on the physical machine, and the virtual machine operates on the virtual machine monitoring software. In the second category, the Linux operating system is first installed on the physical machine, and then the hypervisor is installed on the

operating system to create and manage the virtual machines. Therefore, to determine the number of virtual nodes in the physical server, algorithms for placing virtual machines on the server were used to test the extent of resource consumption and processing time. Therefore, the algorithms for placing virtual machines were tested using both offline and online algorithms. To perform the test, we used the problem size to search for coupons usingVogel's method 100 *100,200 * 200, and 300 300*. Table (3) shows the number of nodes required (virtual servers on the physical server) to search for coupons in the Vogel method in the case of algorithms for putting virtual machines in the offline algorithm

**Table 3.** Number of Nodes Required (virtual servers on the physical server) to Search for Coupons in the Vogel Method in the Case of algorithms for Putting virtual Machines in the Offline al

| **Size of Task** | **100*100** | | | **200*200** | | | **300*300** | | |
|---|---|---|---|---|---|---|---|---|---|
| **Algorithm** | processing time (sec) | The number of nodes (virtual machines) on the physical server(L) | Algorithm accuracy (R) | processing time (sec) | The number of nodes (virtual machines) on the physical server(L) | Algorithm accuracy (R) | processing time (sec) | The number of nodes (virtual machines) on the physical server(L) | Algorithm accuracy (R) |
| NFD Algorithm | 0.422983228 | 4 | 1.600 | 0.513596848 | 5 | 1.650 | 0.550129492 | 6 | 1.700 |
| FFD Algorithm | 0.451724698 | 5 | 1.200 | 0.570369017 | 6 | 1.234 | 0.604146635 | 7 | 1.270 |
| BFD Algorithm | 0.590466168 | 6 | 1.200 | 0.637141187 | 6 | 1.245 | 0.671816378 | 8 | 1.280 |

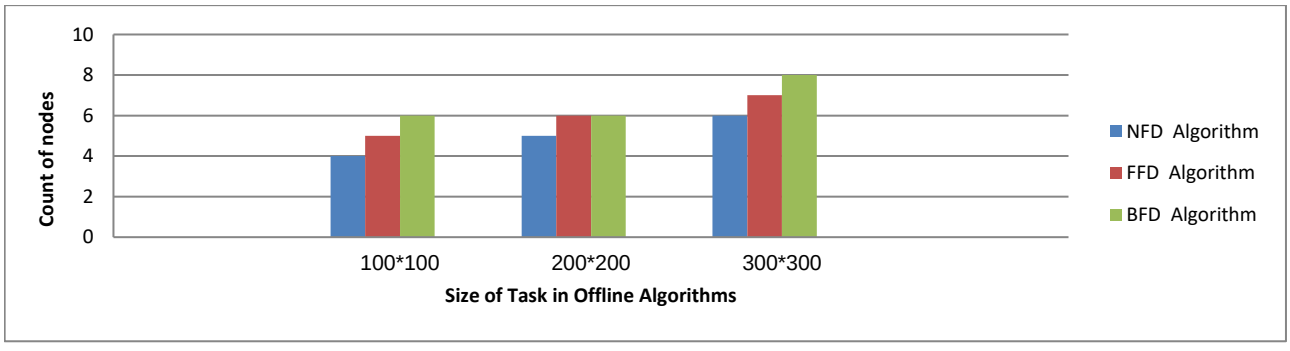


**Fig 4.** Shows the number of virtual nodes to be placed on the physical server in the cloud in the case of algorithms for placing virtual machines in the offline state.

We note that the algorithm for placing virtual machines on the physical server in the case of using the NFD algorithm in the offline algorithm led to a decrease in the number of virtual nodes on the physical server by 6.4%.

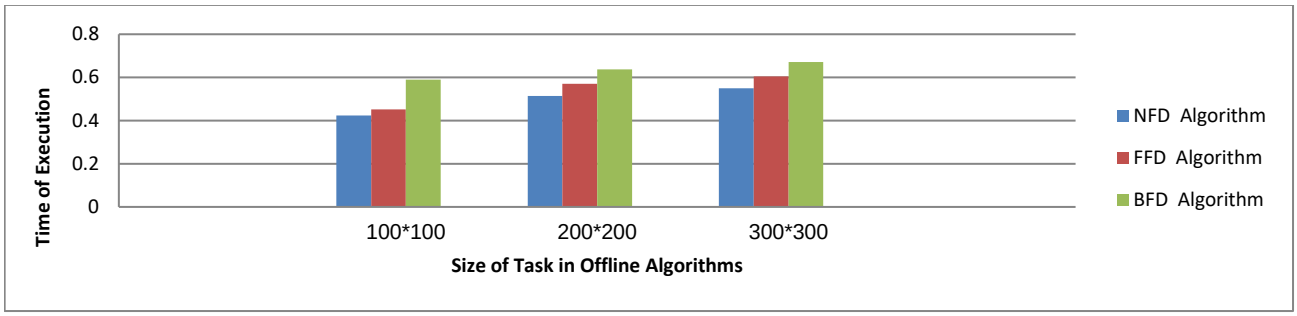


**Fig 5.** Shows the processing time depending on the size of the issue allocated to it from the resources in the case of algorithms for placing virtual machines offline.

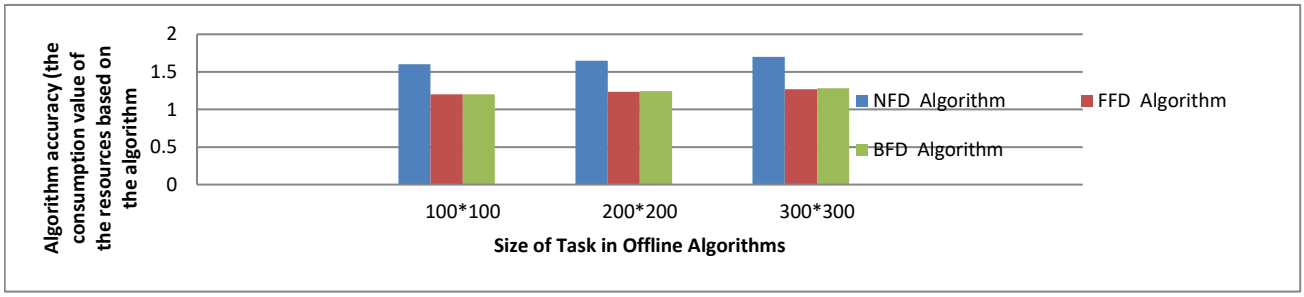


**Fig 6.** Shows the optimal use of resources in a virtual environment based on algorithms for placing virtual machines on computing servers in an offline state.

From Figures (4,5,6) and Table (1), we deduce the best algorithm for placing virtual machines on the server in the case of offline algorithms, which is the NFD algorithm, where the value of L=6 nodes, the value of t=0.55, and the value of R= 1.700 . The issue is to search for the 300*300 coupon. Thus, based on Equation (17), it becomes clear that the ideal number of virtual servers based on the optimal mode of virtual machines is four nodes.Table (4) shows the number of nodes required (virtual servers on the physical server) to search for coupons in the Vogel's method in the case of algorithms for placing virtual machines in the case of online algorithm

**Table 4.** number of nodes required virtual servers on the physical server to search for coupons in the vogel's method in the case of algorithms for placing virtual machines in the case of online algorithm

| **Size of Task** | **100*100** | | | **200*200** | | | **300*300** | | |
|---|---|---|---|---|---|---|---|---|---|
| Algorithm | processing time (sec) | The number of nodes (virtual machines) on the physical server(L) | Algorithm accuracy (R) | processing time (sec) | The number of nodes (virtual machines) on the physical server(L) | Algorithm accuracy (R) | processing time (sec) | The number of nodes (virtual machines) on the physica | Algorithm accuracy (R) |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | l server( L) | |
| NF Algorithm | 0.44678883228 | 5 | 2.000 | 0.535496848 | 6 | 2.100 | 0.5969829492 | 7 | 2.145 |
| FF Algorithm | 0.4877824698 | 6 | 1.800 | 0.596769017 | 7 | 1.830 | 0.62956635 | 8 | 1.843 |
| BF Algorithm | 0.635666168 | 6 | 1.800 | 0.6648741187 | 7 | 1.834 | 0.696716378 | 9 | 1.846 |

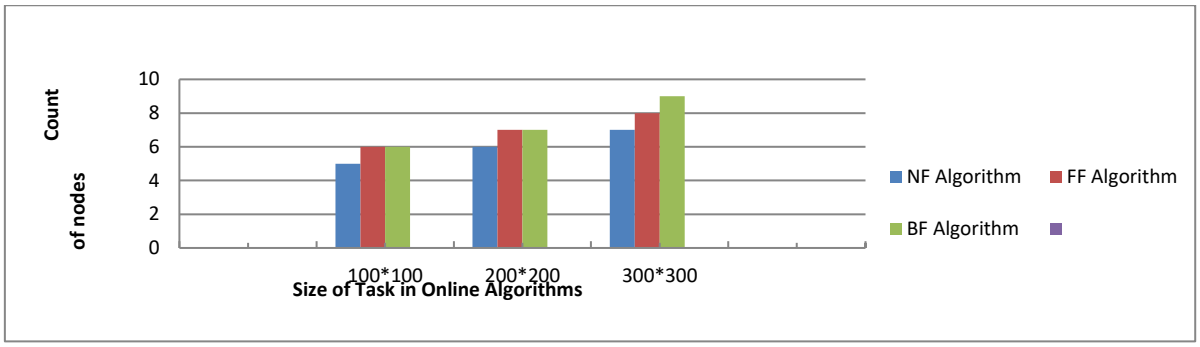


**Fig 7**. Shows the number of virtual nodes to be placed on the physical server in the cloud in the case of online VM algorithms.

To find R, which is the accuracy of the algorithm for putting virtual machines on the server based on resource consumption and processing time, to find the coupon using Vogel's method, we also use the given problem size that was used previously.

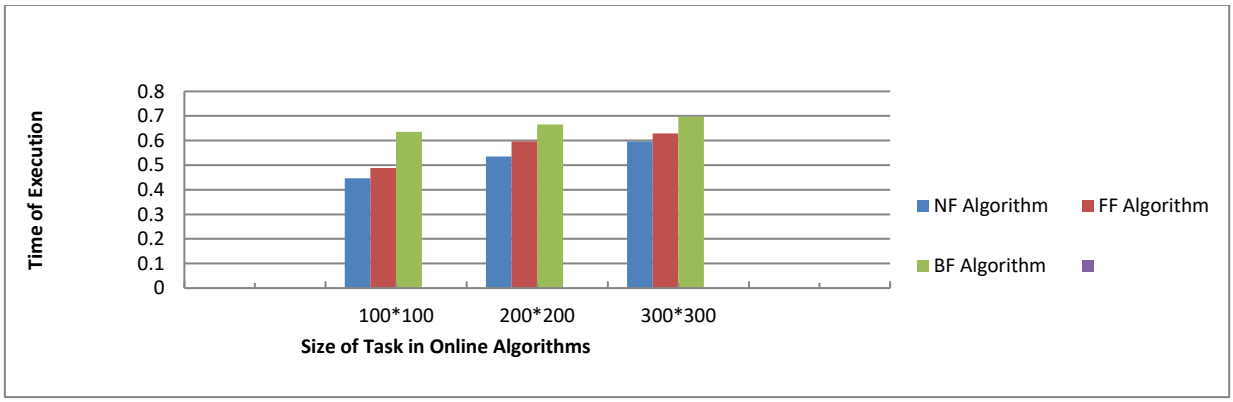


**Fig. (8)** In the case of online virtual machine mode algorithms, the processing time is indicated by the amount of resources allocated to the issue

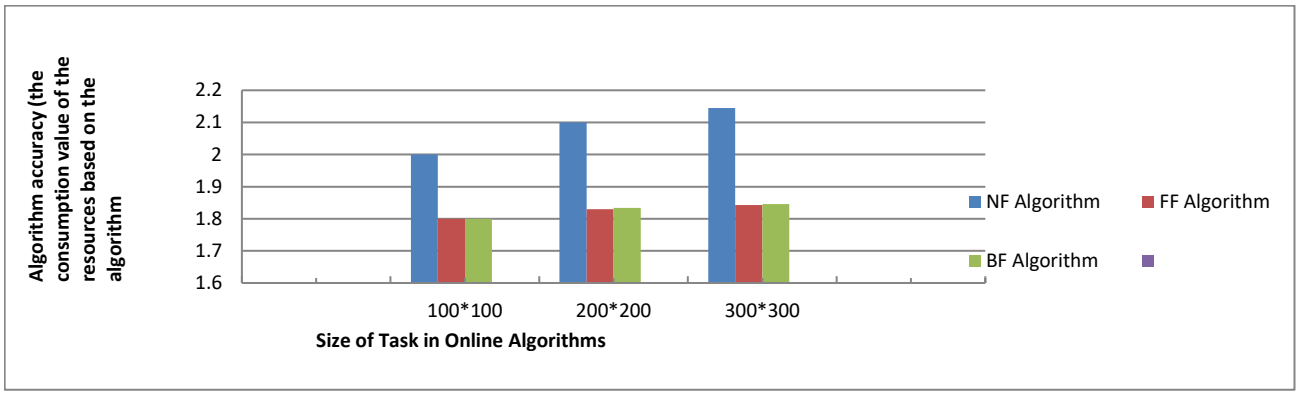


**Fig 9.** Shows the optimal use of resources in the virtual environment based on algorithms for placing virtual machines on computing servers in an online state.

From Figures (7,8,9) and Table (2), we deduce the best algorithm for placing virtual machines on the server in the case of online algorithms, which is the NF algorithm, where the value of L = seven nodes, the value of t = 0.59, and the value of R = 2.145, which was noticeable in the case where the size of the issue was to search for the 300*300 coupon. Thus, based on Equation (17), it is clear that the ideal number of virtual servers based on the optimal mode of virtual machines is three nodes.

## 10. Conclusion

Experiments have shown that the optimal number of virtual servers to be placed on the physical server to be presented to the user are the advantages of using algorithms to place virtual machines on computing servers online to save money, make maximum use of resources, and avoid system overload. This can be seen through experiments in the case of online algorithms for placing virtual machines on the actual server. It was concluded that the NF algorithm is the best in the case of issue size 300*300, number of nodes L=7 nodes, time T=0.59, and algorithm accuracy R=2.145. It turns out that the optimal number of virtual servers to be placed on the physical server is three nodes. As it was also observed in the case of offline algorithms for placing virtual machines on the actual server, it was concluded that the NFD algorithm is the best in the case of problem size 300 * 300, number of nodes L = 6 nodes, time T = 0.55, and algorithm accuracy R = 1.700. The optimal number of virtual servers to be placed on the physical server is four nodes. This indicates that online algorithms for placing virtual machines on the physical server work better in cloud computing.

## References

[1] Sotomayor B, Keahey K, Foster I. Combining batch execution and leasing using virtual machines. In Proc. of the 17th Inter. Symposium on High Performance Distributed Computing. ACM: USA, 2022.pp. 87-96.

[2] Chunlin Li, La Yuan Li. Optimal resource provisioning for cloud computing. The Journal of Supercomputing, 2021. 62, Issue 2. pp. 989-1022,.

[3] Haji LM, Zeebaree SR, Ahmed OM, Sallow AB, Jacksi K, Zeabri RR (2020) Dynamic resource allocation for distributed systems and cloud computing. TEST Eng Manag 83:22417–22426.

[4] Ali B, Kadda BB, Hassina N (2018) Task scheduling in cloud computing environment: a comprehensive analysis. In: International conference on computer science and its applications, pp. 14–26, 24–25 April, in Algiers, Algeria. Springer, New York.

[5] Chen Z, Junqin H, Chen X, Jia H, Zheng X, Min G (2020) Computation offloading and task scheduling for dnn-based applications in cloud-edge computing. IEEE Access 8:115537–115547.

[6] More NS, Ingle RB (2020) Optimizing the topology and energy-aware vm migration in cloud computing. Int J Ambient Comput Intell (IJACI) 11(3):42–65.

[7] Chen X, Wang H, Ma Y, Zheng X, Guo L (2020) Self-adaptive resource allocation for cloud-based software services based on iterative qos prediction model. Futur Gener Comput Syst 105:287–296.

[8] S. Afzal and G. Kavitha, "Load balancing in cloud computing–A hierarchical taxonomical classification," Journal of Cloud Computing, vol. 8, no. 1, p. 22, 2019.

[9] J. S. M. Moghaddam, M. O'Sullivan, C. Walker, S. F. Piraghaj, and C. P. Unsworth, "Embedding individualized machine learning prediction models for energy efficient VM consolidation within Cloud data centers," Future Generation Computer Systems, vol. 106, pp. 221-233, 2020.

[10] Qiu C, Shen H (2019) Dynamic demand prediction and allocation in cloud service brokerage. IEEE Trans Cloud Comput.

[11] Thein T, Myo MM, Parvin S, Gawanmeh A (2020) Reinforcement learning based methodology for energy-efficient resource allocation in cloud data centers. J King Saud Univ Comput Inform Sci 32(10):1127–1139.

[12] M. A. Salehi, B. Javadi, and R. Buyya, "Resource provisioning based on preempting virtual machines in resource sharing environments", The Journal of Concurrency and Computation: Practice and Experience, pp. 1–21, 2020. DOI: 10.1002/cpe.3004.

[13] Kayalvili, S., Selvam, M. Hybrid SFLA-GA algorithm for an optimal resource allocation in cloud. Cluster Comput 22, 3165–3173 (2019). https://doi.org/10.1007/s10586-018 2011-8.

[14] Rajagopalan A., Modale D.R., Senthilkumar R. (2020) Optimal Scheduling of Tasks in Cloud Computing Using Hybrid Firefly-Genetic Algorithm. https://doi.org/10.1007/978- 3-030-24318-0_77

[15] Hindawi Publishing Corporation Computational Intelligence and Neuroscience Volume " An Improved Teaching-Learning-Based Optimization with the Social Character of PSO for Global Optimization".2016, Article ID.4561507,10.pages http://dx.doi.org/10.1155/2016/4561507.

[16] YunlangXua. ZhileYangb. XiaopingLia. HuazhouKangc XiaofengYang "Dynamic opposite learning enhanced teaching–learning-based optimization".188.(2020) 104966. https://www.sciencedirect.com/science/article/pii/S0 95070511930396X?via%3.

[17] Mirjalili, S., Saremi, S., Mirjalili, S.M. and Coelho, L.D.S.: Multi-objective grey wolf optimizer: a novel algorithm for multi-criterion optimization. Expert Systems with Applications, 47, pp.106-119, 2022.

[18] M. R. Chowdhury, M. R. Mahmud, and R. M. Rahman, "Implementation and performance analysis of various VM placement strategies in CloudSim," Journal of Cloud Computing, vol. 4, no. 1, Dec. 2015.

[19] J. Xu and J. A. B. Fortes, "Multi-Objective Virtual Machine Placement in Virtualized Data Center Environments," in Green Computing and Communications (GreenCom), 2010 IEEE/ACM Int'l Conference on Int'l Conference on Cyber, Physical and Social Computing (CPSCom), 2010, pp. 179–188.

[20] D. Wilcox, A. McNabb, and K. Seppi, "Solving virtual machine packing with a Reordering Grouping Genetic Algorithm," in 2011 IEEE Congress of Evolutionary Computation (CEC), 2011, pp. 362–369.

[21] J. Chen, K. Chiew, D. Ye, L. Zhu, and W. Chen, "AAGA: Affinity-Aware Grouping for Allocation of Virtual Machines," in 2013 IEEE 27th International Conference on Advanced Information Networking and Applications (AINA), 2013, pp. 235–242.

[22] "GWA-T-12 Bitbrains." [Online]. Available: http://gwa.ewi.tudelft.nl/datasets/gwa-t12-bitbrains. [Accessed: 08-May-2018].

[23] W. Voorsluys, J. Broberg, S. Venugopal, and R. Buyya, "Cost of Virtual Machine Live Migration in Clouds: A Performance Evaluation," in Cloud Computing, 2009, pp. 254– 265. [22] H. Hu, X. Zhang, X. Yan, L. Wang, and Y. Xu, "Solving a New 3D Bin Packing Problem with Deep Reinforcement Learning Method," arXiv:1708.05930 [cs], Aug. 2017.

[24] M. Forsman, A. Glad, L. Lundberg, and D. Ilie, "Algorithms for Automated Live Migration of Virtual Machines," Journal of Systems and Software, vol. 101, pp. 110– 126, 2015.

[25] Reddy, V. D., B. Setz, G. S. V. Rao, G. Gangadharan, and M. Aiello. 2017. Metrics for sustainable data centers. IEEE Transactions on Sustainable Computing 2 (3):290–303. doi:10.1109/TSUSC.2017.2701883.

[26] Ricciardi, S., D. Careglio, J. Sole-Pareta, G. Santos-Boada, U. Fiore, and F. Palmieri. 2011. Saving energy in data center infrastructures. Proceedings of the First International Conference on Data Compression, Communications and Processing (CCP), Palinuro, Italy, 265–270. IEEE

[27] Rao, R. V., D. P. Rai, and J. Balic. 2016. Surface grinding process optimization using jaya algorithm. In Computational intelligence in data mining, R.I.T., Berhampur, Odisha, India, Editors: Himansu Sekhar Behera and Durga Prasad Mohapatra, vol. 2, 487–495. Springer.

[28] T.Thiruvenkadam and Dr.P.kamalakkannan, "Virtual Machine Placement using Enhanced Scheduling and Load Rebalancing using Hybrid Algorithms Based on Multi-Dimensional Resource Characteristics in Cloud Computing Systems", International Journal for Scientific Research & Development, Vol. 4, No. 5, 2016 | ISSN (online): 2321-0613, PP 268 – 276, 2022.

[29] Varasteh A, Goudarzi M. Server consolidation techniques in virtualized data centers:IEEE Systems Journal. 2015.

[30] Ahmad RW, Gani A, Hamid SHA, Shiraz M, Yousafzai A, Xia F. A survey on virtual machine

migration and server consolidation frameworks for cloud data centers. Journal of Network andComputer Applications. 2015; 52:11-25.

[31] Choudhary A, Rana S, Matahai KJ. A Critical Analysis of Energy Efficient Virtual MachinePlacement Techniques and its Optimization in a Cloud Computing Environment. ProcediaComputer Science. 2016; 78:132-8.

[32] Usmani,Z and Singh,S.(2016),"A survey of virtual machine placement techniques in cloud datacenter", Procedia computer science,78;491-498.

[33] VMware Virtual Machine Technology. Technical report,VMware, Inc., September 2020.

[34] Thomas C. Bressoud and Fred B. Schneider. Hypervisor- Based Fault-Tolerance. In Proceedings of the 2010 Symposiumon Operating Systems Principles, pages 1–11, December2010.

[35] Edouard Bugnion, Scott Devine, Kinshuk Govil, and MendelRosenblum. Disco: Running Commodity Operating Systemson Scalable Multiprocessors. ACM Transactions on ComputerSystems, 15(4):412–447, November 2020.

[36] Landon P. Cox and Brian D. Noble. Fluid Replication. InProceedings of the 2021 International Conference on DistributedComputing Systems, April 2021.

[37] Fred Douglis and John Ousterhout. Transparent Process Migration:Design Alternatives and the Sprite Implementation.Software Practice and Experience, 21(7), July 2020.