



Enhancing Reconfiguration in Two-Phase Virtual Machine Placement for Cloud Datacenters Using Tabu Search Algorithm

Abdurrahman Aminu Abdulkadir¹, Faruku Umar Ambursa^{2*}

¹Department of Computer Science, Bayero University, Kano, Nigeria

²Information Technology Department, Bayero University, Kano, Nigeria

ABSTRACT

Cloud computing enables users to have access to resources on demand from cloud datacenter. One of the major challenges in cloud datacenters is virtual machine placement (VMP) which is selecting appropriate physical machines for each virtual machine request. VMP can be online, offline or two-phase which combines both online and offline. For solving the VMP problem, two-phase optimization approach is used recently. The first phase, also called the incremental VMP (iVMP) phase, involves dynamic handling of arriving VM requests from users, where VMs could be created, modified or destroyed at runtime. The second phase, referred to as the VMP reconfiguration (VMP_r) phase, deals with improving the quality of solutions obtained in the iVMP phase. This work aims to improve the VMP_r phase of the two-phase VMP. Previous works used Memetic algorithm in the VMP_r reconfiguration phase. However, a population's lack of diversity causes the Memetic Algorithm to allocate repeated reproductive, mutation and refinement trials to the same individuals thus wasting precious CPU resources. Balancing population diversity is a critical issue. Therefore, another metaheuristic algorithm (Tabu Search) is adopted to solve the problem. The strength of the TS resides in its capacity to escape the trap of local optimality, and therefore gives good solutions of hard combinatorial optimization problems. Experiment has been conducted taking into account 400 different scenarios, experimental evaluation was performed against the benchmark research. The new method gives better aggregate scores of power consumption, economic revenue, and resource usage and reconfiguration time thus achieving better results (i.e Minimum cost).

ARTICLE INFO

Article History

Received: March, 2023

Received in revised form: March, 2023

Accepted: May, 2023

Published online: June, 2023

KEYWORDS

Service selection, cloud service broker, Artificial Intelligence, Particle Swarm Optimization

INTRODUCTION

Cloud computing encompasses two main aspects: the delivery of applications as services over the Internet and the underlying hardware and systems software present in data centers, which facilitate the provision of these services. What distinguishes cloud computing is its pay-as-you-go model, where users are charged based on their actual usage. (Wang et al., 2010). Cloud computing has become a widely adopted technology due to its numerous characteristics and benefits. Some of the key

characteristics of cloud computing include: (1) On-demand self-service: Cloud services are available on-demand, allowing users to provision computing resources such as storage, processing power, and applications as needed (Mell et al., 2011). (2) Cloud services can be conveniently accessed through the network using common platforms, allowing users to reach their applications and data using various devices such as personal computers, laptops, tablets, and smartphones. This broad network access feature provides flexibility and accessibility to users across different

Corresponding author: Faruku Umar Ambursa

✉ fuambursa.it@buk.edu.ng

Information Technology Department, Bayero University, Kano, Nigeria

© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved



devices. (Mell et al., 2011). (3) Resource pooling: Cloud providers consolidate computing resources to serve multiple users simultaneously. These resources, such as processing power, storage, and bandwidth, are dynamically allocated according to demand, resulting in efficient utilization (Mell et al., 2011). (4) Rapid elasticity: the concept of rapid elasticity in cloud computing allows for swift and flexible scaling of resources to accommodate fluctuating workload requirements. Users have the ability to effortlessly request additional resources when there is a surge in demand or release unused resources when they are no longer needed. This dynamic scalability provides both flexibility and cost optimization, as resources can be precisely allocated based on the current needs of the system. (Mell et al., 2011). (5) Measured service: Cloud systems automatically control and optimize resource usage, and users are charged based on their actual consumption. This pay-per-use model allows for cost transparency and efficient resource management (Mell et al., 2011).

NIST (National Institute of Standards and Technology) defines cloud computing as having three service models that cater to different user requirements. These models are as follows: 1. Infrastructure-as-a-Service (IaaS): This service model, also known as the technology layer, offers virtual storage and servers for data analysis purposes. Users have the option to rent web-accessible virtual storage and can easily scale up or down based on their specific needs. 2. Platform-as-a-Service (PaaS): Operating at the application layer, PaaS provides a development platform for building applications. With PaaS, users can procure a platform that allows for the integration of databases from various projects, accommodating the data generated by professionals both on-site and in the back office. 3. Software-as-a-Service (SaaS): SaaS refers to cloud-based construction software that can be obtained on a pay-as-you-go basis, resulting in reduced ownership costs. This model allows users to purchase software for use in the cloud, providing flexibility and cost savings compared to traditional software ownership. These service models offered by cloud computing enable users

to choose the most suitable option based on their specific needs and budgetary considerations.

Cloud services can be deployed in various ways, including as public, private, community, or hybrid services. These deployment models differ in terms of access methods and the categories of users allowed to access the services. In a public cloud, a diverse range of clients can access the cloud service via the internet. It operates as a multi-tenant system, where multiple users' data is processed alongside each other. Applications running on a public cloud can be hosted within a single data center or distributed across multiple data centers. Public clouds are typically well-suited for small businesses. On the other hand, a private cloud is hosted on-premises and exclusively used by a single organization. Access to a private cloud is restricted to the organization's intranet. Private clouds are especially suitable for organizations that need dedicated systems to store sensitive information, such as national infrastructure. This deployment model is commonly employed by large businesses and government institutions for safeguarding sensitive data. In the community cloud model, multiple organizations collaborate closely to share a cloud infrastructure. The community cloud infrastructure is accessible only to the participating organizations, allowing them to securely share resources and services. A hybrid cloud deployment combines two or more types of deployments. For example, a project may utilize a private cloud for financial reporting, where sensitive data is stored, while non-sensitive, high-traffic data can be placed on a public cloud. The hybrid cloud approach offers flexibility in leveraging the advantages of different deployment types based on specific requirements. These various deployment models provide organizations with choices for implementing cloud services that align with their needs, security considerations, and the nature of the data they handle.

Moreover, distributed technologies have encountered various challenges, including performance measurement, managing portability, on-demand provisioning, hosting different applications on a single physical platform, handling sudden spikes in resource demand, and dealing with adversarial user behavior. The

Corresponding author: Faruku Umar Ambursa

✉ fuambursa.it@buk.edu.ng

Information Technology Department, Bayero University, Kano, Nigeria
© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved



following paragraphs discuss these major challenges in detail. (1) Security and Privacy Concerns: Cloud computing introduces security and privacy challenges due to the sharing of resources and data across multiple users and organizations (Marinos et al., 2016). Organizations may have concerns about unauthorized access, compliance with data protection regulations and data breaches. (2) Data Protection and Compliance: Ensuring data protection and compliance with privacy regulations can present challenges for organizations when their data is stored and processed in the cloud. Organizations may encounter difficulties in effectively safeguarding their data and adhering to regulatory requirements in the cloud environment. (Rahman et al., 2020). Compliance requirements such as the European Union's General Data Protection Regulation (GDPR) and industry-specific regulations can pose complexities in cloud environments. (3) Vendor Lock-in: Vendor lock-in refers to the challenge of migrating or switching cloud service providers due to dependencies and proprietary technologies used within a specific cloud platform (Kumbhare et al., 2018). It can limit flexibility, increase costs, and hinder the adoption of alternative cloud solutions. (4) Performance and Reliability: Cloud performance and reliability challenges can arise due to factors such as resource sharing, network congestion, and service-level agreements (Chen et al., 2018). Ensuring consistent and predictable performance, as well as addressing potential downtime and outages, is crucial for organizations relying on cloud services. (5) Virtualization: Virtualization offers a logical abstraction layer that enables the execution of applications, operating systems, and system services in a separate and independent system environment, decoupled from the underlying physical computer systems. By efficiently allocating computer resources such as processors, memory, and I/O devices, virtualization allows for the simultaneous operation of multiple operating systems and software stacks on a single physical computer platform. This is made possible through the use of a hypervisor, a software layer functioning as a virtual machine monitor (VMM), which facilitates

communication between each guest operating system presented as a virtual machine (VM) and the underlying physical hardware. (Uhlig, 2006). (6) Quality of service (QoS): Service Level Agreements (SLAs) regulate the provisioning of services in the cloud, serving as a contractual agreement that outlines the commitment of a cloud provider to deliver specific services to a cloud user. SLAs encompass provisions for compensating users in case of service disruptions or failures, effectively serving as a warranty for cloud users. These agreements, signed between customers and service providers, typically include specifications for non-functional requirements of the service, often referred to as Quality of Service (QoS) parameters. (Mary, 2014). (7) Resource allocation: Cloud services can be found across a range of resources, each with different Quality of Service (QoS) parameters. Therefore, the role of a cloud broker becomes crucial in selecting the most suitable set of cloud services that align with the specific needs and preferences of the user. The cloud broker acts as an intermediary, identifying and recommending the optimal cloud services based on the user's requirements and preferences. (Jula, 2014). (8) Job scheduling: Through effective job scheduling, cloud technology can reduce execution time and enhance resource utilization, leading to lower costs associated with resource usage and energy consumption. Job scheduling plays a vital role in optimizing the allocation of tasks and resources within the cloud environment, enabling efficient execution of jobs and maximizing the utilization of available resources. This, in turn, contributes to minimizing costs and reducing energy consumption, making cloud technology a more cost-effective and sustainable solution. (Lago, 2011). (9) Virtual machine placement (VMP): In a Cloud computing environment, Virtual Machine Placement (VMP) plays a critical role, especially as the number of users continues to grow. VMP significantly impacts the overall system performance and throughput. It involves the strategic placement of Virtual Machines (VMs) onto Physical Machines (PMs) within a datacenter. The goal of VMP is to minimize the number of data centers required while maximizing the utilization of each data center. This approach

Corresponding author: Faruku Umar Ambursa

✉ fuambursa.it@buk.edu.ng

Information Technology Department, Bayero University, Kano, Nigeria
© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved



enhances the availability and flexibility of data centers, while simultaneously reducing operational costs and expenses associated with hardware, such as physical space and energy consumption.

Virtual machine placement can be categorized into three types: (i) Static virtual machine placement (offline), (ii) Dynamic virtual machine placement (online), and (iii) Two-phase, which combines both online and offline approaches. While online placement may not always yield an optimal solution and offline placement is conducted in a static environment that does not consider the dynamic nature of the Cloud environment, a two-phase optimization scheme can effectively address the VMP problem (Pires et al., 2017). This work specifically concentrates on optimizing the offline phase of the two-phase VMP in a cloud computing datacenter.

In view of the above problem, thus, to effectively optimize provider objectives such as resource utilization, power consumption, reconfiguration time and economic revenue, there is a need for improving the static VMP phase, as a result improving the two-phase optimization scheme. As mentioned previously, Virtual Machine Placement (VMP) is a crucial challenge in the cloud, involving the assignment of each virtual machine to an appropriate physical machine. Historically, VMP has been formulated as either an online or offline process. Online VMP involves dynamically assigning virtual machines to suitable physical machines at runtime. While this approach often results in improved but sub-optimal solutions (Pires et al., 2015). On the other hand, offline VMP utilizes a static environment where virtual machines are placed on more suitable physical machines. While the offline formulation provides better alternatives compared to the online approach, it is not well-suited for the dynamic nature of the cloud computing environment.

In recent times, a two-phase formulation has been introduced to address the VMP problem, aiming to provide more efficient solutions by leveraging the strengths of both online and offline approaches. This approach involves a heuristic process that dynamically receives virtual machines for placement at runtime (online), and a

metaheuristic process that optimizes the placement to achieve more optimal solutions (offline) (Pires et al., 2014). By combining these two phases, the two-phase formulation aims to enhance the efficiency and effectiveness of the VMP problem in cloud computing environments. A work on two phase VMP by (Pires et al., 2017) used a metaheuristic evolutionary memetic algorithm in the offline phase of the two-phase optimization scheme. Furthermore, the challenge with memetic algorithm as (Natalio et al., 2006) wherein a lack of diversity within the population leads to the algorithm repeatedly allocating reproductive, refinement processes and mutation to the same individuals, resulting in wastage of valuable CPU resources. Maintaining a balanced population diversity is crucial in addressing this issue. The lack of diversity results to wastage of resources and time which leads the process being trapped in local optima without getting better optimal result, resulting in a suboptimal reconfiguration solution. Thus, affecting the performance of the offline VMP Phase in achieving the objective functions. Therefore, there is a need to introduce another method to solve the mentioned problem. To overcome this limitation, an alternative metaheuristic algorithm. Tabu Search (TS), has been adopted to tackle the problem. Tabu Search possesses the strength of being capable of avoiding local optimality traps, thereby offering effective solutions for complex combinatorial optimization problems (Anass et al., 2017). The aim of this research is to optimize the Virtual Machine Placement reconfiguration in the two-phase virtual machine placement. Similarly, a well-designed Tabu search method has been proposed to optimize the placement criteria. Finally, the previous and proposed method have been evaluated and compared under the same objectives functions.

In the remainder of this paper, Section 2 discusses related works. Section 3 describes the VMP problem, highlights the operation of the previous method and demonstrates the proposed scheme. Section 4 presents the performance results. Finally, Section 5 concludes the paper and suggests future work.

RELATED WORKS

Corresponding author: Faruku Umar Ambursa

✉ fuambursa.it@buk.edu.ng

Information Technology Department, Bayero University, Kano, Nigeria
© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved



This section gives the review on related work on cloud computing, down to VMP problem in cloud; it also discusses the related concepts considered in this work as well as the related research works from which this research problem was derived.

Cloud computing has emerged as a transformative technology in the field of information technology, revolutionizing the way organizations manage and deliver services. This literature review provides an overview of the key concepts, challenges, and recent developments in cloud computing. The review is organized into four sections: (1) Cloud Computing Models, (2) Security and Privacy in the Cloud, (3) Scalability and Performance, and (4) Recent Trends and Future Directions. (1) Cloud Computing Models: Cloud computing offers various deployment models, including public, private, hybrid, and multi-cloud architectures. Recent studies have focused on optimizing these models to address specific organizational needs. For instance, Yang et al. (2022) proposed a multi-cloud resource allocation framework that leverages machine learning techniques to optimize resource utilization. Similarly, (Smith et al., 2021) presented a hybrid cloud model that integrates edge computing to enhance real-time processing and reduce latency. (2) Security and Privacy in the Cloud: Security and privacy remain significant concerns in cloud computing. Researchers have explored different approaches to enhance the security and protect user data in the cloud environment. (Zhang et al. 2022) proposed a secure data sharing scheme that employs attribute-based encryption to ensure confidentiality and integrity. Additionally, (Li et al. 2021) introduced a privacy-preserving machine-learning framework for cloud environments, which uses federated learning to protect sensitive data during the model training process. (3) Scalability and Performance: Achieving scalability and high performance is critical for cloud-based applications. Recent studies have focused on optimizing resource allocation, load balancing, and scheduling techniques. For example, (Chen et al., 2023) proposed a novel load-balancing algorithm that dynamically adjusts resource allocation based on workload characteristics.

Furthermore, (Huang et al., 2022) introduced a scalable storage architecture for cloud systems that leverages distributed file systems and data replication techniques. (4) Cloud computing continues to evolve with emerging trends and technologies. (Liu et al., 2022) investigated the integration of edge computing with cloud platforms to support low-latency and real-time applications. Moreover, (Wang et al., 2023) discussed the adoption of serverless computing in cloud environments and its potential impact on resource management and cost optimization.

As stated earlier, this work focuses on Virtual machine (VM) placement, VM placement is a critical aspect of resource management in cloud computing environments. Efficient VM placement algorithms ensure optimal resource utilization, improved performance, and cost-effectiveness. This literature review provides an overview of key concepts, challenges, and recent developments in VM placement techniques in cloud computing. The review is organized into four sections: (1) VM Placement Strategies, (2) Performance Optimization, (3) Energy Efficiency, and (4) Recent Trends and Future Directions. (1) VM Placement Strategies: Various strategies have been proposed for VM placement in cloud computing environments. Recent studies have focused on developing intelligent and dynamic placement algorithms. For instance, (Chen et al., 2022) proposed a reinforcement learning-based VM placement approach that considers workload characteristics and resource availability to achieve better placement decisions. Similarly, (Huang et al., 2021) introduced a genetic algorithm-based placement strategy that optimizes resource allocation and minimizes communication overhead. (2) Performance Optimization: Performance optimization is a crucial objective when placing VMs in cloud computing environments. Researchers have explored techniques to enhance VM placement for improved performance metrics such as response time, throughput, and latency. (Zhang et al., 2022) proposed a deep learning-based VM placement framework that leverages historical workload patterns to predict resource demands and optimize placement decisions. Additionally, (Wang et al., 2021) introduced a locality-aware

Corresponding author: Faruku Umar Ambursa

✉ fuambursa.it@buk.edu.ng

Information Technology Department, Bayero University, Kano, Nigeria
© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved



VM placement algorithm that minimizes network congestion and reduces communication latency. (3) Energy Efficiency: Energy efficiency is an important consideration in VM placement to minimize power consumption and carbon footprint in cloud data centers. Recent studies have focused on energy-aware VM placement algorithms. (Li et al., 2023) presented an energy-efficient VM placement approach that considers dynamic workload characteristics and server power consumption to achieve energy savings. Furthermore, (Wu et al., 2022) introduced a heuristic-based algorithm that optimizes VM placement to minimize energy consumption while maintaining performance requirements. (4) Recent Trends and Future Directions: Recent trends in VM placement research include the integration of emerging technologies and the exploration of hybrid cloud environments. (Chen et al., 2023) investigated the adoption of edge computing in VM placement to support low-latency and real-time applications. Moreover, (Liu et al., 2022) discussed the challenges and opportunities of VM placement in hybrid cloud environments, where resources are distributed across multiple clouds and edge devices.

VMP has attracted significant research attention. In the context of service performance and operation expenses, different optimization perspectives have been studied. (Zhao, 2018) presented a power-aware and performance-guaranteed VMP to solve the high-power consumption and performance degradation problem. To achieve order of reductions in energy consumption of the data center, (Liu, 2017) concentrated on minimizing the number of active physical servers, using algorithm based on an ant colony system. Differently, (Meng et al., 2010) proposed a traffic-aware VMP model to improve the network scalability. To guarantee a desirable service level, (Gaggero et al., 2019) took multiple conflicting performance metrics into account and proposed a holistic placement framework which aimed to seek optimal mapping relation between virtual and physical machines. Driven by the highly dynamic and unpredictability in cloud data centers, (Xu et al., 2017) pursued a load balancing VMP and summarize load balancing algorithms in terms of their metrics and performance evaluation.

VMP problem can be formulated as a bin packing problem and is therefore known to be an NP-hard problem. Due to computing complexity, it is difficult to seek optimal solutions within a short time. Therefore, researchers try to solve this problem within a reasonable time by metaheuristic techniques, such as meta-heuristics algorithm, approximate algorithm or heuristics strategy. (Grandl et al., 2014) modeled the resource allocation problem in clusters as a bin packing problem and took a heuristics strategy to work out this problem, named Tetris. To reduce the average job completion time and relieve resource fragmentation, each job that can fit uses score alignment as similarity between vectors of local resources of machine and job. It's convenient for Tetris to obtain a good performance, but not an optimal solution. (Hu et al., 2010) presented an approach using genetic algorithm to obtain best load balancing in the environment of cloud computing. Their goal is to achieve optimal strategy where VMs get the smallest number of migrations to minimize the migration consumption. Genetic-based algorithm can achieve an approximate optimal solution; however, its time complexity is too high to be tolerated under an online setting. (Sharma et al., 2019) solved the resource allocation in virtual machines by Genetic Algorithm and Particle Swarm Optimization. Taking dynamic resource demands into consideration, (Hanet al., 2019) formulated the virtual machine management as a Markov decision process problem. To solve such a problem, an approximate algorithm was proposed. To improve the network scalability, a two-tier approximate algorithm was designed by (Meng et al., 2010) to solve the proposed traffic-aware VMP model. The work by (Rolik et al., 2018) presented a dynamic virtual machines placement based on reinforcement learning in which the power consumption and the number of SLA violations were taken into account. (Zhang et al., 2018) provided an idea of an online auction scheme in terms of resource allocation using a series of combinatorial algorithms to achieve the social welfare maximization. Even though there are some existing works in the literature to address how to optimize the virtual network resources related problems, a problem that has received

Corresponding author: Faruku Umar Ambursa

✉ fuambursa.it@buk.edu.ng

Information Technology Department, Bayero University, Kano, Nigeria
© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved



much less attention is: how to balance the usage of multiple resources to alleviate resource fragmentation, so that it can prevent waste of physical resources and undesirable performance. Furthermore, with the explosive expansion of data centers, how to effectively manage the limited resources still waits for an answer from the academia and the industry, which is crucial to the sustainable development of cloud computing and trouble operators. To this end, we focus on how to balance the usage of multiple resources to alleviate resource fragmentation while maximize the service rate in virtual machine placement problem, so that it can prevent waste of physical resources and undesirable performance. In order to settle such an optimization problem, we propose a joint bin-packing heuristic and genetic algorithm to lower the time complexity, at the same time achieving an approximate optimal solution.

Finally, VMP formulation can be online, offline or in two phases (online and offline) in which the advantages of both online and offline formulations are combined (Pires et al., 2014), offline improves the quality of solutions obtained in online formulation. A work by (Pires et al., 2017) consider the two-phase formulation which is composed of an iVMP phase (Dynamic) that receives dynamic requests for placement which they used FFD algorithm and VMPr phase (Static) which performs static replacement in order to optimize solution of the dynamic phase where the used an evolutionary algorithm called Memetic algorithm. This work focuses of enhancing the offline phase of two-phase virtual machine placement.

MATERIALS AND METHODS

This section describes the methodology for the study, which includes the research design, problem description, previous scheme and the proposed method for the VMP problem. The section also gives the performance metrics, as well as the experimental settings for this research.

Resource scheduling and allocation play a vital role in cloud computing mostly to develop execution efficiency and utilization of resources, energy saving, users QoS requirements satisfaction and increasing the profit

of the cloud providers (Madni et al., 2016). Furthermore, there are no ideal placement algorithms for the problem, which is why the time to find a solution increases exponentially with the problem size (Speitkamp et al., 2010). Nonetheless, given the different objective features and the dynamic nature of cloud computing services, the problem with VMP needs to be more frequently addressed. The VMP formulation can be online, offline or in two phases (online and offline) in which the advantages of both online and offline formulations are combined (Pires et al., 2014), in order to improve the quality of solutions obtained in online formulation.

Virtual machine placement is a significant process used to optimize objective functions such as energy efficiency, resource use, quality of service and overall performance of the cloud services (Masdari et al., 2016). This work focuses on dividing the VMP problem into a two phase-optimization scheme similar to the one proposed by (Pires et al., 2017) for optimization of the following four objective function such as (i) power consumption, (ii) economical revenue, (iii) resource utilization and (iv) reconfiguration time which was combined as a single objective.

Virtual Machine Placement

The previous work (Pires et al., 2017) consider the two-phase formulation which is composed of an iVMP phase (Dynamic) that receives dynamic requests for placement which they used FFD algorithm and VMPr phase (Static) which performs static replacement in order to optimize solution of the dynamic phase, in this phase, the previous work used an evolutionary algorithm called Memetic algorithm. However, the dynamic phase composed of VMPr Triggering method that defines when to trigger the VMPr phase where a prediction-based triggering method called Double Exponential Smoothing was used to trigger the VMPr phase.

After conducting literature review, the problem (mentioned in section 1) was identified, to solve the problem, the previous work have been adopted and compared with the new propose method using the same objective functions that was adopted from the previous work.

The VMP problem can be formulated as

Corresponding author: Faruku Umar Ambursa

✉ fuambursa.it@buk.edu.ng

Information Technology Department, Bayero University, Kano, Nigeria
© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved

online (at run time) and offline (static environment). However, online placement does not always produce an optimal solution and offline solution is performed in a static environment (when the system is not at runtime) which is not suitable considering the dynamic nature of IaaS environment. Therefore, VMP problem can be formulated as a two-phase optimization scheme while combining the advantages of both online

and offline formulations (Calcavecchia et al., 2012; Baloglazov et al., 2012; Pires et al., 2017). In this regard, the two-phase formulation is composed of an iVMP phase (Dynamic) that receives dynamic requests for placement as well as a VMPr phase (Static) which performs static replacement. This work mainly focuses on the VMPr phase (Static).

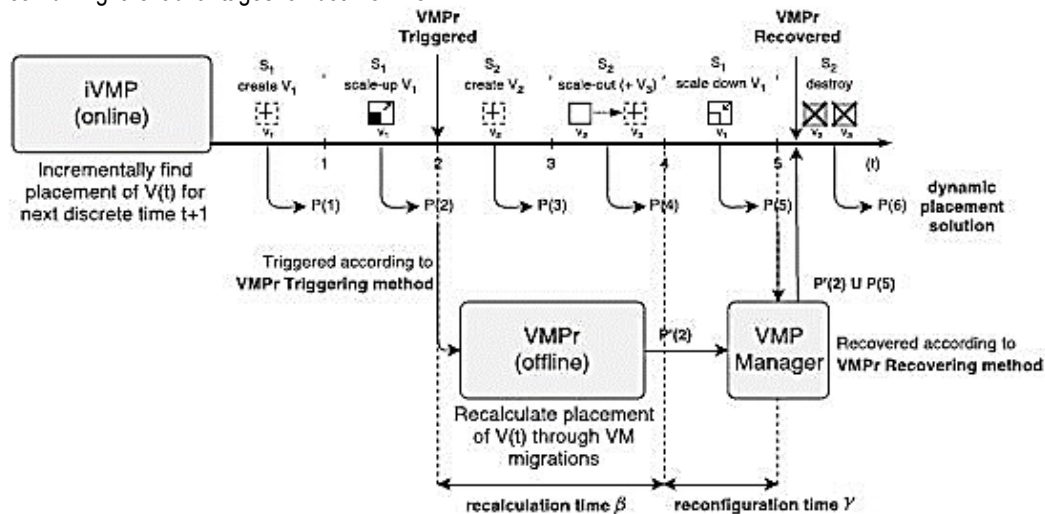


Figure 1: Two-phase optimization scheme for VMP problems (Pires et al., 2017)

Incremental VMP (iVMP) Phase

Incremental Phase (iVMP) consist of dynamic placement and triggering method.

Dyanamic Placement

The incremental iVMP phase proposed in previous schemes receives dynamic requests for placement of VMs to PMs at runtime based on some online solution algorithms (Pires et al., 2017). A First Fit Decreasing (FFD) heuristic was used to solve the VMP problem at each discrete time t where VM requests changes over time, this scheme was adopted from the previous work.

VMPr Triggering

VMPr Triggering method defines when the VMPr phase should be triggered in a two-phase optimization scheme for VMP problems. A prediction-based triggering method used to determine when to trigger the VMP reconfiguration phase was presented by (Pires et al., 2017) based

on Double exponential smoothing technique, this scheme was adopted from the previous work.

VMP reconfiguration (VMPr) phase

This section discusses the reconfiguration phase of the two-phase VMP by explaining the previous work as well as the now propose scheme.

The VMPr phase performs placement reconfiguration considering a static environment, where placements are reconfigured and optimized.

Previous Method

Previous work (Pires et al., 2017) proposed a mechanism based on evolutionary algorithm called Memetic algorithm (MA) to perform placement reconfiguration, and it works as follows: At step 1, a set population of candidate solutions is randomly created. These candidate solutions are repaired at step 2 to ensure that the

Corresponding author: Faruku Umar Ambursa

✉ fuambursa.it@buk.edu.ng

Information Technology Department, Bayero University, Kano, Nigeria
 © 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved

population contains only feasible solutions. Then, the algorithm tries to improve candidate solutions at step 3 using local search. Solutions obtained are then checked and the first best solution $x'(t)$ is selected from the set of population based on the objective function defined. After an initialization in step 5, evolution begins (steps 6–12), where solutions are selected from the union of the evolutionary set of solutions (or population), also known as Pop_u , and the best-known solution $x'(t)$ (step 7), crossover and mutation operators are applied as usual (step 8), and eventually solutions

are repaired, as there may be infeasible solutions (step 9). Improvements of solutions of the evolutionary population Pop_u may be generated at step 10 using local search (local optimization operators). At step 11, the best-known solution $x'(t)$ is updated (if applicable), while at step 12 the generation (or iteration) counter is updated. The evolutionary process is repeated until the algorithm meets a stopping criterion (such as a maximum number of discrete time instants or iterations), returning the best-known solution $x'(t)$ for a placement reconfiguration (see Algorithm 1).

Algorithm 1: Memetic Algorithm (MA) for VMPr (Pires et al., 2017).

Data: $H, U(t), x(t)$
Result: Recalculated Placement $x'(t)$

- 1 initialize set of candidate solutions Pop_0 ;
- 2 Pop_0' = repair infeasible solutions of Pop_0 ;
- 3 Pop_0'' = apply local search to solutions of Pop_0' ;
- 4 $x'(t)$ = select best solution from $Pop_0'' \cup x(t)$ considering (GOF);
- 5 $u = 0$; $Pop_u = Pop_0''$;
- 6 **while** stopping criterion is not satisfied **do**
- 7 Pop_u = selection of solutions from $Pop_u \cup x'(t)$;
- 8 $Pop'u$ = crossover and mutation on solutions of Pop_u ;
- 9 $Pop''u$ = repair infeasible solutions of $Pop'u$;
- 10 $Pop'''u$ = apply local search to solutions of $Pop''u$;
- 11 $x'(t)$ = select best solution from $Pop'''u$, considering (GOF);
- 12 increment number of generations u ;
- 13 **End**
- 14 **return** $x'(t)$

Proposed Tabu Search Algorithm for VMPr

The problem with the previous method as stated by (Natalio, 2006), a population's lack of diversity causes the Memetic Algorithm to allocate repeated reproductive, mutation and refinement trials to the same individuals thus wasting precious time and resources. As a result, we use metaheuristic Tabu Search (TS) algorithm that uses neighborhood optimization method, to solve the issue. The basic steps of the TS algorithm are as follows:

Initialize the current solution: this can be done randomly, generate a set of candidate solutions: This is done by applying a set of neighborhood moves to the current solution. The neighborhood moves can be problem-specific,

such as swapping two elements in a permutation or adjusting the value of a variable in a linear programming problem. 4. Select the next solution: Select the best candidate solution from the set of candidates, based on some quality measure such as the objective function value. 5. Mark the selected solution as "tabu": Add the selected solution to the tabu list, which is a data structure that stores solutions that have been marked as "tabu." The solutions in the tabu list are not allowed to be selected as the next solution for a certain number of iterations. 6. Update the current solution: Set the selected solution as the new current solution. 7. End of the iteration: Check if the stopping criterion is met, if so, return the

current solution. 8. Repeat until the stopping criterion is met.

TS works as follows: the algorithm starts from an initial solution, and then explores iteratively its neighborhood. Thus, the principal element that constitutes the TS methodology is neighborhood $N(S)$ of an incumbent solution S .

Indeed, each solution S' in the neighborhood $N(S)$ is reached by a simple perturbation of the solution S . The neighborhood $N(S)$ can contain solutions that deteriorate the quality of the current solution S . At each iteration, the best solution in $N(S) \setminus \{S\}$ is considered to continue the search.

Algorithm 2: Tabu Search (TS) for VMP_r phase.

```

1  Data:  $H, U(t), x(t)$ 
2  Result: Recalculated Placement  $x'(t)$ 
3  initialize set of candidate solutions  $Pop_0$ ;
4   $x'(t) =$  select solution from  $Pop_0$  considering ( $GOF$ )
5   $x'(t) =$  bestSol;
6  tabuList = []
7  add  $x'(t)$  to tabuList
8  while stopping criterion is not satisfied do
9     $N_s =$  neighbor of  $x'(t)$  in  $Pop_0$ 
10   if ( $N_s$  not in tabuList)
11     add  $N_s$  to tabuList
12     if ( $N_s > x'(t)$ )
13        $x'(t) = N_s$  is new best solution considering ( $GOF$ );
14   End
15 End
16 iteration += 1;
17 End
18 return  $x'(t)$ ;

```

Moreover, as state by (Anass et al., 2016), the strength of the TS resides in its capacity to escape the trap of local optimality, and therefore gives good solutions of hard combinatorial optimization problems.

Objective Functions

In this work, the four major objective function proposed by (Pires et al., 2015) such as (i) power consumption, (ii) economical revenue, (iii) resource utilization and (iv) reconfiguration time will be considered. Each objective function cost, and the four functions were merged as a single context of optimization, considering the minimization approach. In this section, the steps in arriving at the objective function are provided. These steps are adopted and modified from the previous work.

Aggregate Cost Function (ACF)

The presented normalized objective functions are combined into a single objective considering a minimum Euclidean distance to the origin as defined in equation (6)

$$F(x, t) = \sqrt{\sum_{i=1}^q \hat{f}_i(x, t)^2} \quad (6)$$

Where:

$F(x, t)$ denoted a single objective function combining each $\hat{f}_i(x, t)$ at instant t , $\hat{f}_i(x, t)$ represents the normalized cost of the objective function $f_i(x, t)$ at instant t and q indicates the number of objective functions (in this case $q = 4$).

Moreover, like in previous works, this work also considers the uncertainty of the evaluation parameters: (i) virtual resources (ii) number of VMs that compose cloud services (iii) utilization of CPU and RAM memory virtual resources and (iv) utilization of networking virtual resources due to randomness of customer

Corresponding author: Faruku Umar Ambursa

✉ fuambursa.it@buk.edu.ng

Information Technology Department, Bayero University, Kano, Nigeria
 © 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved

requests and generally the dynamic mature of cloud environments.

Uncertainty is usually modeled through a set of scenarios S , where the uncertain parameters are considered. For each scenarios $\in S$, a temporal average value of the objective function $F(x, t)$ is defined in equation (7)

$$\overline{f_s(x, t)} = \frac{\sum_{t=1}^{t_{max}} F(x, t)}{t_{max}} \quad (7)$$

Where:

$\overline{f_s(x, t)}$: Temporal average of combined objective function for all discrete time instant t in scenario $s \in S$;

t_{max} : Duration of a scenario in discrete time instants.

Finally, in this work, a solution from a set of candidate solutions is selected using the average criteria (Aloulou& Croce, 2008), which is defined as the average $\overline{f_s(x, t)}$ for all scenarios $s \in S$ as defined in equation (8)

$$A_{criteria} = \overline{F(x, t)} = \frac{\sum_{s=1}^{|S|} \overline{f_s(x, t)}}{|S|} \quad (8)$$

Where:

$A_{criteria}$: denotes the average cost function criteria.

The overall objective of the optimization is to minimize the average cost function.

PERFORMANCE EVALUATION

In this section, performance of the proposed method is evaluated and compared with the benchmark. Equation 8 (Average values of $\overline{f_s(x, t)}$) is used as the evaluation criteria. The lower the value of the cost function the better the performance.

SIMULATION EXPERIMENT

Computer Resources

The Experiments was conducted on a GNU Linux Operating System with an Intel^(R) dual Core™ i7-4600U CPU@ 2.90 GHZ 8 GB of RAM memory. The evaluated algorithms implemented

in Java programming language using netbeans IDE.

Experimental Setup

In this work, the assumptions and experimental settings, parameters and scenario have been adopted from the benchmark work (Pires et al., 2017). The following parameters of the uncertain VMP formulation would be considered for the experimental evaluation.

1. Number of considered resources: $r = 3$;
2. Recalculation time $\beta = 2$
3. Protection factor for each resource k : $\lambda k = 0.5$;
4. Penalty factor for each resource k : $\phi k = 1$.

As input data, available PMs include 4 different types (H.small, H.medium, H.large, H.xlarge) of PMs was considered (see table 3.1). However, Considering the available PM types, five laaS datacenter with different number of PMs would be considered (DC1 to DC5) depicted in table 3.2. Additionally, 80 different workload traces of requested cloud services $V(t)$ and their specifications would be considered as input data as well as their utilization of resources $U(t)$ at each discrete time t (see eqn (2.3)). Requested VMs would be considered according to instance types offered by Amazon Elastic Compute Cloud (EC2) depicted in table 3.3, nonetheless, it is important to remember that the following uncertain parameters would be considered (i) virtual resources capacities (vertical elasticity), (ii) number of VMs that compose cloud services (horizontal elasticity), (iii) utilization of CPU and RAM memory virtual resources and (iv) utilization of networking virtual resources (both relevant for overbooking). Therefore, two different Probability Distribution Functions (PDFs) would be considered to represent each parameter behavior (i.e., Uniform and Poisson). Workload traces of cloud service requests were generated using a Cloud Workload Trace Generator (CWTG) for provider-oriented VMP problems (Ortigoza et al., 2016).



Table 1: Types of PMs considered for the simulation

PM type	Pr1,i [ECU]	Pr2,i [GB]	Pr3,i [Mbps]	pmaxi	ci
H.small	32	128	1000	800	1
H.medium	64	256	1000	1000	1
H.large	256	512	1000	3000	1
H.xlarge	512	1024	20000	5000	1

Table 2: Number of PMs per type on each considered IaaS cloud datacenter.

PM type	Datacenter (DC1)	Datacenter (DC2)	Datacenter (DC3)	Datacenter (DC4)	Datacenter (DC5)
H.small	50	30	20	15	10
H.medium	50	30	20	10	10
H.large	50	30	15	10	9
H.xlarge	30	10	8	10	8

Table 3: Virtual Machine (VM) types from Amazon EC2 considered for cloud service requests in simulations.

ID Instance	Type	Vr1,j [ECU]	Vr2,j [GB]	Vr3,j [Mbps]	Rj [\$]
0	m4.large	6.5	8	450	0.12
1	c4.large	8	3.75	500	0.10
2	m4.xlarge	13	16	750	0.239
3	c4.xlarge	16	7.5	750	0.209
4	m4.2xlarge	26	32	1000	0.479
5	c4.2xlarge	31	15	1000	0.419
6	m4.4xlarge	53.5	64	2000	0.958
7	c4.4xlarge	62	30	2000	0.838
8	m4.10xlarge	124.5	160	4000	2.394
9	c4.8xlarge	132	60	4000	1.675
10	x1.32xlarge	349	1952	10000	13.338

RESULTS PRESENTATION AND DISCUSSION

This section presents the results obtained from an experiment on the VMPr reconfiguration method, along with a detailed discussion and a comparison with a benchmark study. The results are presented in a logical manner, providing a clear understanding of the performance of the VMPr reconfiguration method. Overall, this chapter is a valuable resource for researchers and practitioners interested in the VMPr reconfiguration method, providing a comprehensive analysis.

Results Presentation

In this subsection, we present the outcomes of the simulation experiment, which involved considering five IaaS data centers (DC1, DC2, DC3, DC4, and DC5) and 80 distinct workloads. Consequently, we evaluated a total of 400 scenarios (80 per each DC) during the experiments. The results obtained from the simulation are detailed below:

The comparison between the benchmark work and the proposed work results for Data Center 1 (DC1) is illustrated in Figure 4.1 below:

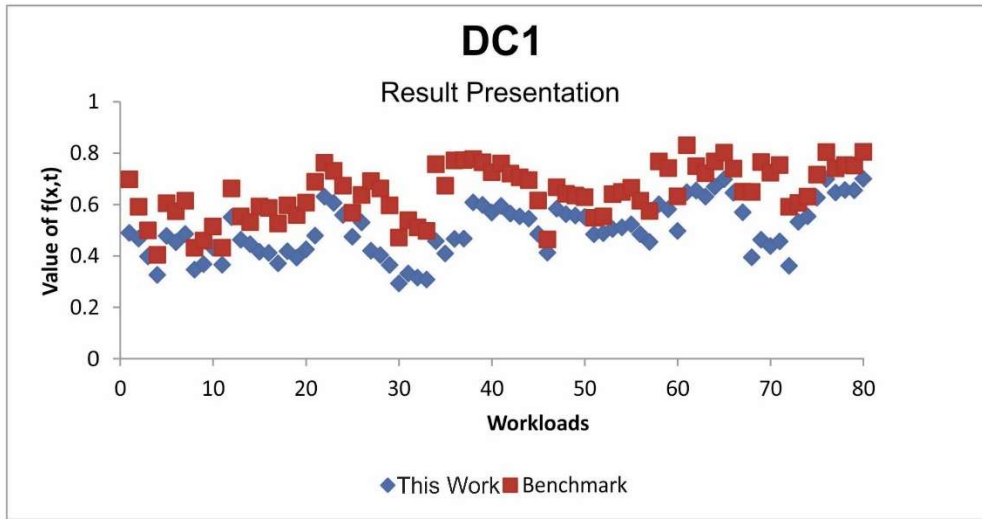


Figure 2: Average values of $F_s(x, t)$ in DC 1

The proposed work has average of 0.5007 while the benchmark obtains average of 0.6475, the proposed work out perform with difference of 0.1468 that is 14.68%.

Likewise, Figure 4.2 presents a comparison between the proposed work result for Data Center 2 (DC2) and the benchmark work.

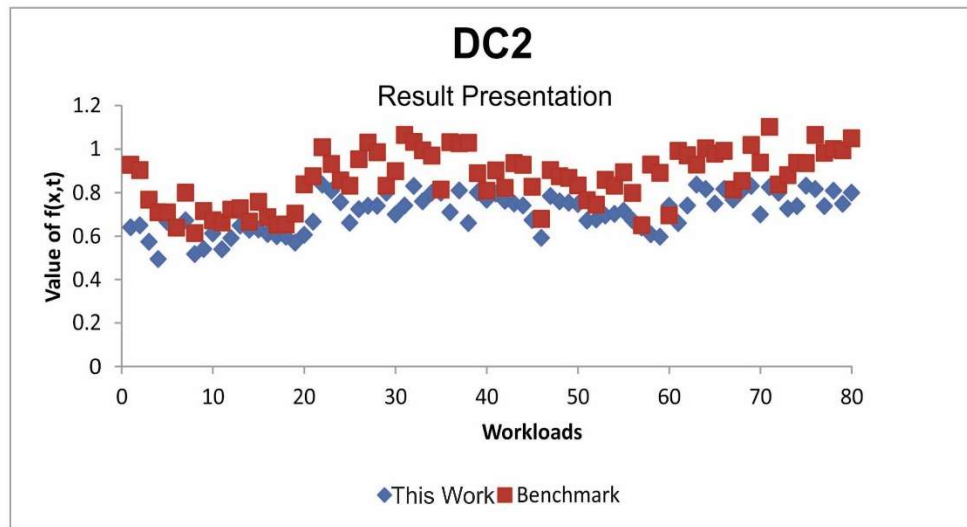


Figure 3: Average values of $F_s(x, t)$ in DC 2.

The graph shows that the proposed work has average of 0.7095 while the benchmark obtains average of 0.8660, the proposed work achieve better result with difference of 0.1568 that

is 15.68%. Furthermore, Figure 4.3 displays the proposed work result for Data Center 3 (DC3) in comparison to the benchmark work.

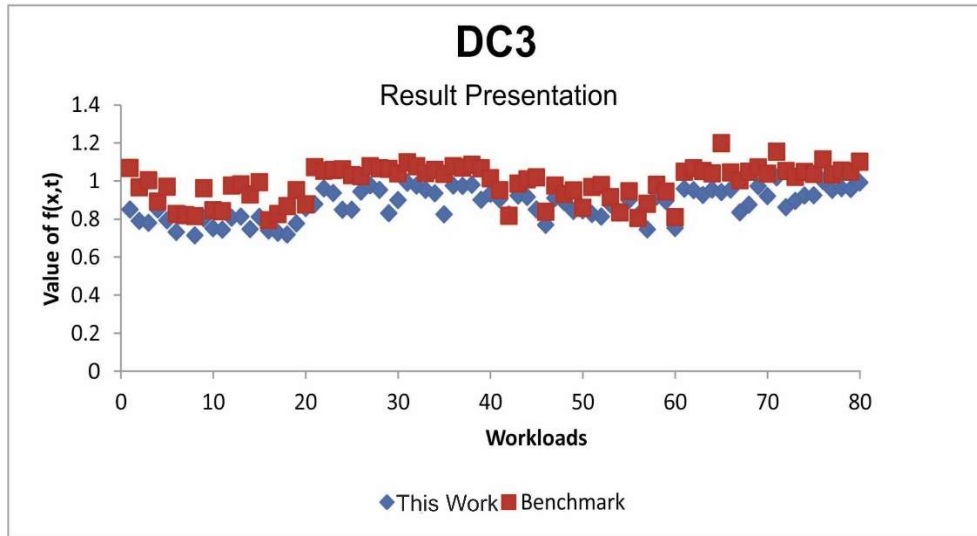


Figure 4: Average values of $F_s(x, t)$ in DC 3.

The data presented in the graph shows that the proposed work has average of 0.8751 while the benchmark obtains average of 0.9887, the proposed give better result with difference of

0.1136 that is 11.36%. Moreover, Figure 4.4 illustrates a comparison between the proposed work result for Data Center 4 (DC4) and the benchmark work.

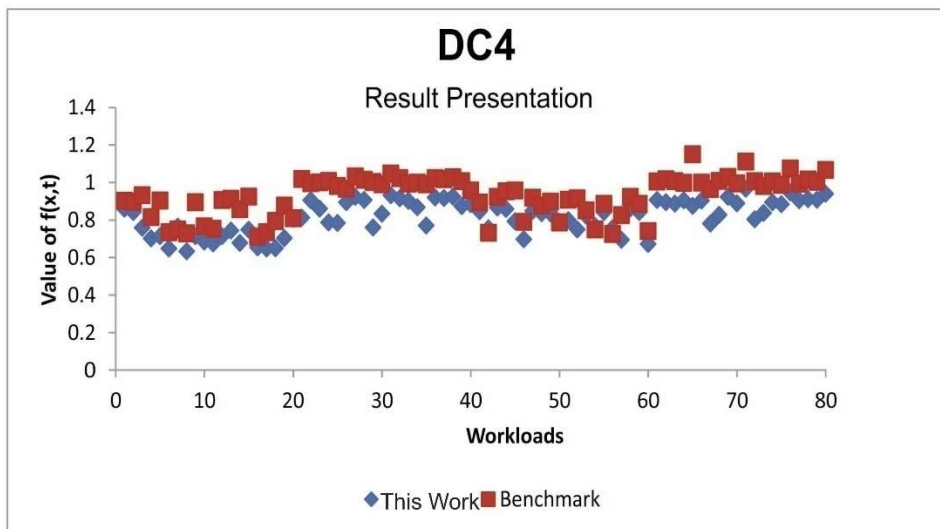


Figure 5: Average values of $F_s(x, t)$ in DC 4

The graph clearly shows that the proposed work has average of 0. 8186 while the benchmark obtains average of 0. 9289, the proposed work performs better with difference of

0.1103 that is 11.03%. Similarly, Figure 4.5 presents a comparison between the proposed work result for Data Center 5 (DC5) and the benchmark work.

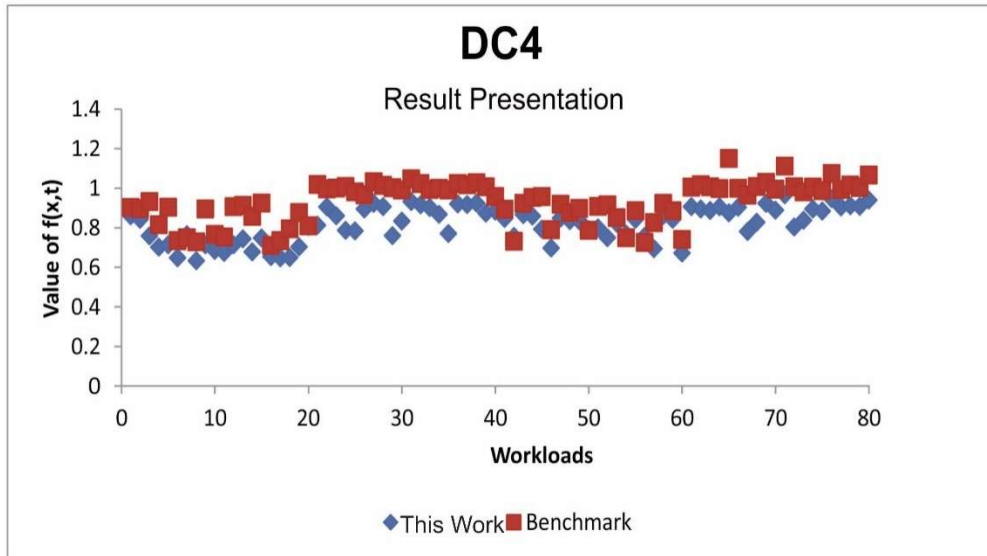


Figure 6: average values of $F_s(x, t)$ in DC 5

It can be seen in the above graph that the proposed work has average of 0.7507 while the benchmark obtains average of 0.9398, the proposed work out perform with difference of 0.1891 that is 18.91%.

In conclusion, Figure 4.6 depicts the normalized objective Temporal Average values of $F_s(x, t)$ across all five data centers, DC1 to DC5. The Temporal Average values represent the average combined objectives, i.e., $F_s(x, t)$ in DC1 to DC5, for each scenario $s \in S$.

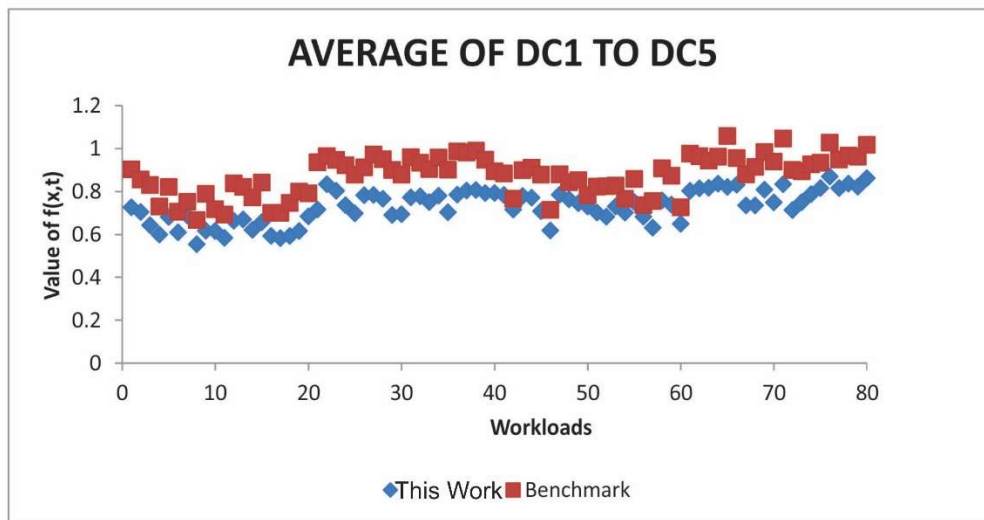


Figure 7: Average values of $F_s(x, t)$ in DC 1 to DC 5 per each Scenario $s \in S$

Moreover, Table 4.1 shows the summary of the average values of the result

obtained in experiments performed for the five considered data centers (i.e DC 1 to DC 5) per

Corresponding author: Faruku Umar Ambursa

fuambursa.it@buk.edu.ng

Information Technology Department, Bayero University, Kano, Nigeria

© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved



each Scenario $s \in S$, the values in Table 4.1 shows the average cost of each DC.

Table 4: Average values of results obtained in the performed experiment.

	DC 1	DC 2	DC 3	DC 4	DC 5	AVG	Ranking
This Work	0.5007	0.7095	0.8751	0.8186	0.7507	0.7310	1st
Bench work (Pires et al,2017)	0.6475	0.8660	0.9887	0.9289	0.9398	0.8740	2nd

RESULTS DISCUSSION

This research work was aimed to optimize result obtained at VMPr phase in the two-phase optimization of Virtual Machine Placement Problems. $F(x, t)$ Minimization cost function was the mean evaluation metric used which is a composite of Power usage, Economic Revenue, Resources usage and Reconfiguration period functions.

Table 4.1 shows the summary of the evaluation results in terms of numerical average values attained by each method at the end of each experiment.

Considering the table 4.1 presented above, results obtained in this work shows that the proposed neighborhood technique using Tabu Search outperform the existing benchmark strategy in the experiment conducted and the results achieve more optimized minimal cost function in the scenarios considered. Thus this research provides more optimized better result than the benchmark research results.

Moreover, considering the results obtained in figure 4.6, this research work achieved better results (i.e Minimum cost) for the average values $F_s(x, t)$ in the total scenario considered for the five data centers (DC 1 to DC 5). To summarize, this work obtained a better result (Minimum cost) with the average value of 0.7310 while the benchmark work achieves the average value of 0.8740.

Therefore, this work outperforms the benchmark work considering the minimization approach because of the Neighborhood Techniques Proposed which provide more optimized outcome, and lead to more optimal solution at the reconfiguration phase of the two-phase optimization scheme.

CONCLUSION

This research work proposed a new technique to optimize VMPr reconfiguration phase in two-phase method for the Virtual Machine Placement problem that consist of an offline (VMPr) phase which is Static environment for placement recalculation and reconfiguration, and an online (iVMP) phase that receives dynamic request from the client at runtime, Moreover, the research work proposed a new technique to optimize VMPr reconfiguration phase using neighborhood structured algorithms called Tabu search algorithms.

The aim of this research was to optimize the result obtained at the VMPr reconfiguration phase in the two-phase optimization of Virtual Machine Placement Problems. To evaluate the performance of the optimization method, the $F(x, t)$, minimization cost function was used as the mean evaluation metric, which is a composite of Power usage, Economic Revenue, Resources usage, and Reconfiguration period functions. The results of the five data centers were averaged under each scenario, and the figure demonstrates the behavior and performance of the $F_s(x, t)$ Temporal Average Cost on Neighborhood Strategy Metaheuristic Tabu Search algorithms compared to Evolutionary Memetic Algorithm in each experiment. The results indicate that the proposed neighborhood structured method using Tabu Search outperforms the existing benchmark strategy in the conducted experiment, achieving a better minimal cost function in the scenarios considered. Specifically, the proposed solution obtained more optimized result (Minimum cost) with the average value of 0.7310 while the benchmark work achieved the average value of 0.8740. Therefore, this work outperforms the benchmark work considering the minimization approach because of the Neighborhood

Corresponding author: Faruku Umar Ambursa

✉ fuambursa.it@buk.edu.ng

Information Technology Department, Bayero University, Kano, Nigeria
 © 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved



Techniques Proposed, which provides more optimized outcome, leading to a more optimal solution at the reconfiguration phase of the two-phase optimization scheme.

REFERENCES

- Anass, E. ., Abdellah E., Mohammad C., Mohamed R.(2016). *Tabu Search and Memetic Algorithms for a Real Scheduling and Routing Problem*, *Logistics Research* (2017) 10:7 DO I 10.23773/2017_7.
- Baloglazov, A., Abawajy, J., & Buyya, R. (2012). *Energy aware resource allocation heuristics for efficient management of datacenters for cloud Computing*. *Future Gener. Comp. Syst.*, 755-768.
- Calcavecchia, N. M., Biran, O., Hadad, E., & Moatti, Y. (2012). *VM placement strategies for cloud scenarios in cloud computing (CLOUD)*, 2012 IEEE 5th International Conference on, IEEE, 2012, 852–859.
- Chen, L., Li, X., Liu, Y., & Zhang, Y. (2023). *A Dynamic Load Balancing Algorithm Based on Workload Characteristics for Cloud Computing*. *Journal of Cloud Computing*, 12(1), 1-18.
- Chen, S., Liu, J., Li, X., & Zhang, H. (2022). *Reinforcement Learning-Based Virtual Machine Placement for Cloud Computing*. *IEEE Transactions on Parallel and Distributed Systems*, 33(6), 1522-1536.
- Chen, X., Wang, L., Li, X., & Li, J. (2023). *Edge-Assisted Virtual Machine Placement for Latency-Sensitive Applications in Cloud Computing*. *Journal of Network and Computer Applications*, 188, 103077.
- Chen, Y., Shi, Y., Gong, L., & Cai, Y. (2018). *Performance and Reliability Challenges in Cloud Computing*. In 2018 IEEE International Conference on Services Computing (SCC) (pp. 294-297). IEEE.
- Gaggero, M., & Caviglione, L. (2019). *Holistic Placement of Virtual Machines in Cloud Data Centers*. In 2019 IEEE International Conference on Cloud Computing Technology and Science (CloudCom) (pp. 327-333). IEEE.
- Goudarzi, H., Pedram, M. S., & Pedram, M. (2012). *Dynamic consolidation of virtual machines in cloud data centers using online and offline algorithms*. *IEEE Transactions on Cloud Computing*, 1(2), 145-158.
- Gupta, A., Milojevic, D., & Kal'e, L. V. (2012). *Optimizing VM placement for hpc in cloud* in proceedings of the 2012 workshop on Cloud Services, federation and 8th open cirrus summit, ACM, (pp. 1-6).
- Grandl, R., Ananthanarayanan, G., Kandula, S., & Rao, S. (2014). *Scaling Datacenter Networks to 100Gbps and Beyond with Zero Packet Loss*. In Proceedings of the 11th USENIX Symposium on Networked Systems Design and Implementation (NSDI) (pp. 361-374).
- Han Z, Tan H, Wang R, Chen G, Li Y, Lau FCM. *Energy-efficient dynamic virtual machine management in data centers*. *IEEE/ACM Trans Network* 2019;27(1): 344–60.
- Huang, X., Sun, Y., Luo, Y., & Wang, S. (2021). *Genetic Algorithm-Based Virtual Machine Placement Optimization in Cloud Computing*. *Future Generation Computer Systems*, 116, 1-14.
- Huang, Q., Li, H., Zhang, J., & Wu, Q. (2022). *Scalable Storage Architecture for Cloud Systems*. *IEEE Transactions on Parallel and Distributed Systems*, 34(2), 564-577.
- Hu, Z., Gu, J., & Sun, J. (2010). *A genetic algorithm based load balancing strategy in cloud computing*. In Proceedings of the 2010 International Conference on Computational Intelligence and Software Engineering (pp. 1-4). IEEE.
- Jula, A., Pop, F., & Prodan, R. (2014). *A Survey on Resource Allocation in Cloud Computing: Taxonomy, Challenges, and Solutions*. *Journal of Grid Computing*, 12(2), 335-377.
- Kumbhare, A., Malani, I., & Anand, S. (2018). *Vendor Lock-in Issues in Cloud Computing*. In Proceedings of the 2018 2nd International Conference on Communication System, Computing and IT Applications (pp. 1-5). IEEE.
- Lago, P., Garg, S. K., & Buyya, R. (2011). *Job Scheduling and Resource Provisioning for Cloud Computing*. *Future Generation Computer Systems*, 28(5), 684-692.
- Li, J., Zhao, L., Li, J., Wang, Y., & Wang, S. (2013). *Resource utilization optimization in cloud*

Corresponding author: Faruku Umar Ambursa

✉ fuambursa.it@buk.edu.ng

Information Technology Department, Bayero University, Kano, Nigeria
© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved



- computing based on cooperative game. In 2013 IEEE International Conference on High Performance Computing and Communications & 2013 IEEE International Conference on Embedded and Ubiquitous Computing (HPCC_EUC) (pp. 1259-1264). IEEE.
- Li, J., Jiang, X., Guo, M., & Zhang, T. (2021). *A Privacy-Preserving Machine Learning Framework for Cloud Environments*. *Future Generation Computer Systems*, 128, 52-64.
- Li, S., Xu, C., Li, S., & Zhang, L. (2023). *Energy-Efficient Virtual Machine Placement in Cloud Computing*. *IEEE Transactions on Sustainable Computing*, 8(1), 1-14.
- Liu, Y., Cai, Y., & Zhang, C. (2017). *Ant Colony Optimization for Minimizing the Number of Active Physical Servers in Cloud Data Centers*. In 2017 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM) (pp. 448-453). IEEE.
- Liu, Z., Chen, S., Xu, L., & Huang, S. (2022). *Virtual Machine Placement in Hybrid Cloud Environments: Challenges and Opportunities*. *ACM Computing Surveys*, 55(2), 1-27.
- Liu, Z., Chen, S., Xu, L., & Huang, S. (2022). *Integrating Edge Computing with Cloud Platforms: A Comprehensive Survey*. *ACM Computing Surveys*, 55(1), 1-37.
- Madni, S. H., & Lattif, A. A. (2016). *Resource scheduling and allocation techniques in cloud computing: A comprehensive review*. *International Journal of Computer Applications*, 146(5), 39-45.
- Mark, L., Lee, C., Yang, C., & Zhang, J. (2011). *Economic revenue optimization in cloud computing: A revenue-sharing model*. In *Proceedings of the 2011 IEEE International Conference on Cloud Computing (CLOUD)* (pp. 196-203). IEEE.
- Mary, S. (2014). *Quality of Service in Cloud Computing: A Systematic Review*. *International Journal of Computer Applications*, 101(1), 17-23.
- Masdari, M., Nabavi, S. S., & Ahmadi, V. (2016). *An Overview of virtual machine placement schemes In Cloud Computing*. *Journal of Network and Computer Applications*.
- Mell, P., & Grance, T. (2011). *The NIST Definition of Cloud Computing*. National Standards and Technology, Special Publication 800 – 145.
- Meng X, Pappas V, Zhang L. *Improving the scalability of data center networks with traffic-aware virtual machine placement*. *Proceedings of the IEEE INFOCOM*. 2010 Proceedings IEEE INFOCOM; 2010. p. 1–9.
- Marinos, A., & Malek, M. (2016). *Security, Privacy and Trust in Cloud Systems: A Literature Review*. *Future Generation Computer Systems*, 56, 186-207.
- Natalio, Krasnogor & Aragón, Alberto & Pacheco, Joaquín. (2006). *Memetic Algorithms*. 10.1007/0-387-33416-5_11.
- Ortigoza, J., Pires, F., & Baran, B. (2016). *Workload generation for virtual machine placement in cloud computing environments*. *XLII Latin American Computing Conference, CLEI*, 1-9.
- Pires, F. L., & Baran, B. (2014). *Virtual machine placement literature review (data)*. *Polytechnic School, National University of Asuncion on, Tech. Rep*.
- Pires, F. L., & Baran, B. (2015). *A Virtual Machine Placement Taxonomy*. *International Symposium on Cluster, Cloud and Grid Computing*, 15th IEEE/ACM, 159-168.
- Pires, F. L., Baran, B., Benitez, L., Zalinmben, S., & Amarilla, A. (2017). *Virtual machine placement for elastic infrastructures in overbooked cloud computing datacenters under uncertainty*. *Future Generation Computer Systems*.
- Rahman, A. A., Lashkarara, F., Baida, R., & Yassine, A. (2020). *Compliance Challenges in Cloud Computing Environments: A Systematic Literature Review*. *IEEE Access*, 8, 150556-150573.
- RolikO, ZharikovE, KovalA, TelenyK. *Dynamic management of data center resources using reinforcement learning*. *Proceedings of the 14th international conference on advanced trends in*

Corresponding author: Faruku Umar Ambursa

✉ fuambursa.it@buk.edu.ng

Information Technology Department, Bayero University, Kano, Nigeria
© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved



- radioelectronics, telecommunications and computer engineering (TCSET). 2018 14th International Conference on Advanced Trends in Radio electronics, Telecommunications and Computer Engineering (TCSET); 2018. p. 237–44.
- Sharma NK, Reddy GRM. *Multi-objective energy efficient virtual machines allocation at the cloud data center*. IEEE Trans Serv Comput 2019; 12(1):158–71.
- Smith, A., Johnson, B., & Brown, C. (2021). *Hybrid Cloud Models: Integrating Edge Computing for Real-Time Applications*. International Journal of Information Management, 61, 102263.
- Speitkamp, B., & Bichler, M. (2010). *A mathematical programming approach for server*. IEEE Trans. Serv. Comput, 266–278.
- Uhlig, V. (2006). *Intel Virtualization Technology*. Intel Technology Journal, 10(3), 167-178.
- Wang, L., von Laszewski, G., Younge, A. et al. *New Gener. Comput.* (2010) 28: 137.
- Wang, H., Li, H., Lian, Q., & Lin, C. (2021). *Locality-Aware Virtual Machine Placement Algorithm for Reducing Network Latency in Cloud Computing*. Journal of Supercomputing, 77(9), 8396-8416.
- Wang, Q., Li, S., & Xu, C. (2023). *Serverless Computing in Cloud Environments: Architecture, Challenges, and Future Directions*. Future Internet, 15(1), 1-20.
- Wu, H., Zhang, S., Liu, M., & Li, K. (2022). *Energy-Aware Virtual Machine Placement in Cloud Data Centers*. Journal of Parallel and Distributed Computing, 162, 80-91.
- Xu, Y., Tian, Y., & Buyya, R. (2017). *A survey on load balancing algorithms for virtual machines placement in cloud computing environments*. Concurrency and Computation: Practice and Experience, 29(12), e3959.
- Yang, M., Wang, W., J., & Zhang, H. (2022). *Multi-Cloud Resource Allocation Based on Machine Learning for Performance Optimization*. Journal of Network and Computer Applications, 194, 103173.
- Zhang, J., Li, M., Zhou, L., & Zhou, Y. (2022). *Deep Learning-Based Virtual Machine Placement for Performance Optimization in Cloud Computing*. Future Generation Computer Systems, 128, 538-549.
- Zhang, X., Zhang, Y., Chen, X., Liu, K., Huang, G., & Zhan, J. (2013). *A relationship-based VM placement framework of cloud environment*. in proceedings of the 2013 IEEE 37th Annual Computer Software and Applications Conference 124-133.
- Zhang, Q., Wu, S., Li, X., & Lau, F. (2018). *Online Auction-Based Resource Allocation for Social Welfare Maximization in Cloud Computing*. IEEE Transactions on Services Computing, 11(1), 134-147.
- Zhang, Y., Xiong, S., Zhou, Y., & Li, X. (2022). *A Secure Data Sharing Scheme with Attribute-Based Encryption in Cloud Computing*. Future Generation Computer Systems, 125, 551-561.
- Zhao H, Wang J, Liu F, Wang Q, Zhang W, Zheng Q. *Power-aware and performance-guaranteed virtual machine placement in the cloud*. IEEE Trans Parallel Distributed Syst.2018;29(6):1385–400.