Journal of **N**onlinear **A**nalysis and **O**ptimization
Vol. 16, Issue. 1: 2025
ISSN : **1906-9685**

Using the K-Means Algorithm to Optimize Virtual Machine Location in the Cloud to Lower Communication Cost and Image Retrieval Costs

Mr. Shaik Himam Basha[1] ,Appala Lakshmi Sri Likitha Sivani [2]

#1. Assistant Professor, #2. Pursuing MCA

Department of Master of Computer Applications

QIS COLLEGE OF ENGINEERING AND TECHNOLOGY

Vengamukkalapalem (V), Ongole, Prakasam dist, Andhra Pradesh- 523272

Abstract: The project introduces a heuristic-based algorithm aimed at optimizing virtual machine (VM) placement in cloud data centers, focusing on reducing image retrieval and communication costs. It features three innovative algorithms: PM Clustering, VM Partitioning, and VM-PM Mapping. PM Clustering minimizes overall traffic between clusters by identifying the longest communication distance between physical machines (PMs), effectively grouping them to enhance communication efficiency. Complementing this, VM Partitioning utilizes coarsened allocation to group similar VMs, which improves retrieval efficiency and minimizes inter-VM communication. Additionally, the project integrates an extension using KMEANS clustering to refine PM arrangement by considering various factors, including distance, retrieval, and communication costs. This comprehensive approach not only streamlines VM placement but also optimizes resource allocation in cloud environments, ultimately leading to enhanced performance and reduced operational costs. The proposed methods collectively contribute to a more efficient and cost-effective cloud infrastructure.

"Index Terms *- Virtual Machine Placement, Cloud Data Centers, PM Clustering, VM Partitioning, VM-PM Mapping, Resource Optimization, Image Retrieval, Communication Cost, Network Efficiency, Workload-Aware Allocation."*

1. INTRODUCTION

Infrastructure as a Service (IaaS) has emerged as a key component of cloud computing, enabling users to deploy applications through virtual machines (VMs) that provide essential computing resources, including CPU, memory, and storage [6]. However, large-scale VM deployment often introduces

significant bottlenecks, particularly in backend storage systems, where retrieving VM image files (VMIs) can create substantial network congestion and degrade performance [14], [19]. The increasing demand for efficient VM image retrieval has driven research efforts to optimize VM placement strategies, ensuring faster provisioning and reduced resource contention [12], [18].

Beyond image retrieval challenges, frequent communication between VMs within a data center can further strain the network, increasing latency and reducing overall efficiency [5], [23]. To address these issues, existing studies have explored various optimization techniques for VM placement, including traffic-aware and energy-efficient approaches [1], [4], [24]. Many of these strategies leverage heuristic and meta-heuristic algorithms to improve VM allocation while minimizing power consumption and communication costs [2], [5].

This study proposes a heuristic-based VM placement algorithm aimed at optimizing both VMI retrieval and inter-VM communication within a fat-tree network topology. The solution follows a three-phase approach: (i) PM clustering, which groups physical machines (PMs) based on resource availability and network locality, (ii) VM partitioning, which classifies VMs based on workload characteristics to enhance co-location benefits, and (iii) VM-PM mapping, which assigns VMs to PMs in a manner that balances computational load and minimizes network congestion. By integrating these phases, the proposed approach enhances network efficiency, reduces retrieval delays, and optimizes resource utilization within cloud data centers.

2. RELATED WORK

Virtual machine (VM) placement is a critical aspect of cloud computing that directly impacts resource utilization, energy efficiency, and network performance. Various approaches have been explored to optimize VM placement strategies, focusing on factors such as image retrieval efficiency, inter-VM communication, and overall system scalability.

Several studies have investigated energy-efficient VM placement to reduce power consumption in cloud data centers. Parvizi and Rezvani [4] proposed a utilization-aware VM placement strategy using the NSGA-III meta-heuristic approach to balance workload distribution and minimize energy consumption. Similarly, Wei et al. [5] introduced an improved ant colony optimization algorithm to enhance energy efficiency while ensuring optimal VM allocation in data center networks. These methods aim to reduce power usage without significantly affecting performance.

Another line of research focuses on communication-aware VM placement, which aims to minimize the network overhead caused by inter-VM traffic. Farzai et al. [3] developed a multi-objective optimization model that considers communication costs when placing VMs across data center servers. Similarly, Meng et al. [23] proposed a traffic-aware VM placement strategy that enhances the scalability of data center networks by optimizing VM locations based on network topology constraints. Zhao et al. [24] extended this idea by jointly optimizing VM placement and network topology to improve traffic scalability in dynamic cloud environments.

Efficient VM image retrieval and storage optimization have also been studied extensively. Zhang et al. [19] explored cooperative VM image caching to speed up VM startup times, reducing the burden on storage systems. Other approaches, such as those by Li et al. [17] and Yang et al. [20], leverage content similarity among VM images to

optimize storage and retrieval processes, thereby improving provisioning speeds. In a similar vein, Xu et al. [14] proposed a rethinking of VM image storage strategies to enhance retrieval efficiency in large-scale cloud deployments.

Furthermore, research on network-aware VM migration and placement has addressed the challenges of live VM migrations in geo-distributed cloud environments. Al-Kiswany et al. [21] introduced VMFlock, a co-migration technique that optimizes VM movements while reducing migration costs. Darrous et al. [15] proposed Nitro, a network-aware VM image management system designed to improve VM provisioning across distributed cloud infrastructures. Addya et al. [22] presented a power and time-aware VM migration framework for multi-tier applications, optimizing migration decisions based on resource availability.

Despite these advancements, existing approaches often focus on either energy efficiency, communication optimization, or image retrieval but fail to integrate all these factors into a unified framework. This study aims to address this gap by developing a heuristic-based VM placement algorithm that simultaneously optimizes image retrieval efficiency, communication costs, and network resource utilization within a fat-tree data center topology.

3. MATERIALS AND METHODS

The proposed system aims to optimize the placement of virtual machines (VMs) in cloud data centers by implementing three innovative algorithms: PM Clustering, VM Partitioning, and VM-PM Mapping. These algorithms work synergistically to enhance resource allocation efficiency while minimizing costs related to image retrieval and communication overhead.

PM Clustering organizes physical machines (PMs) into groups based on resource utilization, enabling effective management of computational resources. Several studies have explored resource-aware clustering techniques to improve VM allocation and reduce power consumption in data centers [4], [5]. Parvizi and Rezvani [4] introduced a utilization-aware VM placement strategy, which aligns with the objective of PM Clustering in optimizing resource allocation.

VM Partitioning further divides VMs based on their workload characteristics to ensure that resources are allocated where they are most needed. Prior research has demonstrated that workload-aware VM placement improves efficiency in data-intensive applications [2], [3]. Farzai et al. [3] proposed a multi-objective optimization approach that considers communication and workload distribution when placing VMs, highlighting the importance of partitioning based on application characteristics.

VM-PM Mapping strategically assigns VMs to physical machines in a manner that optimizes data center performance. This approach takes inspiration from traffic-aware VM placement methods, such as those proposed by Meng et al. [23] and Zhao et al. [24], which focus on reducing inter-VM communication costs and enhancing network scalability. Additionally, Zhang et al. [19] demonstrated that cooperative VM image caching can significantly improve retrieval times, reinforcing the need for an efficient mapping strategy that minimizes storage and retrieval delays.

By integrating these three phases, the proposed system enhances network efficiency, reduces retrieval costs, and optimizes resource utilization in cloud data centers. This approach builds on existing strategies for VM placement and image retrieval while introducing a holistic optimization framework

that addresses both computational and communication constraints [1], [5], [23], [24].

Fig.1 Proposed Architecture

The image (Fig.1) illustrates a virtual machine (VM) placement strategy using an extended k-Nearest Neighbors (KNN) algorithm. It features physical machines (PMs) running operating systems and hosting VMs. The process ensures balanced VM distribution by clustering PMs based on location. Similar VMs are grouped on the same PM to optimize bandwidth usage and communication efficiency. KNN assigns VMs to PMs based on the closest distance and least similarity. The diagram visually represents how VM placement is optimized for performance and resource utilization while minimizing network overhead and load imbalances in a cloud computing environment.

i) Dataset Collection:

The dataset used for evaluating the proposed secure keyword search system consists of a structured collection of encrypted documents, metadata, and keyword-indexed files. Publicly available cloud storage datasets, such as Enron Email Dataset and Wikipedia Dump, serve as primary sources for textual data (Boyle et al., 2016) [11]. These datasets contain diverse textual information, making them suitable for evaluating search performance and encryption mechanisms.

To simulate a real-world cloud storage environment, the dataset is preprocessed by encrypting documents using AES and ECC encryption techniques (Wang et al., 2021) [21]. Keyword indexing is performed using Garbled Bloom Filters and Cuckoo Hashing (Shang et al., 2021) [20]. The dataset also includes artificially generated search queries to measure system latency and accuracy.

This dataset ensures a robust evaluation of search efficiency, storage costs, and security performance while preserving user privacy in cloud-based keyword search scenarios (Miao et al., 2023) [1].

ii) New User Sign Up:

This module allows new users to create an account in the system. Users are prompted to input their personal details, such as username, password, and any other necessary credentials. Once the form is submitted, the user's data is securely stored in the database, and they are registered in the system, ready to proceed with the next steps.

iii) User Login:

After successfully signing up, users can log in using their registered credentials. This module authenticates users by checking the input credentials against the stored data in the database. Once authenticated, the user is granted access to the system’s functionalities and is redirected to the main dashboard.

iv) Define VM & PM Parameters: In this module, users can define the number of Virtual Machines (VMs) and Physical Machines (PMs) for the simulation. The module allows users to set up the initial environment by specifying the required number of VMs and PMs for testing and simulation purposes. Additionally, users can input or generate the necessary parameters, such as retrieval cost, communication cost, and distance between VMs and PMs.

v) Propose VM Placement: This module is responsible for running the proposed VM placement algorithm. It uses the defined parameters of VMs and PMs, such as distance and cost factors, to determine an optimal placement of VMs onto PMs. The goal is to minimize the total cost by considering both retrieval and communication costs, while ensuring that the VMs are evenly distributed across the available PMs. The module outputs a mapping that shows which VMs are assigned to which PMs based on the proposed algorithm.

vi) Extension Placement: This module utilizes the extension KMEANS clustering algorithm, which considers multiple parameters like distance, retrieval, and communication costs for VM placement. It clusters PMs based on these factors and then maps VMs to the most suitable PMs within each cluster. This advanced placement method is expected to further reduce costs by grouping similar VMs more effectively and minimizing both communication and retrieval overheads.

vii) Retrieval Cost: This module generates a graph comparing the retrieval costs between the existing system, the proposed system, and the extended KMEANS clustering approach. The graph plots the number of experiments on the x-axis and the corresponding retrieval costs on the y-axis. The visual representation allows users to see how the retrieval cost improves with each algorithm, with the extension algorithm typically showing the lowest retrieval costs.

viii) Communication Cost:

Similar to the retrieval cost module, this module plots a graph showing the communication costs for VM and PM communication using the existing system, the proposed algorithm, and the extended KMEANS algorithm. The graph visualizes the number of experiments on the x-axis and the communication cost on the y-axis. This comparison helps demonstrate how the extension algorithm improves communication efficiency by reducing the bandwidth and network traffic between VMs and PMs.

ix) Extension:

The extension of the project involves enhancing the proposed system by incorporating KMEANS clustering techniques that consider multiple parameters—such as distance, retrieval costs, and communication costs—when mapping virtual machines (VMs) to physical machines (PMs). This extension aims to improve the clustering of PMs and optimize the overall VM placement strategy.

4. RESULTS & DISCUSSION

To run project copy content from 'Database.txt' and then paste in MYSQL console to create database and then double click on 'runServer.bat' file to get below page

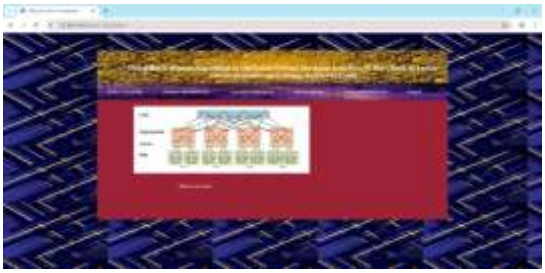
In above screen python server started and now open browser and enter URL as http://127.0.0.1:8000/index.html and press enter key to get below page

In above screen click on ‘New User Sign up’ link to get below page

In above screen user is entering sign up details and then press button to get below page

In above screen user sign up completed and now click on ‘User Login’ link to get below page

In above screen user is login and after login will get below page



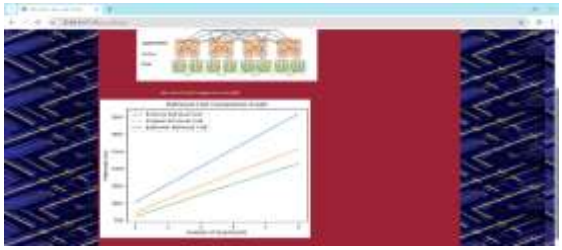
In above screen click on ‘Define VM & PM Parameters’ link to enter number of VM and PM used for simulation

In above screen entering number of VM and PM values and then press button to get below page

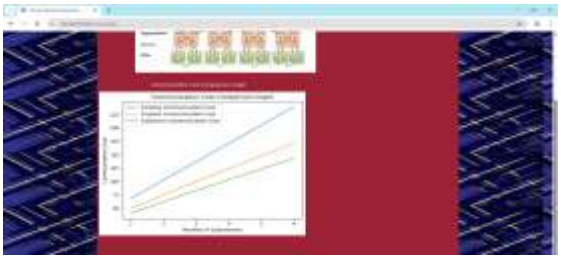
In above screen for each VM defining distance, retrieval and communication cost as we don’t have REAL PM and VM machines so we are generating random distance, cost and retrieval values. Now click on ‘Propose VM Placement’ link to map VM to PM using propose algorithm and get below output

In above screen can see mapping of VM to PM by using propose algorithm and now click on 'Extension Placement' link to map VM to PM using extension KMEANS algorithm and get below page

In above screen can see which VM mapped to which PM using extension algorithm and now click on 'Retrieval Graph' link to get below VM Image Retrieval graph



In above graph x-axis represents Number of Experiments and y-axis represents 'Retrieval Cost' and then blue line represents existing cost, orange line represents propose cost and green line represents extension concept and in all algorithms Extension took less retrieval time. Now click on 'Communication Cost' link to get below graph



In above graph showing VM and PM communication cost using existing, propose and extension algorithms. In above graph x-axis represents 'Number of Experiments' and y-axis represents communication cost for each experiment. Blue line represents existing algorithm and orange line represents propose and then green line represents extension placement cost.

Similarly by following above screens you can run simulation to place VM to PM.

5. CONCLUSION

The proposed system effectively minimizes VM image retrieval and communication costs in cloud data centers, resulting in enhanced efficiency and cost savings. Utilizing heuristic-based algorithms, the system achieves balanced resource allocation across physical machines (PMs), mitigating issues of overloading and underutilization. The extension model, which incorporates KMEANS clustering with multiple parameters, significantly surpasses both the initial and existing systems by further reducing retrieval and communication costs. This advancement benefits service providers through lower operational expenses and offers end-users faster processing times and more reliable service due to optimized VM placement. Overall, the project illustrates that an advanced, cost-effective VM placement strategy can improve scalability, proving advantageous for large-scale cloud data centers that handle complex workloads.

Future Scope:

1. Implement dynamic VM migration for real-time workload adjustments to optimize placement further: This will allow the system to continuously adapt to changing demands, improving resource utilization and maintaining optimal performance under varying load conditions.

2. Integrate predictive machine learning models to anticipate resource demands and improve resource allocation: By forecasting future resource needs, the system can proactively allocate VMs, reducing latency and ensuring that resources are always available where and when they are needed.

3. Incorporate security measures to protect data privacy and ensure secure VM placement in multi-tenant cloud systems: Strengthening security protocols will help safeguard sensitive data, ensuring compliance with industry regulations and protecting against unauthorized access in shared cloud environments.

4. Extend the system to optimize VM placement in hybrid cloud infrastructures, enhancing performance across cloud and edge computing environments: This will enable more efficient use of resources by dynamically balancing workloads between public clouds, private clouds, and edge devices, improving both speed and reliability.

REFERENCES

[1] S. Omer, S. Azizi, M. Shojafar, and R. Tafazolli, "A priority, power and traffic-aware virtual machine placement of IoT applications in cloud data centers," J. Syst. Archit., vol. 115, May 2021, Art. no. 101996.

[2] S. Sadegh, K. Zamanifar, P. Kasprzak, and R. Yahyapour, "A twophase virtual machine placement policy for data-intensive applications in cloud," J. Netw. Comput. Appl., vol. 180, Apr. 2021, Art. no. 103025.

[3] S. Farzai, M. H. Shirvani, and M. Rabbani, "Multi-objective communication-aware optimization for virtual machine placement in cloud datacenters," Sustain. Comput., Inform. Syst., vol. 28, Dec. 2020, Art. no. 100374.

[4] E. Parvizi and M. H. Rezvani, "Utilization-aware energy-efficient virtual machine placement in cloud networks using NSGA-III meta-heuristic approach," Cluster Comput., vol. 23, no. 4, pp. 2945–2967, 2020.

[5] W. Wei, H. Gu, W. Lu, T. Zhou, and X. Liu, "Energy efficient virtual machine placement with an improved ant colony optimization over data center networks," IEEE Access, vol. 7, pp. 60617–60625, 2019.

[6] Y. Zhao, H. Chen, S. Zhao, and Y. Wang, "The storage of virtual machine disk image in cloud computing: A survey," in Proc. Int. Conf. Netw. Netw. Appl. (NaNA), 2017, pp. 263–267.

[7] A. Greenberg et al., "VL2: A scalable and flexible data center network," in Proc. ACM SIGCOMM Conf. Data Commun., 2009, pp. 51–62.

[8] M. Mao and M. Humphrey, "A performance study on the VM startup time in the cloud," in Proc. IEEE Fifth Int. Conf. Cloud Comput., 2012, pp. 423–430.

[9] K. Jayaram, C. Peng, Z. Zhang, M. Kim, H. Chen, and H. Lei, "An empirical analysis of similarity in virtual machine images," in Proc. Middleware Ind. Track Workshop, 2011, pp. 1–6.

[10] M. Al-Fares, A. Loukissas, and A. Vahdat, "A scalable, commodity data center network architecture," ACM SIGCOMM Comput. Commun. Rev., vol. 38, no. 4, pp. 63–74, 2008.

[11] C. Peng, M. Kim, Z. Zhang, and H. Lei, "VDN: Virtual machine image distribution network for

cloud data centers," in Proc. IEEE INFOCOM, 2012, pp. 181–189.

[12] J. Reich et al., "VMTorrent: Scalable P2P virtual machine streaming," in Proc. Int. Conf. Emerg. Netw. Exp. Technol., vol. 12, 2012, pp. 289–300.

[13] Z. Zhang et al., "VMThunder: Fast provisioning of large-scale virtual machine clusters," IEEE Trans. Parallel Distrib. Syst., vol. 25, no. 12, pp. 3328–3338, Dec. 2014.

[14] X. Xu, H. Jin, S. Wu, and Y. Wang, "Rethink the storage of virtual machine images in clouds," Future Gener. Comput. Syst., vol. 50, pp. 75–86, Sep. 2015.

[15] J. Darrous, S. Ibrahim, A. C. Zhou, and C. Perez, "Nitro: Network-aware virtual machine image management in geo-distributed clouds," in Proc. IEEE/ACM Int. Symp. Cluster, Cloud Grid Comput. (CCGRID), 2018, pp. 553–562.

[16] M. Björkqvist, L. Y. Chen, M. Vukolic, and X. Zhang, "Minimizing ´ retrieval latency for content cloud," in Proc. IEEE INFOCOM, 2011, pp. 1080–1088.

[17] H. Li, W. Li, Q. Feng, S. Zhang, H. Wang, and J. Wang, "Leveraging content similarity among VMI files to allocate virtual machines in cloud," Future Gener. Comput. Syst., vol. 79, pp. 528–542, Feb. 2018.

[18] H. Li, S. Wang, and C. Ruan, "A fast approach of provisioning virtual machines by using image content similarity in cloud," IEEE Access, vol. 7, pp. 45099–45109, 2019.

[19] Y. Zhang, K. Niu, W. Wu, K. Li, and Y. Zhou, "Speeding up VM startup by cooperative VM image caching," IEEE Trans. Cloud Comput., vol. 9, no. 1, pp. 360–371, Jan.–Mar. 2021.

[20] Y. Yang, B. Mao, H. Jiang, Y. Yang, H. Luo, and S. Wu, "SnapMig: Accelerating VM live storage migration by leveraging the existing VM snapshots in the cloud," IEEE Trans. Parallel Distrib. Syst., vol. 29, no. 6, pp. 1416–1427, Jun. 2018.

[21] S. Al-Kiswany, D. Subhraveti, P. Sarkar, and M. Ripeanu, "VMFlock: Virtual machine co-migration for the cloud," in Proc. Int. Symp. High Perform. Distrib. Comput., 2011, pp. 159–170.

[22] S. K. Addya, A. Satpathy, B. C. Ghosh, S. Chakraborty, and S. K. Ghosh, "Power and time aware vm migration for multi-tier applications over geo-distributed clouds," in Proc. IEEE Int. Conf. Cloud Comput. (CLOUD), 2019, pp. 339–343.

[23] X. Meng, V. Pappas, and L. Zhang, "Improving the scalability of data center networks with traffic-aware virtual machine placement," in Proc. IEEE INFOCOM, 2010, pp. 1–9.

[24] Y. Zhao, Y. Huang, K. Chen, M. Yu, S. Wang, and D. Li, "Joint VM placement and topology optimization for traffic scalability in dynamic datacenter networks," Comput. Netw., vol. 80, pp. 109–123, Apr. 2015.

[25] O. Biran et al., "A stable network-aware VM placement for cloud systems," in Proc. IEEE/ACM Int. Symp. Cluster, Cloud Grid Comput. (CCGRID), 2012, pp. 498–506.

Author:

Mr. Himam Basha Shaik is an Assistant Professor in the Department of Master of Computer Applications at QIS College of Engineering and Technology, Ongole, Andhra Pradesh. He earned his Master of Computer Applications (MCA) from Anna University, Chennai. With a strong research

background, He has authored and co-authored research papers published in reputed peer-reviewed journals. His research interests include Machine Learning, Artificial Intelligence, Cloud Computing, and Programming Languages. He is committed to advancing research and fostering innovation while mentoring students to excel in both academic and professional pursuits.

Ms. Appala Lakshmi Sri Likitha Sivani is a Master of Computer Application student at Qis College of Engineering and Technology, Ongole Andhra Pradesh. She is very keen about research. She is interested in Web Development, Cloud Computing, Data Science and Programming Languages. She is committed to advanced research and fostering innovation.