Journal of Information Systems Engineering and Management
2025, 10(3)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Proficient Resource Allocation Technique for Cloud Resource Allocation using Deep Learning

V.Nisha[1], N. Gomathi[2], G. Linda Rose[3]
[1]*Assistant Professor Faculty of Science and Humanities Department of Computer Applications SRM Institute of Science and Technology, Chennai.*
[2]*Assistant Professor Department of Computer Science (SFS) SDNBVC, Chrompet, Chennai*
[3]*Assistant Professor Division of Digital Sciences Karunya Institute of Technology and Sciences, Coimbatore*

ARTICLE INFO

Received: 07 Nov 2024
Revised: 28 Dec 2024
Accepted: 10 Jan 2025

ABSTRACT

Cloud Provider (CP) offers resources to the various categories of clients according to the consumer's required demand for quality of service (QoS). When a physical machine (PM) is overloaded, the performance of its virtual machines (VMs) may degrade. Idle PMs can be shut down to conserve energy. This paper introduces a new approach for resource provisioning through VM consolidation and migration, aiming to meet user demands, minimize Service Level Agreement (SLA) violations, and reduce performance degradation during resource shortages. Initially, the workload of PMs for future time interval is predicted from the workloads of several previous time intervals of PMs using deep learning. If resource utilization across PMs is uneven, the resource provisioning method is regularly activated during these intervals.

Keywords: resource allocation, modified spline linear interpolation, resource provisioning, resource utilization.

1. INTRODUCTION

Cloud computing [1] is an on-demand infrastructure platform that provides pay-per-use software for data centers. Resource provisioning [2, 3] are typically used to the find the right amount of resources needed for the job to minimize the budgetary costs from the consumers viewpoint and to optimize the usage of services from the service provider's viewpoints.

An efficient resource allocation method called hierarchy based least square approximation and interpolation method [4] was proposed in the cloud environment. The user's required bandwidth and memory are compared to the available resources using a weighted approximation method. This approach generates a group of linear equalities that aligns the customer's demand with the available resources. By using iterative interpolation technique predict the suitable resources using set of lines. However, it is not much more effective for a heterogeneous cloud environment.

For distributed resource allocation in a heterogeneous cloud environment, a modified spline interpolation method [5] was proposed. This method considered user requirements such as bandwidth and memory for job scheduling in a heterogeneous cloud environment. Resource provisioning is also an important component in cloud computing. Resource provisioning involves identifying, configuring, and managing software and hardware resources during operation.

In order to meet user demands, reduce SLA violations, and minimize performance degradation during the unavailability of resources, a novel method is proposed. Initially in the novel method, deep learning is introduced to predict the future workload of PM from their previous workload. Next, statistic metrics are used to assess irregularities in the predicted resource usage of a physical machine (PM). If uneven resource utilization is identified, the PM's state is evaluated for resource provisioning based on a cost metric. A cost measure which is the weighted sum of the resource utilization, energy, and traffic is calculated for each PM. If the cost measure is greater than a threshold value, then the PM is considered as overloaded PM otherwise it is considered as underutilized PM. If the state of the PM is overloaded, then the VM migration is invoked were some of the VMs running on the overloaded

PM are migrated away. If the state of the PM is underutilized, then the VM consolidation is invoked where the VM is consolidated and the PM is turned off to save the energy. The resource provisioning and the MSLI are processed simultaneously for effective allocation of resources in a heterogeneous cloud environment.

2. LITERATURE SURVEY

Lin et al. [6] proposed a resource allocation scheme, that dynamically allocated virtual resources according to their load changes, and the threshold method was utilized to fine-tune the decision of resource reallocation. Baranwal and Vidyarthi [7] introduced a fair, multi-variable combinatorial action scheme model that considers software pricing and efficiency criteria to allocate resources for cloud operations. In order to ensure consistency and robustness, the auctioneer modified some of the deal criteria. If the price was not met, the customer and the manufacturer should be responsible for a fine. During the following rounds, the supplier's prestige has declined. Tafsir and Yousefi [8] presented a combinatorial multi auction method for division of resource in the cloud computing market. This is the allocation of resource issue was initially modeled as an integer programming problem aimed at maximizing total profits for both providers and users. This auction-based resource allocation method is further refined by dynamically adjusting prices based on profits from prior rounds. Mergence & Korpeoglu [9] proposed generic resource allocation for heterogeneous cloud infrastructures. Novel metrics such as network bandwidth CPU, disk, and memory are used for VM allotment. This approach will be further developed to allocate requests considering physical proximity constraints, allowing VMs serving the same application to communicate more efficiently.

Vhatkar and Bhole [10] proposed a novel hybrid model recognized as the Whale Random-update reinforced Lion Algorithm(WR-LA)[11] for allocation of resource in cloud environments. WR-LA combines the Lion Algorithm(LA) [12] with the Whale Optimization Algorithm(WOA). The WR-LA performed the arbitrary update assessment of WOA in place of the female update in the productiveness process of the LA algorithm. The cloud environment's resource allocation was determined by maximizing the objective function in WR-LA. However, this model was more suitable only by considering a small number of instances for resource allocation in the cloud. Abed et.al.[12] Reviewed from the existing paper from workload prediction for dynamic resource allocation using machine learning techniques in cloud environment. Decision Tree, Support Vector Machine and Neural Networks are used to predict the resource demand in cloud environment. Optimized algorithm helps to reduce over provisioning and under provisioning of resources and also helps to utilize the resources properly.

Ahmed et. al.[13] compares various resource allocation algorithm to optimize the resource allocation in cloud. Author emphasis on deep learning reinforcement algorithm performs well when compare to the traditional method and also find the correlation among cost efficiency and quality of services. Zhao et.al. [14] proposed hybrid approach for scheduling task to improve resource allocation algorithm using machine learning techniques. With the help of machine learning algorithm predict the demand of the workload to schedule tasks. This method gives better utilization of resources, enhance the quality of service with less cost. Zhou et.al. [15] proposed optimized deep learning algorithm is used for dynamic provisioning to improve the resource utilization. Sharma et. al. [16] proposed predictive approach to optimize the resource allocation cost, to avoid the overprovisioning of resources and to optimize the performance.

Sharma & Kumar[17] proposed actor-critric method and deep reinforcement learning for resource allocation. The strategies for resource allocation is generated by actor network and resource allocation in optimized way, utilize energy in efficient way, cost reduction are done by critric method. Vasquez et. al.[18] proposed intelligent resource allocation using reinforcement algorithm to predict the future workload, scheduling tasks, identification of traffic. This algorithm optimize the resource allocation, reduce violation of SLAs and response time, and cost. Chen et.al.[19] drives a solution for challenges in combining compute and storage in large scale cloud network, it helps to increase the speed and efficiency of the model. Khan et. al. [20] explores the various resource allocation and workload scheduling strategies for large scale clod environment. Author highlights challenges in scheduling, heterogeneity and fault tolerance

3. PROPOSED METHODOLOGY

In this section, the proposed Resource Provisioning with MSLI (RP-MSLI) method for resource provisioning is described in detail for effective resource allocation in a heterogeneous cloud environment.

3.1. Problem Formulation

User request (X) such as number of CPU needed, Memory required, network bandwidth are represented in matrix form

$$X=\begin{bmatrix} x_{00} & \cdots & x_{0n} \\ \vdots & \ddots & \vdots \\ x_{mo} & \cdots & x_{mn} \end{bmatrix}$$

Allocated resources(Y) are represented as follows:

$$Y=\begin{bmatrix} y_{00} & \cdots & y_{0n} \\ \vdots & \ddots & \vdots \\ y_{mo} & \cdots & y_{mn} \end{bmatrix}$$

3.2 Resource Allocation

Deep Belief Network is a collection of RBM (Restricted Boltzmann Machines) layers are arranged hierarchically. One RBM layer output is the input for the next RBM layer. Each RBM layer is transformed as

$$LH^{(l)} = f(W^{(l)}\ X\ LH^{(l-1)} + b^{(l)})$$

Output of the hidden layer l is represented as $LH^{(l)}$, Weight Matrix of Hidden Layer is denoted by $W^{(l)}$, Hidden layer bias is symbolized as $b^{(l)}$, f denotes activation function.

Fine tune the Output layer which is in the final state with the help of supervised learning.

$$Y = og(W^{(out)}\ X\ LH^{(L)} + b^{(out)})$$

Weight matrix of output layer is denoted by $W^{(out)}$, Bias of output Layer is symbolized as $b^{(out)}$, og denotes output layer activation function.

The future workload of PM is predicted using a deep learning method based on the past external behaviors of VMs. The workload data is random and not linear. The data-driven model can figure out the inferred a variety range of patterns and central aspects from a vast quantity of the formerly loaded data. This data-driven technique is the composed of Deep Belief Network (DBN) and a logistic regression. From the previous workload data, the DBN extracts the high-level features in a free-range manner. Generally, DBN is a neural network with multiple layers which grows exceedingly tedious because of the flaw slop would terminate with the elaboration of cumulative number of hidden layers. This dilemma has been cleared up by quickly employing the free-range progressive greedy layer wise approach training procedure to optimize the neural network architecture for task volume estimation. The resource exploitation of all VMs of various amounts of prior time spans is given as intake to the DBN to anticipate the resource efficiency of VM in the forthcoming period. Figure 1 depicts the layered design for the subsequent resource load forecasting of VM.

The feedback which we are provide into the computational architecture is the all scrutinized VMs resource consumption in various amount of time intervals. Obviously we set the normalization range in between 0 to 1 to analyze the utilization of the available resources. From the many layer of the proposed structure the first one is the product of the total number of VMs N_{vm} which are available in the cloud and the average number of the time intervals N_{interv} happening. By frequently applying this approach, the upcoming workload requirements of independent as well as dependent VMs can able to be predicted at the common time. For the proposed data-driven architecture are original values, it used Gaussian Bernoulli RBM (CBRBM). CBRBM is composed of visible layer, hidden layer and output layer. The potential energy function and the provisional statistical distribution is can be presented as:

$$E(v,h|\theta) = \sum_{i=1}^{V} \frac{(v_i-a_i)^2}{2\sigma_i^2} - \sum_{i=1}^{V} b_j h_j - \sum_{i=1}^{V}\sum_{j=1}^{H} \frac{v_i}{\sigma_i} h_j w_{ij}$$

$$p(h_i|v;\theta) = \delta\left(\sum_{i=1}^{V} w_{ij} v_i + b_j\right)$$

$$p(v_i|h;\theta) = N\left(\sigma_i \sum_{j=1}^{H} w_{ij} h_j + a_i, \sigma_i^2\right)$$

where $\theta = (w, a, b)$, w_{ij} is described as the weight coefficient in between the observed unit v_i and unobserved unit h_j, a_i is the bias coefficient of v_i, b_i is the bias parameter of h_j, $\delta(x)$ can be logistic function and $N(\mu, \sigma_i^2)$ is the probability with the mean μ and variance σ^2.

3.4 Determination of hot spots and cold spots

After the determination of unevenness in the resource utilization of PM, the hot spots and cold spots are determined based on a cost measure. A cost measure is the weighted sum of the resource utilization, energy and traffic. The cost measure is calculated as,

$$cost(s) = W_1 \sum_{u \in U}(u)^2 + W_2 E + W_3 T$$

In above Eq. U is the set of overloaded resources in s, E is the energy and T is the traffic. $W_1 + W_2 + W_3 = 1$. A PM is defined as hotspot when its cost measure is above a user specified threshold σ. Therefore, some VMs operating on it should be migrated to other servers. The cost measure of a hotspot indicates its level of overload or underutilization. If a server is not considered a hotspot, its cost measure is zero.

A physical machine (PM) is classified as a cold spot when the utilization of all its resources falls below a threshold σ, indicating that the PM is primarily idle and may be a suitable candidate for shutdown to conserve energy. However, this action is only taken if the cost measure of the PM is also below σ. If the PM hosts at least one running VM, it must remain active; otherwise, it is deemed inactive. Additionally, a threshold is defined as the level of resource utilization that is sufficiently high to warrant the operation of a server, yet not so high that it risks becoming a hotspot due to temporary demand fluctuations from applications. For CPU and memory resources, the thresholds are set at 90% and 80%, respectively. If CPU utilization exceeds 90% or memory usage surpasses 80%, the PM is considered a hotspot. Figure 3.1 overall flow diagram of RP-MSLI method.

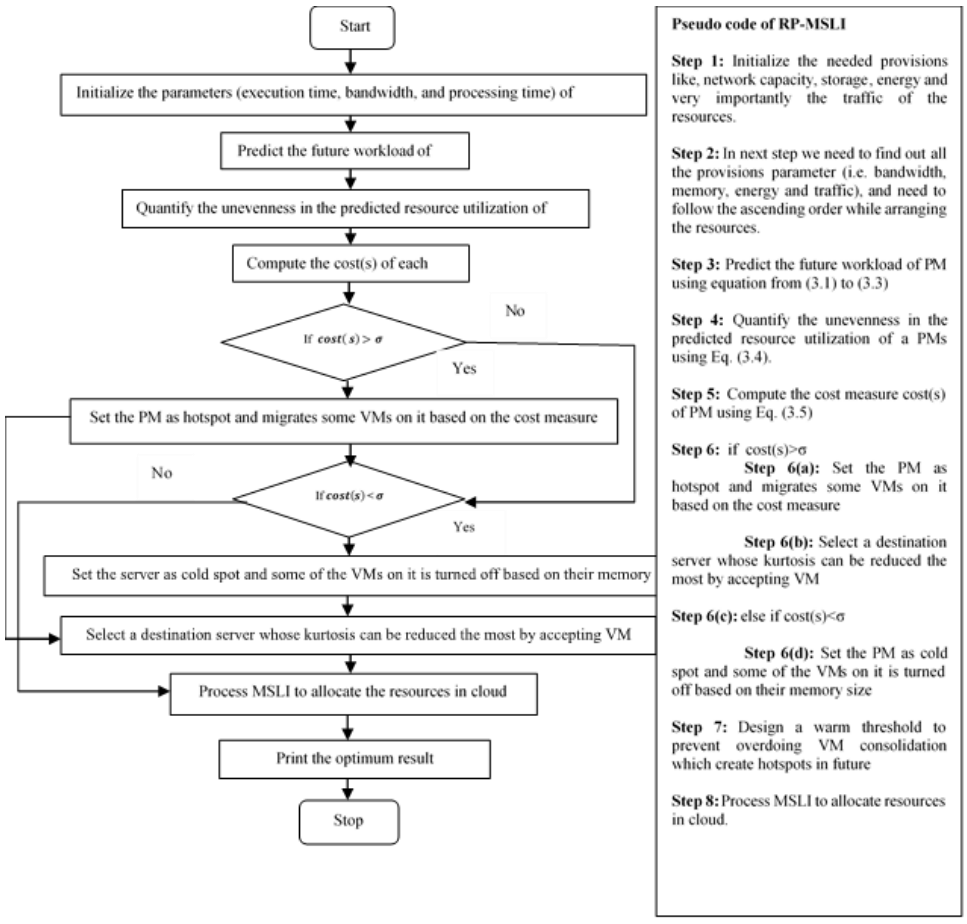

Figure 3.1 Overall flow of RP-MSLI method

4. SIMULATION RESULT

This section evaluates the Modified Spline Linear Interpolation (MLSI)[5] and Proficient resource allocation (PRA) algorithm efficacy in terms of turnaround, waiting, and completion times. It offers resources for the problem of resource allocation design. Twenty user tasks are in the wait list for the experiment, and five PMs are taken into account. The Modified Spline Linear Interpolation and PRA methods' completion, waiting, and turnaround times for varying task counts are displayed in Table 4.1.

Task Number	Completion Time		Waiting Time		Turnaround Time	
	MSLI	RP-MSLI	MSLI	RP-MSLI	MSLI	RP-MSLI
200	15.96	12.32	0	4	15.96	13.21
400	10.68	8.14	31.19	28.54	41.87	37.64
600	23.07	20.45	19.54	15.74	42.61	39.13
800	14.4	12.32	0	1	14.41	12.09
1000	13	11.03	24.41	21.21	37.41	34.31
1200	23.08	21.04	56	52	79.08	75.36
1400	21.73	19.32	28.54	25.34	50.67	45.21
1600	8.98	6.64	41.86	37.24	50.84	46.54
1800	19.54	17.74	0	0	19.54	15.32
2000	5.29	3.32	15.96	12.22	21.25	18.15

Table.4.1 Comparison of MSLI and RP-MSLI

The Comparison of completion time of MSLI and RP-MSLI is shown in Figure 4.1. The Comparison of waiting time of MSLI and RP-MSLI is shown in Figure 4.2. The Comparison of Turnaround time of MSLI and RP-MSLI is shown in Figure 4.3.

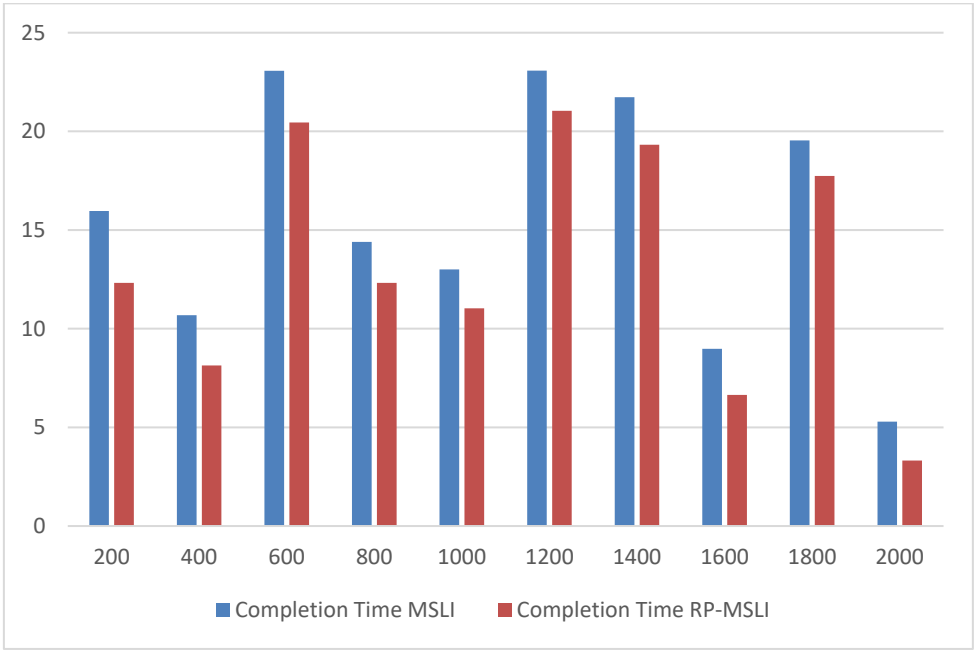

Figure 4.1: Completion Time of MSLI and RP-MSLI

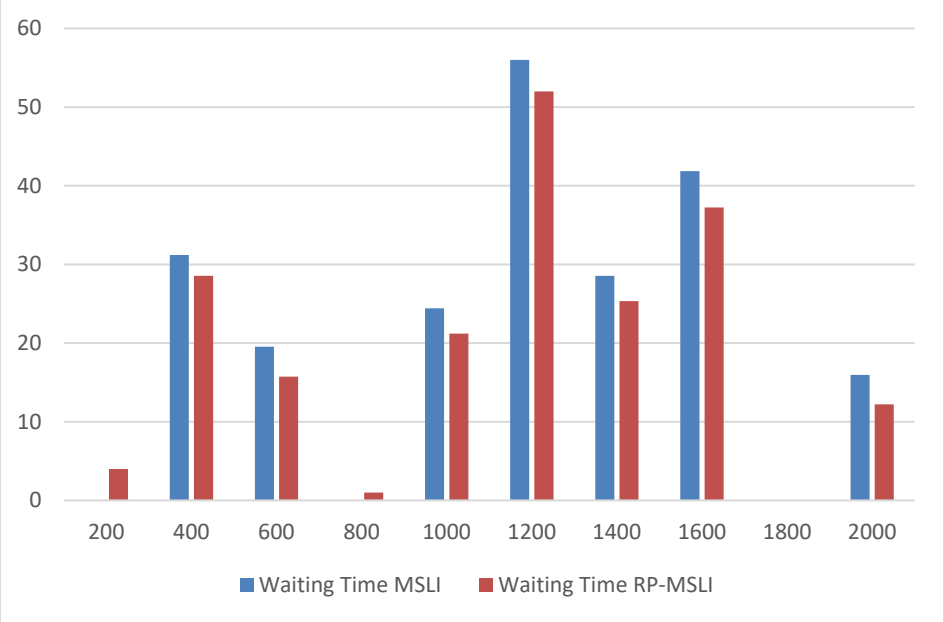

Figure 4.2: Completion Time of MSLI and RP-MSLI

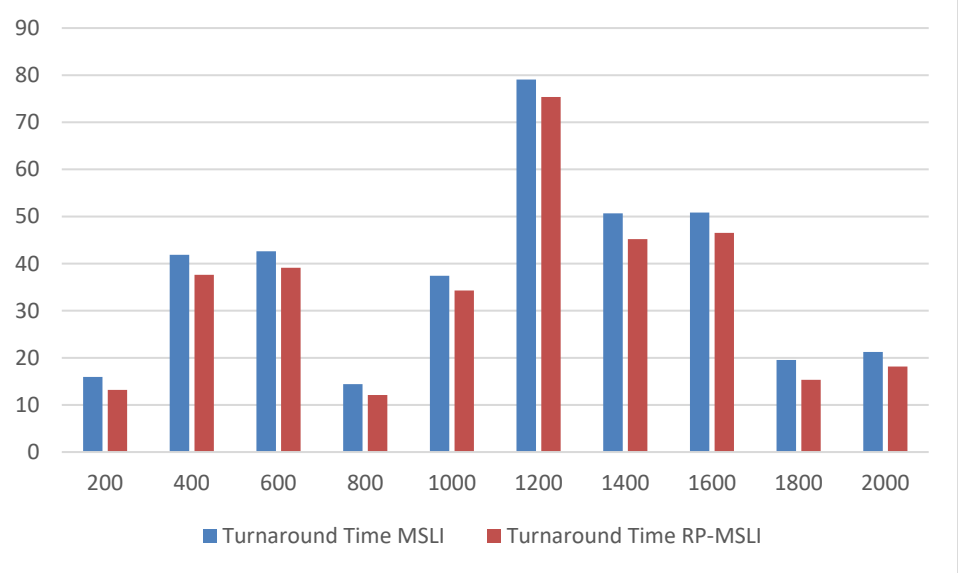

Figure 4.3: Completion Time of MSLI and RP-MSLI

When there are 20 jobs, PRA takes 37.24% less time to complete than the Modified Spline Linear Interpolation method for allocating resources. In terms of completion time, this analysis shows that the PRA approach performs better than the Modified Spline Linear Interpolation method. Waiting time is the amount of time that jobs remain in a queue while they wait for resources to be executed. When 20 jobs are used, the waiting time for PRA is 23.43% shorter than that of the Modified Spline Linear Interpolation method, suggesting that PRA performs better when it comes to waiting times. Turnaround time calculates how long it takes from the time a task is submitted until the user receives the finished product. All things considered, this analysis demonstrates that the PRA approach outperforms the Modified Spline Linear Interpolation method in terms of processing delay time and termination time.

5. CONCLUSION

In this article, a new method is recommended for effective resource allocation in heterogeneous cloud environment. Initially, a deep learning method called DBN is introduced for prediction of future workload of PMs based on the previous workload of PMs. The VM migrated when the PM is overloaded and the VM consolidated when the PM is underutilized. PRA algorithm processed simultaneously for effective resource allocation. The simulation results

prove that the proposed PRA method has better completion time, waiting time and turnaround time than Modified Spline Linear Interpolation method.

REFERENCES

[1] Alnajdi, S., Dogan, M., & Al-Qahtani, E. (2016). A survey on resource allocation in cloud computing. *International Journal of Cloud Computing: Services and Architectures (IJCCSA), 6*(5), 1-11.

[2] Bhavani, B. H., & Guruprasad, H. S. (2014). Resource provisioning techniques in cloud computing environment: A survey. *International Journal of Research in Computer and Communication Technology, 3*(3), 395-401.

[3] Calzarossa, M. C., Della Vedova, M. L., & Tessera, D. (2019). A methodological framework for cloud resource provisioning and scheduling of data parallel applications under uncertainty. *Future Generation Computer Systems*, *93*, 212-223.

[4] Nisha, V., & Vimala, S. (2019). Hierarchy based least square approximation and interpolation method for resource allocation in cloud environment. *International Journal of Advanced Computer Technology, 8*(11), 3494-3500.

[5] Vimala, S., & Nisha, V. (2020). Modified spline interpolation method for resource allocation in heterogeneous cloud environment. *International Journal of Scientific & Technology Research, 9*(3), 1610-1614.

[6] Lin, W., Wang, J. Z., Liang, C., & Qi, D. (2011). A threshold-based dynamic resource allocation scheme for cloud computing. *Procedia Engineering*, *23*, 695-703.

[7] Baranwal, G., & Vidyarthi, D. P. (2015). A fair multi-attribute combinatorial double auction model for resource allocation in cloud computing. *Journal of systems and software*, *108*, 60-76.

[8] Tafsiri, S. A., & Yousefi, S. (2018). Combinatorial double auction-based resource allocation mechanism in cloud computing market. *Journal of Systems and Software*, *137*, 322-334.

[9] Leontiou, N., Dechouniotis, D., & Denazis, S. (2018). A hierarchical control framework of load balancing of cloud computing services. *Computers and Electrical Engineering 67*(2018), 235-251.

[10] Vhatkar, K. N., & Bhole, G. P. (2019). Optimal container resource allocation in cloud architecture: A new hybrid model. *Journal of King Saud University-Computer and Information Sciences*.

[11] Beloglazov, A., Abawajy, J., & Buyya, R. (2012). Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. *Future generation computer systems*, *28*(5), 755-768.

[12] Abed, H., Hossain, M. S., & Abdur Rahim, M. (2023). A review on machine learning methods for workload prediction in cloud computing. IEEE Access, 11, 11745-11763. https://ieeexplore.ieee.org/document/10326297.

[13] Ahmed, N., & Khan, W. Z. (2024). Optimizing cloud resource allocation with machine learning: A comparative study. Journal of Cloud Computing: Advances, Systems, and Applications, 12(4), 210-225. https://www.researchgate.net/publication/378983298.

[14] Zhao, M., Li, Y., & Wang, X. (2023). A novel approach to cloud resource management: Hybrid machine learning and task scheduling. Springer Journal of Cloud Computing, 11(3), 89-101. https://link.springer.com/article/10.1007/s10723-023-09702-w.

[15] Zhou, X., & Tan, S. (2024). Optimizing resource allocation in cloud for large-scale deep learning applications. Journal of Cloud and AI Computing, 5(2), 134-142. https://journal.esrgroups.org/jes/article/view/652.

[16] Sharma, V., & Jain, P. (2023). Enhancing cloud cost efficiency: A predictive ML approach for optimized resource allocation. International Journal of Science and Research, 12(8), 255-263. https://www.ijsr.net/archive/v12i8/SR23816170845.pdf.

[17] *Sharma, S., Kumar, V., & Aggarwal, M.* (2023). Adaptive resource allocation in cloud data centers using actor-critic deep reinforcement learning. *International Journal of Recent Innovations in Technology, 11*(8), 112-118. https://ijritcc.org/index.php/ijritcc/article/view/6671.

[18] *Vasquez, E., & Tian, H.* (2023). Deep reinforcement learning-based intelligent resource allocation techniques with applications to cloud computing. In *Deep Learning Techniques for Cloud Resource Management* (pp. 153-170). Springer. https://link.springer.com/chapter/10.1007/978-3-031-53082-1_12.

[19] *Chen, L., & Xu, S.* (2024). Dynamic resource allocation for deep learning clusters with separated compute and storage. *IEEE Transactions on Cloud Computing, 12*(4), 301-310. https://ieeexplore.ieee.org/document/10228920.

[20] *Khan, A., & Alam, M.* (2024). Resource allocation and workload scheduling for large-scale distributed deep learning: A survey. *arXiv*. Retrieved from https://arxiv.org/abs/2406.08115.