Grenze International Journal of Engineering and Technology, June Issue

A Review on Resource Provisioning Technique's in Cloud Environment

Ranjana C[1], Anusha Bamini. A.M[2] and R. Chitra[3]
[1]Research Scholar, Division of Computer Science and Engineering, Karunya Institute of Technology and Science, India
Email: ranjanavipil14@gmail.com
2-3Division of Computer Science and Engineering, Karunya Institute of Technology and Science, India
Email: anushabamini@gmail.com, chitrajegan5@gmail.com

***Abstract*—Resource provisioning is the method of making sure that the cloud service can be efficiently resourced to the customers as demand increases. Resource provisioning is the procedure of selecting, deploying, and managing software like load balancers and database server management systems and hardware sources such as CPU, storage, and networks to guarantee utility performance. To do this in powerful manner, a CSP could want to take sure measures to effectively deliver on its SLAs. As an instance, the CSP may have procedures in place to feature servers or storage as call for increases. It goals to make sure that an organization can seamlessly get right of entry to the specified sources in an optimized and efficient manner. The general aim of aid provisioning is to permit the packages to utilize computational strength, garage, and offerings. Cloud aid provisioning requires the CSP to scale and manage consumer needs seamlessly the steps in resource provisioning are record Infrastructure, Configuring and booting, Provisioning API and Provisioning VMs. This paper suggests an evaluation between provisioning techniques utilized in cloud. The useful resource provisioning was formerly done with the aid of considering carrier level goals and with numerous scheduling methods Here we evaluate distinct strategies used in present day cloud computing environment.**

***Index Terms*— Resource provisioning, dynamic provisioning, job computation time, scheduling, Quality of Service.**

I. Introduction

Cloud computing is a broad and subterranean platform that helps customers to build advanced and scalable applications. The use of cloud computing should tackle over provisioning and under provisioning. Over provisioning means the purchased resources are not fully utilized it may lead to cost more than necessary. Under provisioning means purchased resources are not sufficient to meet the needs, it will lead to affect the application performance badly. There are three types of provisioning methods.

- Advanced Cloud Provisioning: also known as "post-scales cloud provisioning" customers get the resources upon contract or provider signup.
- Dynamic Cloud Provisioning: also referred to as "on-demand cloud provisioning," Clients are supplied with assets on runtime. It allows storage volumes to be created on-demand. Absolutely, without dynamic provisioning in Kubernetes, managing storage resources becomes a manual and time-consuming process for cluster administrators. They must individually contact their cloud or storage

Grenze ID: 01.GIJET.10.2.565_1

provider to generate new storage volumes, followed by creating persistent volume objects to represent these volumes within the Kubernetes cluster.

In the Cloud self-carrier, the client uses a shape from internet to accumulate resources from the cloud provider, sets up a consumer account, and will pay with a credit card whilst required.

The parameters that are used in resource provisioning are fine of provider, time, energy, and price. From these parameters QoS is considered as the principle parameter for diverse cloud workloads [12]. Cloud facts centre comprise hundreds and thousands of resources that wishes to be expected in prior for efficient provisioning. Cloud provisioning refers to the strategies for the disposition and mixture of cloud computing offerings within an organisation IT infrastructure. That is a vast time period that consists of the rules, tactics and an company's objective in sourcing cloud offerings and answers from a cloud provider issuer. Load balancing is a primary issue faced in cloud. It is defined as how the load is shared among the users. It improves the system output, usage of resource and performance of device. Different methods are proposed for load balancing

To identify a new or novel approach for load balancing. Load balancer allows to allocate sources equal to the jobs for resource performance and person delight at low price, offering extra quality.

A. Different Load Balancers

TABLE I. TYPES OF CLOUD LOAD BALANCERS

Load balancer	Explanation
Application Load Balancer	Routes request to server primarily based on the utility content material.
Gateway Load Balancer	Helps to set amount of resources available all time.
Network Load Balancer	Distributes network traffic across multiple VMs and servers with simple routing protocols.

Load balancer distributes the workload among different servers and arrange such a way that no server is overloaded or under loaded. There are different forms of load balancers. They are Application load balancer, Gateway load balancer and Network load balancer.

B. Load Balancing Approaches

TABLE II. LOAD BALANCE APPROACHES

Ref	Load Balancing Approaches
[16]	Load Balancer act as a central server. A new task allocation approach to allocate incoming task. The system model is based on Markov process. From the balanced state probabilities are obtained and expected VM utilisation is done. Using task allocation policy with time slot VMs are all equally used by the users.
[17]	Particle swam optimization is an Swam Intelligence based technique to optimize cloud. Offloading decisions in computing is noted. It has been applied to optimize task and workflow scheduling.
[18]	Request migration policies among multi servers for balancing load in cloud. Each server request migration verdict through information exchange. Each server's average response time is calculated. It's miles a disutility function and constantly seek to diminish value. Here we try to solve the load balance issue by embedding variational inequality theory and state there exist Nash Equilibrium.
[19]	Here we use classification approach based on number of files present. The classification is based on file type formatting. SVM uses formats like audio, video, text messages and images in cloud. Ant colony optimization mechanism with combined File formatting is proposed.
[20]	Load balancing is done using Predictive priority based Modified Heterogenous Earliest Finish Time Algorithm. It predicts the applications forthcoming demands of resources. Calculate the load demand rate for each request in the scheme priority. Need to add request to queue based on examination of load demand and emergency request. Next calculate advanced resource demand from observed or historical database.
[21]	Software Definition Network allows the organisation to divide different virtual network within a single network or to connect devices on different physical network to construct a single virtual network.
[22]	Using ML algorithms here we check the nodes are not overloaded. Here we examine the overload on network links and we cut down the overburden.

Modern approaches are used by cloud servers to distribute the work. Many of the applications include processing of millions of texts, videos, images, and all types of data. Recently machine learning techniques has been added to improve the accuracy.

II. METHODOLOGY OF RESEARCH

Cloud Service Providers (CSPs) encounter several challenges in resource management:

Resource Monitoring Complexity: Monitoring resource utilization across diverse tools and techniques poses a challenge. Various resources need constant oversight to ensure efficient usage and identify potential bottlenecks.

Availability and Accessibility: Ensuring continuous availability and easy access to resources for users' varying demands is crucial. Users should be able to access resources promptly as per their requirements.

Cost Management: Automated provisioning, while efficient, can lead to increased costs if not managed effectively. Proper allocation of resources is vital to prevent unnecessary expenses. Regular alerts about approaching cost thresholds are necessary to notify both cloud users and CSPs, enabling proactive cost management.

Performance Optimization: The primary purpose of provisioning is to maximise overall performance within the least quantity of time. Efficient allocation of resources should aim to optimize data transfer while minimizing costs. Striking a balance between these factors is essential for effective resource utilization.

To address these challenges, CSPs continually strive to enhance resource monitoring capabilities, improve resource accessibility, implement cost-effective provisioning strategies, and optimize. resource allocation to maximize performance while minimizing expenses. Automating alerts and refining provisioning mechanisms are critical steps in maintaining efficient resource utilization in the cloud environment.

RQ1: How to firmly optimize resource provisioning plan as a multi objective optimization problem in real world?

RQ2 : How to deal with over or under provisioning?

RQ3: How cost optimisation is achieved?

This paper compares different techniques to solve all these research related questions [13].

III. ANALYSIS

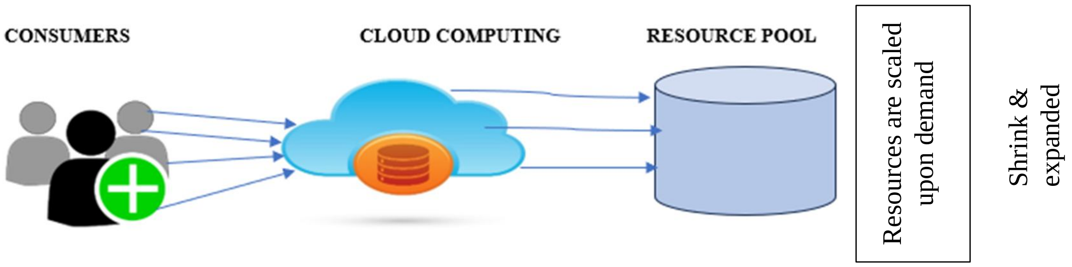

Fig.1 Resource scaling

A. How to firmly optimize resource provisioning plan as a multi objective optimization problem in real world?

When cloud resources encounter delays in reconfiguring or are allocated inadequately, it can lead to significant fluctuations in the execution times of adjacent tasks, causing either resource scarcity or wastage. To address this, a clustering ensemble technique is hired to partition project facts before scheduling. This method complements the alignment between workload necessities and the allocated sources.

With the aid of employing this technique, scheduling improves resource energy intake and utilization. Moreover, the system leverages the ARIMA model to forecast the task volume within each cluster. This prediction guides the dynamic initiation or shutdown of both virtual and physical resources, aiming to minimize both task waiting times and overall energy consumption.

The primary objective remains consistent: minimize energy usage while effectively provisioning resources from the active pool to meet the demands of ongoing tasks. This adaptive approach ensures that resources are dynamically adjusted based on predicted task volumes, ultimately

optimizing energy consumption and resource allocation. [1].

An alternative resource provisioning approach involves deploying a dedicated predictor for each cluster. These predictors forecast the number of tasks expected, thereby reducing task wait times and enhancing overall system resource utilization. Subsequently, based on these predictions, the resource configuration undergoes dynamic adjustments.

Additionally, a heuristic scheduling algorithm is introduced, prioritizing power consumption as its essential goal. This algorithm optimizes task scheduling to minimize energy usage while efficiently allocating resources. By dynamically adjusting configurations based on the predicted task volumes, this approach intentions is to strike a balance between task execution, resource utilization, and energy efficiency, thus enhancing the overall performance of the system.

In ERPDC (Enterprise Resource Planning in Distributed Cloud), when a new task enters the system, it undergoes a process of allocation. Initially, the task is allocated to the closest cluster by measuring the Euclidean distance between the task and each cluster's centre. This proximity-based assignment helps determine the most suitable cluster for task execution.

As soon as allocated, the gadget predicts the variety of responsibilities predicted for each cluster in the next time frame using the Prediction Task () function. This predictive function provides an estimate of the workload distribution across clusters, aiding in resource planning and allocation.

Following this prediction, the Scheduling Task() function comes into play. This function is responsible for intelligently scheduling tasks to appropriate resources within the clusters. Its primary objective is to ensure that tasks are executed efficiently and within the specified deadlines. By strategically assigning tasks to available resources based on their predicted workload, this scheduling mechanism optimizes task execution, contributing to timely completion and effective resource utilization within the distributed cloud infrastructure of ERPDC[3].

The iGniter approach is a periodic process designed to allocate GPU (Graphics Processing Unit) resources for newly incoming inference workloads. It's composed of 3 key modules: an inference workload placer, a GPU aid allocator, and an inference performance predictor.

Users interact with iGniter by submitting their Deep Neural Network (DNN) models along with request arrival rates and Service Level Objectives (SLOs) through the iGniter portal. Upon submission, iGniter initiates a lightweight workload profiling procedure across various GPU devices. This profiling aims to gather workload-specific and hardware-specific coefficients. These coefficients serve as crucial parameters.

Using these coefficients, the inference performance predictor module estimates the inference latency for the submitted workload. iGniter then guides both the GPU resource allocator and the inference workload placer. Their task is to identify an appropriate GPU device that not only minimizes performance interference but also guarantees meeting the specified SLOs among the available candidate GPUs. The system orchestrates the allocation of resources based on user demands within the cloud environment. Fig 1 likely illustrates this resource provisioning process, showcasing how resources are allocated in response to user demands within the cloud infrastructure.

TABLE III: COMPARISON OF RESOURCE PROVISIONING TECHNIQUES

Reference Paper	Objective	Methodology/ Techniques Used	Parameters	Advantages	Disadvantages
[1]	To improve utilization of resources an awareness mechanism on resource provisioning is done with iterative workloads on Apache Spark.	iSpark proactively preempts underutilized executors and safeguards the cached intermediate data from these executors, ensuring data consistency.	QoS, Time, Cost	This method improves average execution time of job.	The efficiency of cluster resources in physical and cloud deployment models must be improved.
[2]	Over allocation of tasks not only extend their own JCTs , but additionally probably make different duties to suffer from underneath allocation	Resource Balancer primarily based on Apache Spark	No of jobs, time	Improves the max-min fairness, shortest-task-optimisation is progressed. The common JCT has been reduced.	Can improve reforming pairwise
[3]	To allocate cloud resources efficiently and flexible.	Based on similarity among the jobs they are effectively partitioned at the time of arrival.	No of tasks, time	An energy-saving resource provisioning	Security breaches
[4]	As different network	ElasticNFV (Network	Cost, capacity	Affords a -segment	An unsuccessful

	functions consume different amount of resources, so we need to avoid unnecessary over provisioning of resources.	Functions Virtualization), a dynamic and first-class-grained cloud useful resource provisioning answer for VNF.	of VM, capacity of PM	minimal migration set of rules to optimize the migration time and embedding price of VNF times.	placement of resources may lead to VM operation overhead.
[5]	To seize the execution interference of inference workloads shared on GPUs.	iGniter	Cost, time	It mutually optimizes the GPU resource allocation and batch length configuration to greedily minimize the performance interference of DNN inference workloads.	System is slow when there is large arrival of requests.
[6]	To optimize resources provisionally offered by multiple cloud providers.	Optimal Cloud resource Provisioning Algorithm is used with a stochastic programming version.	Cost, performance	Reduce total value of useful resource provisioning in cloud surroundings.	The surest pricing scheme for cloud providers with consideration of competition within the marketplace is inquired.
[7]	To deal with applications that has time constraints that is deadlines running on resource-constrained clouds.	Time-touchy useful resource allocation and digital device (VM) provisioning framework. A request-to-node mapping set of rules based totally on the concept of Euclidean Distance that reveals the node with the first-rate suit of its resource requirements for every request.	Time, memory	Overloading can be minimized	No of requests discarded due to deficiency of resources should be monitored.
[8]	To reduce price and to improve customer satisfaction degree by using SAAS providers	SLA based resource provisioning algorithms to limit cost with the aid of minimizing useful resource and penalty value and through improving consumer pleasure level via minimizing SLA violations.	Time, cost	Overall value and SLA violations may be decreased	To improve customer satisfaction levels in negotiation with Cloud computing environments

B. How to deal with over and under provisioning?

Data configurations such as input data size and DAG (Directed Acyclic Graph) structures are gathered and utilized as inputs for a prediction model. This model is designed to anticipate the variability in Job Computation Time (JCT) for individual jobs. Additionally, it accommodates fluctuations in CPU core numbers, ensuring a comprehensive understanding of potential variations in computational demands.

The anticipated JCT values derived from this predictive model are employed by the Heuristic Allocation Initialization Algorithm. This algorithm operates to establish the preliminary allocation plan for the submitted jobs. Following this allocation determination, the jobs are executed via the Spark master, leveraging the allocated resources to carry out their computational tasks.

It was observed that larger input data volumes led to prolonged job starvation due to more instances experiencing severe under-allocation. However, ReB showcased significant advancements compared to other existing methods. This study concludes that ReB exhibits robustness in handling changes in input size, making it a resilient solution for managing variable workloads effectively [2].

ElasticNFV introduces a dynamic and finely detailed approach to cloud resource provisioning for Virtualized Network Functions (VNFs). It comprises two pivotal modules:

REP Module: This module utilizes a DMR (Dynamic Resource Management) model to capture real-time resource demands across one or more service chains. It incorporates both elastic resource provisioning mechanisms and resource provisioning techniques, adapting resource allocation dynamically to match varying demands.

SCH Module: This To reduce price and to improve customer satisfaction degree not only predicts the optimal time to trigger migrations and the resource requirements of VNFs but also formulates migration performance. It includes a TPMM (Time and Cost-Optimized Migration Management) algorithm to streamline migration time and embedding costs effectively.

ElasticNFV is built on the KVM and Open Switch infrastructure. Its implementation demonstrates significant enhancements in VNF performance, achieving optimized resource utilization and expedited migration times at a reduced cost. This solution stands out for its ability to dynamically adjust resource allocation, predict migration requirements, and optimize both performance and costs for VNFs. [5].

TIMER-Cloud refers resource allocation and provisioning of resources in the cloud environment. Resource allocation hits the selection, availability and management of CPU, memory and network and database management systems to the cloud users. The main problem of cloud is making available of these resources at right time at correct amount to the needers. Timer cloud is a time-based allocation of resources and VM based provisioning technique. User requests are prioritise based on deadlines and demands for resources. There are three mechanisms used here. Three priority-based methods are Time sensitive resource factor, Dominant share and unified k-Earliest Deadline First. Euclidian distance is used to find out the mapping distance between node and request . [7].

C. How cost optimisation is achieved?

The OCRP set of rules optimally provisions assets from a couple of cloud providers with the aid of solving stochastic integer programming with multistage recourse. Bender's decomposition method breaks down the OCRP problem into solvable subproblems, enabling parallel processing. BFResvResource, in optimizing the price of new account additions, allocates more resources than requested based totally at the customer's credit score stage, which is tied to their real wishes (first zero for brand new requests). As soon as a request's credit level exceeds the issuer's anticipated fee, additional assets are provisioned to expedite user account setup and minimize time.The algorithm focuses on minimizing penalty costs resulting from new account additions by reserving resources aligned with customer requirements, directly reducing SLA violations. Furthermore, resource reservations consider historical data and customer estimates, optimizing VM costs. As a result, the algorithm aims to minimize total costs (VM and penalty costs), effectively reducing SLA violations. [8].

To minimise the cost of cloud practice certain methods.

- Identifying mishandled resources.
- Display screen cost Anomalies.
- Use automatic resource-scaling to reduce prices.
- Employ reserved instances (RI)
- Do not forget poignant to a microservices surroundings.
- Use warmness maps to understand what goes on for your system.
- Put off the shadow IT practices.

TABLE IV: TECHNIQUES FOR COST OPTIMISATION

Reference paper	**Objective**	**Methodology**	**Technique used**	**Advantages**	**Parameter used**
[9]	To optimize resource allocation for heterogeneous IoT applications, make real-time decisions regarding the provisioning of MEC (Multi-Access Edge Computing) and cloud resources through an online greedy approach.	On-line cloud-edge resource provisioning framework	Delay-aware Lyapunov optimization	To lessen long term fee for cloud area applications.	Cost
[10]	Enhance the coverage of roadmaps across a diverse set of IaaS cloud resources while staying within a specified budget constraint.	Greedy Resource Provisioning and Heterogenous Earliest Finish Time with Duplication	The HEFT algorithm is modified to Certain budget limit	Minimize the cost	Cost
[11]	Optimizing global resource allocation frequently consequences in incredible inefficiencies within the neighbourhood allocation of resources for individual data facilities and cloud carriers. This imbalance contributes to unfairness in their revenue and income distribution.	A contractual resource-sharing model designed for federated geo-distributed clouds.	An algorithm designed for job scheduling and provisioning that takes into consideration both cost and time considerations.	Balancing global resource allocation productivity while ensuring local fairness in profit distribution.	Cost

IV. CONCLUSION

The efficient resource provisioning can be done by cloud providers offering VMs according to user needs. Virtualization technology helps providers to pack sufficient hardware resources into different types of VMs. Multiple pricing models are available like pay as you do model, on demand basics model and spot instances model. The application demand is not known early so based on the demands of users the resources must be scaled and shrieked after usage. The cost of resources on demand and spot will be varying. In terms of automation, cloud computing takes the tasks which are manual and repeatable and their automation reduces IT operations cost. In this review paper we mention on different provisioning issues and methods to tackle it.

REFERENCES

[1] Dynamic Resource Provisioning for Iterative Workloads On Apache Spark. Dazhao Cheng, Member, IEEE, Yu Wang, Fellow, IEEE, And Dong Dai,IEEE Transactions On Cloud Computing, Vol. 11, No. 1, January-March 2023

[2] Balance Resource Allocation for Spark Jobs Based On Prediction Of The Optimal Resource, Tsinghua Science And Technology Issnll1007-0214 05/10 Pp487–497DOI: 1 0. 2 6 5 9 9/T ST. 2 0 1 9. 9 0 1 0 0 5 4

[3] Elastic Resource Provisioning Using Data Clustering In Cloud Service Platform, Ei, Xiaomin Zhu, Member, IEEE, Daqian Liu, Junjie Chen, Weidong Bao , And Ling Liu , Fellow, IEEE Transactions On Services Computing, Vol. 15, NO. 3, May/June 2022

[4] Fine-Grained Cloud Resource Provisioning for Virtual Network Function. Hui Yu, Student Member, IEEE, Jiahai Yang, Member, IEEE, And Carol Fung IEEE Transactions on Network And Service Management, Vol. 17, No. 3, September 2020

[5] Igniter: Interference-Aware GPU Resource Provisioning for Predictable DNN Inference In The Cloud. IEEE Transactions on Parallel And Distributed Systems, Vol. 34, No. 3, March 2023

[6] Optimization Of Resource Provisioning Cost In Cloud Computing, . Ieee Transactions On Services Computing, Vol. 5, No. 2, April-June 2012.Sivadon Chaisiri, Student Member, Ieee,Bu-Sung Lee, Member, Ieee, And Dusit Niyato, Member, Ieee

[7] TIMER-Cloud: Time-Sensitive VM Provisioning In Resource-Constrained Clouds, Rehana Begam, Wei Wang, Member, IEEE, And Dakai Zhu, Senior Member, IEEE,2022

[8] SLA-Based Resource Provisioning For Software-As-A-Service Applications In Cloud Computing Environments, Linlin Wu1, Saurabh Kumar Garg1, Steve Versteeg2 And Rajkumar Buyya1

[9] CE-IoT: Cost-Effective Cloud-Edge Resource Provisioning for Heterogeneous IoT Applications, Zhi Zhou , Shuai Yu , Wuhui Chen , and Xu Chen, IEEE Internet Of Things Journal, Vol. 7, No. 9, September 2020

[10] GRP-HEFT: A Budget-Constrained Resource Provisioning Scheme for Workflow Scheduling in IaaS Clouds, Hamid Reza Faragardi , Mohammad Reza Saleh Sedghpour , Saber Fazliahmadi, Thomas Fahringer , Member, IEEE, and Nayereh Rasouli, IEEE Transactions On Parallel And Distributed Systems, Vol. 31, No. 6, June 2020

[11] Optimized Contract-Based Model for Resource Allocation in Federated Geo-Distributed Clouds ,Jinlai Xu , Student Member, IEEE and Balaji Palanisamy, Member, IEEE, IEEE Transactions On Services Computing, Vol. 14, No. 2, March/April 2021

[12] Optimizing Cloud-Service Performance: Efficient Resource Provisioning Via Optimal Workload Allocation, Zhuoyao Wang, Student Member, IEEE, Majeed M. Hayat, Fellow, IEEE, Nasir Ghani, Senior Member, IEEE, and Khaled B. Shaban, Senior Member,

[13] A Survey of Load Balancing in Cloud Computing: Challenges and Algorithms, Klaithem Al Nuaimi, Nader Mohamed, Mariam Al Nuaimi and Jameela Al-Jaroodi, 2012 IEEE Second Symposium on Network Cloud Computing and Applications.

[14] Performance Analysis and Optimization on

[15] Scheduling Stochastic Cloud Service Requests: A Survey, Shuang Wang, Xiaoping Li , Senior Member, IEEE, Quan Z. Sheng , Member, IEEE, and Amin Beheshti. Vol. 19, No. 3, September 2022

[16] Mastering Cloud Computing Textbook. Book by Christian Vecchiola, Rajkumar Buyya, and S.Thamarai Selvi**.**

[17] A Comprehensive Study of Load Balancing

[18] Approaches in the Cloud Computing Environment and a Novel Fault Tolerance Approach Muhammad Asim Shahid , Noman Islam, Muhammad Mansoor Alam Mazliham Mohd Su'ud , And Shahrulniza Musa,IEEEJune 30, 2020,

[19] A Fair, Dynamic Load Balanced Task Distribution Strategy for Heterogeneous Cloud Platforms Based on Markov Process Modeling. Stavros Souravlas 1,2, (Member, Ieee), Sofia D. Anastasiadou2, Nicoleta Tantalaki3, And Stefanos Katsavounis IEEE Access March 2, 2022,

[20] A fast converging and globally optimized

[21] approach for load balancing in cloud computing. Mana Saleh Al Reshan, Darakhshan Syed , Noman Islam , Asadullah Shaikh , (Senior Member, Ieee), Mohammed Hamdi , Mohamed A. Elmagzoub, Ghulam Muhammad2, And Kashif Hussain Talpur IEEE Access

[22] A Game Approach to Multi-Servers Load

[23] Balancing with Load-Dependent Server Availability Consideration Chubo Liu , Kenli Li , Senior Member, IEEE, and Keqin Li , Fellow, IEEE JANUARY-MARCH 2021

[24] A Hybrid Model for Load Balancing in Cloud Using File Type Formatting. Muhammad Junaid1, Adnan Sohail1, Adeel Ahmed 2, (Graduate Student Member, Ieee), June 19, 2020,

[25] A Predictive Priority-Based Dynamic Resource Provisioning Scheme With Load Balancing in Heterogeneous Cloud ComputingMayank Sohani , (Member, Ieee), And S. C. Jain April7,2021 IEEE Access

[26] Machine Learning-Based Load Balancing Algorithms in Future Heterogeneous Networks: A Survey,IEEE March 16, 2022,