UGC Approved Indexed Peer Reviewed Journal

ISSN: 2320-6586 (Online)

IILM UNIVERSITY

Technologies for Sustainable Development Goals 2025 (TECH4SDG-2025)

School of Computer Science & Engineering

IILM University Greater Noida

Presents

1st National Conference

on

Technologies for Sustainable Development Goals 2025 (TECH4SDG-2025)

3rd March 2025

Technically Co-Sponsored by IEEE UP SECTION (INDIA)

Venue:

IILM University, Greater Noida

Plot No.18,16, Knowledge Park II, Greater Noida, Uttar Pradesh 201306

Procedings Published at

International Journal of Research in Modern Engineering and Emerging Technology

Vol. 13, Sp. Issue 1, April 2025

Doi: https://doi.org/10.63345/ijrmeet.org.v13.si1.1301

Editor-in-Chief

Dr. Mylavarapu Seetharama Prasad

www.ijrmeet.org

INDEX

S. No.	Particulars	Page No.
1	A Comparative Study On Load Balancing Algorithms And Its Types **Methily Johri & Aditri Ashish**	1–10
2	Optimizing Supply Chain Management with Ethereum 2.0: Innovations in Blockchain Technology **Ayush Gupta & Dr. Ambuj Kumar Agarwal**	11–17
3	Smart Surveillance: "A Real-Time Crowd Analysis System for Density and Anomaly Detection" **Komal Sharma, Dr. Pankaj Kumar Varshney, Aditi Aggarwal & Gaurav Kumar**	17–22
4	AI-Driven Network Optimization: Matching Policyholders to Cost-Effective Providers Using Business Intelligence **Ruchi Mangharamani & Prof. (Dr) Sangeet Vashishtha**	23–29
5	The Future of UX: Designing for Emerging Technologies (AI, AR, VR, Voice UI) **Ojas Gupta & Prof.(Dr.) Arpit Jain**	30–37
6	Enhancing Liquidity Stress Testing: Best Practices for Modern Banking Risks **Padmini Bulani & Dr Abhishek Jain**	38–45
7	AI-Powered Investment Strategies: Optimizing Risk and Returns in Modern Banking **Aditya Mehra & Dr. Shakeb Khan**	46–53
8	Scaling Customer Journey Optimization in a Multi-Tenant Distributed System **Sumit Kumar Agrawal & Prof. (Dr) MSR Prasad**	54–61
9	Object Detection Using Convolutional Neural Networks: Advances, Techniques, and Applications	62–69

S. No.	Particulars	Page No.
	Bharath Thandalam Rajasekaran, Saravanan Thirumazhisai Prabhagaran & Prof. (Dr) Punit Goel	
10	AI-Augmented Intrusion Detection Systems for High-Speed Optical Networks **Sattvik Sharma & Prof.(Dr) Avneesh Kumar**	70–77
11	Generative AI in Data Security: Opportunities and Challenges **Vybhavreddy Kammireddy Changalreddy, Piyush Bipinkumar Desai & Prof (Dr) Ajay Shriram Kushwaha**	78–86
12	Integrating Salesforce with Cloud-Based Data Warehousing for Unified Analytics **Nidhi Shashikumar & Dr S P Singh**	87–94
13	AI-Powered Cybersecurity: A Unified Approach to Protecting Enterprise, Cloud, and SaaS Applications **Varun Grover, Khushmeet Singh & Er. Siddharth**	95–103
14	Sales and Order Analysis Using Excel-Based Dashboard: A Case Study on Vrinda Store **Mohd Ammar Khan & Harun Faridi**	104–106
15	ChatMania: A Realtime Chat Application **Dr. Kirti Shukla & Anshika**	107–110
16	Digital Surveillance and Domestic Abuse: A Legal Vacuum in Indian Marital Relationships **Vishal Nehra & Dr. Rakesh Kumar**	110–116
17	The Role of Emotional Appeal in Online Advertisements Targeting Working Women in Delhi-NCR **Mona Mishra & Dr. Praveen Mittal**	116–120
18	AI-Driven Data Governance Strategies for Enhanced Cloud Security in Enterprises **Chandana Pandey & Dr Gauri Shanker Kushwaha**	121–123

S. No.	Particulars	Page No.
19	Evolution of Social Media Advertising and Consumer Behavior **Mr. Pramod Rana & Prof (Dr) Punit Goel**	124–129
20	Sustainability Accounting Practices in Indian Manufacturing Firms **Mrs. Sonal Jain & Dr. Ajit Singh**	129–134
21	Fundamental Rights vs. Reasonable Restrictions: A Study of Evolving Judicial Balances **Mrs. Anshika Agarwal & Dr. Vijay Kumar Singh**	134–139
22	सतत शिक्षा के लिए हिंदी साहित्य की भूमिका: सतत विकास लक्ष्य-4 (गुणवत्तापूर्ण शिक्षा) का सांस्कृतिक विश्लेषण **श्रीमती रेणु गुप्ता, डॉ. बृजलता शर्मा**	140–144
23	Mind–Gut Synergy in Chronic Lifestyle Conditions: Evaluating Psychotherapy-Driven Behavioral Change and Microbiota-Mediated Metabolic Repair **Krithika. V, Dr. Sarbesh Kumar Singh**	145–151
24	प्राचीन शिक्षा प्रणाली से आधुनिक AI आधारित शिक्षा की ओर संक्रमण **गौरव जोशी, डॉ सारिका गोयल**	152–159
25	पर्वतीय क्षेत्रों में शिक्षा के डिजिटलीकरण की चुनौतियाँ एवं संभावनाएँ **राकेश उनियाल, डॉ. मीनाक्षी हर्राव वर्मा**	160–168
26	Women Leadership and Democratic Participation in India: Challenges and Opportunities in Parliamentary Politics **Renu Yadav, Dr. P.P.S. Rana**	169–176
27	Evaluating GST Effects on Competitiveness and Industrial Development **Aman Goel, Dr. Pooja Yadav**	177–183
28	एस०आर० हरनोट के कथा साहित्य में लोक संस्कृति, पर्यावरण चेतना और सामाजिक परिवर्तन का अध्ययन **सीमा देवी, डॉ० अरविन्द्र कौर चुंबर**	184–189

S. No.	Particulars	Page No.
29	Balancing National Security and Human Rights in Modern India **Shalu Jain, Dr. Vijay Kumar**	190–197
30	Economic Performance Drivers and Fiscal Resilience in Indian Airports **Lalit Kumar Sharma, Dr. Pooja Yadav**	198–203
31	Educational Governance and AI Learning Outcomes in Government Elementary Schools **Dr. Shubhra Chaturvedi**	203–210
32	Artificial Intelligence and the Future of Teacher Roles in India Classroom **Neeru Verma**	211–220
33	Research Culture Among Undergraduate Students in Indian Universities **Dr. Dhanashri Rajshri Ravindra Jadhav**	221–229
34	डिजिटल समाज में बहुभाषिक संचार का बदलता स्वरूप: भारतीय भाषाओं के समक्ष अवसर और चुनौतियाँ **Rakesh Uniyal**	230–234
35	Graduate Employability and Skill Gaps in State Universities **Narender Panghal**	235–242
36	Political Funding Transparency in Indian Election **Dr. Anand Kumar Dwivedi**	243–252
37	AI-Powered Platforms for Data Analytics and Audience Development **Sachin Sangal**	253–258
38	Engineering Leadership Models in AI Startups: Balancing Innovation and Operational Efficiency **Dr. Kalpesh Rasiklal Rakholia**	259–269
39	Challenges in Integrating CWSN in Regular Classrooms **Anil Kumar**	270–280
40	An Artificial Intelligence-Based Legal Compliance Framework for Real-Time Environmental Governance to Advance SDGs 9, 12, 13, and 16 **Dr. Jyoti Janardhan Reddy**	281–288

S. No.	Particulars	Page No.
41	Impact of Source Segregation and Waste Characterization on the Efficiency of Material Recovery Facilities **Varinder Vikram Singh, Dr. Pankaj Kumar Bharti**	289–296
42	Significance of Emotional Intelligence of schedule caste students in relation to Self-concept at secondary level **Mrs. Varsha Asher Wasker, Dr. Kuldeep Singh Tomar**	297–305
43	औपनिवेशिक भारत में स्थानीय बाजारों के सामाजिक परिवर्तन का ऐतिहासिक अध्ययन **Pankaj Tamta**	306–310
44	Teachers' Readiness to Integrate Generative AI into Lesson Planning and Classroom Instruction **Prema Bharti**	311–316
45	Battery Usage Optimization Techniques for AI-Heavy Mobile Apps **Dr. Akankasha Kathuria**	317–321
46	Gender, Technology, and Changing Family Structures in Urban India: A Socio-cultural Analysis **Prof. Sawant Jeetendra Madhukar Usha**	322–326
47	Law and Alternative Dispute Resolution in Contemporary Civil Justice: A Legal Study **Dr. Pooja Khurana**	327–331
48	English Literature and the Representation of Women in Contemporary Fiction: A Corpus-Based Study of Narrative Agency, 2000–2024 **Dr. Abu Farah Hoque**	332–336
49	Agricultural Impact of Biochar and Reduced Fertilizer on Mustard Soil Health and Yield **Dr. Bibhu Santosh Behera**	337–346
50	Ethical and Privacy Frameworks for AI-Driven Customer Data Analytics in Medical Profession **Dr. Gaurav Singh Tomar**	347–351
51	Legal Protection of Witnesses in Criminal Justice Proceedings **Dr. Jyoti Janardhan Reddy**	352–357

S. No.	Particulars	Page No.
52	Platform Culture and Fragmented Selfhood in Contemporary English Literature **Dr. Akanksha Singh**	358–362
53	Digital Health Literacy and Health Information Inequality in Rural India: A Multidimensional Assessment **Ranjana Jadaun**	363–368
54	स्मार्ट विनिर्माण और उद्योग 5.0 में कृत्रिम बुद्धिमत्ता, मानव-केंद्रित तकनीक और सतत विकास का अंतःविषयक अध्ययन **Dr. Ghamendra Kumar Sahu**	369–375
55	Contribution of Yoga-Based Breathing Techniques to Daily Stress Management **Dr. Sweta Kumari**	376–387
56	Algorithmic Silence and Unspoken Meaning in Contemporary English Fiction **Dr. Jyoti Devi**	388–396
57	Cost-Benefit Analysis of Custom Image Analysis Tools in Oncology Research **Sumit Malhotra**	397–405
58	उच्च शिक्षा में सूक्ष्म अधिगम तकनीकों का शैक्षिक प्रभाव **Ram Singh**	406–412
59	Optimising Laboratory Workflows with Low-Code/No-Code Software Solutions **Dr. Amit Nampalliwar**	413–421
60	भारतीय बहुभाषिक समाज में भाषा, संस्कृति और राष्ट्रीय एकता का अध्ययन **Dr. Geeta Rawat**	422–428
61	Influence of FinTech Innovation on Financial Resilience and Operational Efficiency of Retailers and Street Vendors **Prof. Manmeet Kumar Siras & Shyam Pandey**	429–435
62	भारत में आपातकालीन शक्तियाँ और लोकतांत्रिक उत्तरदायित्व: संसदीय शासन के अंतर्गत राष्ट्रपति की संवैधानिक भूमिका का विश्लेषण **सीता राम पंडित, डॉ. पी. पी. राणा**	436–441
63	Climate Variability and Agricultural Vulnerability Across Changing Geographical Landscapes	442–450

S. No.	Particulars	Page No.
	Dr. Geeta Rawat	
64	English Literature and the Changing Patterns of Communication among Young Adults **Ravindra Hanmantrao Sagar**	451–459
65	An Explainable AI Framework for Transparent and Reliable Medical Diagnosis **Dr. Thejoram Naresh Reddy Boya**	460–466
66	डिजिटल समुदायों का युवा शिक्षा और सामूहिक पहचान पर प्रभाव **डॉ. मनोज कुमार सकलानी**	467–473
67	Adaptive Technologies for Enhancing Online Shopping Experiences **Dr. Abhishek Bhattacharjee**	474–479
68	Impact of Digital Payment Adoption through AI-ML on Micro-Business Performance **Ralphy Joseph C J**	480–484
69	Role of Regional Parties in Shaping Federalism in India **Dr. Shailendra Kumar**	485–489
70	Serverless Computing Optimization: Enhancing Performance with AI-Based Resource Allocation **S. Venkatasubramanian**	490–495
71	Student Satisfaction in Autonomous vs. Affiliated Colleges **Dr. Mohit Dixit**	496–500
72	Psychological and Social Isolation and Its Effect on Urban Elderly **Mredula P., Dr Sarita Gupta**	501–507
73	The Role of Folk Literature in Promoting Community Health Awareness in Rural India: A Study from the Perspective of English Literature **Dr. Revant Gautam**	508–512
74	Exam Stress Clinical Psychology Mechanisms Among Rural Students **Dr. Avinash D. Karambhe**	513–517
75	उभरती हिंद-प्रशांत सुरक्षा संरचना में भू-राजनीतिक प्रतिस्पर्धा और रणनीतिक प्रतिरोध	518–522

S. No.	Particulars	Page No.
	Dr. Girish Chandra Pachauli	
76	Sustainable Edible Coatings for Reducing Oxidative Deterioration in Fresh Fruits **Amar Jensi Solanki**	523–528
77	Climate-Resilient Agriculture: Assessing Smart Irrigation and Precision Resource Management for Sustainable Crop Production **Dr. Bibhu Santosh Behera**	529–534

A COMPARATIVE STUDY ON LOAD BALANCING ALGORITHMS AND IT'S TYPES

Methily Johri
School of Computer Science
Galgotias University
Greater Noida, India
methily.johri@gmail.com
Aditri Ashish
School of Computer Science
Galgotias University
Greater Noida, India
aditriashish1@gmail.com

ABSTRACT
Load balancing refers to the situation where the servers provided by a website are busy means there are very much traffic on the server. To maintain this traffic a proper process or we can say a proper algorithm is evolved that is known as Load balancer. So, what actually load balanced do is it reduces the load (traffic) from one of the particular machine and distribute among all the machines of the particular servers. So, it automatically reduces the load(traffic) from the servers and now the website can work properly on the systems. We basically use Virtual Machines (VMs) for this purpose, it is the major - key factor of the load balancing systems. In the load balancers scheduling of tasks are most important and the virtual machines can reduce it very well, that is the reason why we use the virtual machines in the load balancing systems. So, cloud computing knowledge i s very important for the implementation of the load balancers. So, by the use of load balancing systems the workload can be managed or rather being distributed between the systems. So, this particular paper represents the review of a particular load balancer, how the load balancer works and what are the functionalities of a load balancer.

1. INTRODUCTION

Cloud computing is a basic & very important part of our proposed subject as it is the building block of the project, the LB is itself a part of the cloud computing algorithm, many of the web services provides load balancing systems on the cloud, such as AWS, GCP, Azure, IBM etc, these all are the examples of the web services that are available on the cloud. Cloud computing provides some of the functionalities such as accessing some data, some of the programs, and the files which are available on the web through cloud [1]. Earlier, thee were many difficulties of getting files, programs remotely, people were facing difficulties in getting and providing the files from different machines, So to overcome this situation in 1996 a theory came up and that was the theory of the cloud computing. By introduction of this type of theory life of the people became very easy and everything was available on the cloud people didn't worry about the storage and files. But, it is believed that the first use of cloud computing was done in the year 2006 when the google CEO "Eric Schmidt" introduced it[2]. By the introduction of the cloud computing the people have to only for that services that they are using, they didn't need to pay the extra money for the same thing, according to the user's needs and specifications the cloud provides services. Means users have to follow the system of per-use or we can also say that it is an example of SaaS [1]. In a basic cloud computing environment we had 2 things first is the front-end part and another is the back-end part. The front-end part basically describes how the whole software looks to the client, means it basically defines the front part what the user's see, the part through which the user interacts with the system. And the back end part which the user's cannot see or through which the user's cannot interact, the backend part mostly comprises of the coding part which is done for the user's interaction, even the non- tech person can easily interact with the cloud- computing technologies. The back-end part also deals with the service models(Basically the cloud service models). Beyond the line means in the back-end many of the physical machines are available which are responsible for providing the web services to the user. These physical machines are pre-installed with the softwares and the services which can be requested by the user, everything is pre-installed and according to the need of the user the things are being presented[3]. So, when the user requests any of the web-services then the requests are being acquired by the particular software or application and are being stored in the VMs(Virtual Machines). Through these virtual machines dynamically the loads are being managed in the whole system. These are responsible for the scheduling purpose. So, here the use of the load balancers came up which easily manage the load or the traffic in its website and can present the resources quickly to the user[4].

Scheduling of the tasks of the user to the particular VM is a specified process, a particular algorithm is working beyond it. In this way the proper utilisation of the resources is done. So that the complete system can work properly. So, by maintaining the proper efficiency in scheduling and to utilise the resources in the most efficient way here we introduce the load balancers, which can do this work In the most proper way by maintaining all the resources and by scheduling all the process and requests by the users[5].

1. Enhancement of Cloud Computing

In now a days most of the people are using cloud computing for any of the services they require because it is very easily available and they need to pay only that amount to the service providers which they are using. Since the year 2006 there always has been certain increase in the users of cloud as a service because it is very beneficial for the users. As, the cloud was first make in use in the early 2000s, it makes a revolutionary change in the computer development field. As earlier if we have to check any of the softwares which were made then we need to test it on every operating systems so that there should not be any bug in the software and the software can run easily in any of the operating systems. It usually takes hours to test the softwares. But by the introduction of the various cloud services we can easily check the softwares thorough the cloud computing web services. By use of the cloud computing services the software can be tested within minutes as it takes very less time to launch and also the user will find it inexpensive.So by these characteristics of the cloud computing environment many people started using it and the number is increasing gradually everyday[6].

2. Literature Review:-

In this section we particularly included the literature survey for the particular paper. First of all review of the existing systems would be explained then the theory and working and functionalities of the load balancers will be explained and then the load balancing algorithms that are being specified by the various researchers will be explained. Also at the later part we will include the new technologies that are coming up in the field of load balancing.

2.1.Existing Models

In the cloud computing environment it is very necessary to develop the load balancers for equal circulation of the process each process is having it's own priority and it can be scheduled according to a particular algorithm. So a proper algorithm should be developed for the data centres. Which may lead to a proper Scheduling algorithm without the failure. There are still some of the limitations in the research [7] , papers which have researched upon as there were less or even no pictorial representation and very less graphs available. While doing review we came across to a point that many of the author's could not define the terms properly and the functionalities of the load balancers[6]. We also found that many authors basically focused on the experimental results which is done to make ease in understanding the paper [8]. While doing these reviews we came across that there are four main parts or process of a load balancer that is the Response Time, Fault Tolerance, Waiting Time and the Latency rate . Response time in load balancer is defined as how much time is the load balancer taking to give the result of the user's request. Fault tolerance in the load balancers can increase the quality of the services that are being seen by the users. Waiting time is being defined as the amount of time that the user is being waited to see the output of the request for which the user has asked [9].

2.2.Load Balancing:-

LB is defined as the process of locating the traffic across the multiple servers. LB can improve the response rate ofthe application, using of the LB can automatically decrease the wait time and can decrease the latency rate of the application. Load Balancers can also increases the availability of the applications, resources of the applications [10]. Load Balancers can also maintain the overloading of the whole website by equally distributing the load or the requests by the user'sto the required virtual machines then the output is shown to the user's by maintaining the response time and decrease the latency of the particular application. It can also increase the performance of the cloud-based applications, the functionalities of the load balancers arediscussed more in the next section.

2.3 Performance Evaluation For Load Balancers

There are several factors on which the function of a load balancer depends, these factors will decide that how the load balancer will work, weather it will work efficiently, or it is more faster or not. So these factors are very much important for the load balancers [11].

Factors for Performance Evaluation:-

A. Throughput:- It defines the amount of data that is processed by load balancer per unit time. It is generally measured in bits or bytes per second. A load balancer having a high throughput rate can handle more traffic & also can process the requests as fast as possible without slowing down .This is very much important for the good performance and a good user experience [12].

B. Latency:- Latency is defined as the time taken by a load balancer to complete a request and to send back the response to the client. As lower as the latency rate, the faster will be the response time. High latent rates can lead to slower down the systems and hence the response time will increase in a load balancer. But reducing latency rates is very challenging specially in complex or distributed systems, because there are many factors that can contribute to latency [12].

C. Resource Utilisation:- This basically includes the resources used during the use case of a load balancers, it includes the CPU utilisation, memory utilisation and the

network utilisation. If resource utilisation will be high, then the overall process time of a load balancer will increase and the quality of the service will decrease. Sometimes it may also lead to complete unavailability of the load balancer. For optimised working of a load balancer one have to choose the load balancer according to the requirements of the infrastructure [12].

D. Availability:- This is the percentage of time the load balancer is operational ans is able to process the requests. High availability is very much important for the benefit of the users. It is the ability of the load balancer to handle requests and to provide a continuous service. High availability is critical for ensuring a good performance and to maintain a good user experience, it is difficult for systems that handles large amount of traffic or mission-critical application.

E. Scalability:- It is a quality of the load balancer to handle increasing traffic load without decreasing the performance. Scalability is important because it allows us to respond the changes in demand. There are several factors which can affect the scalability of a load balancer, including the architecture and design. To achieve a good scalability in load balancing, it is important to choose a load balancer which is designed for growth and flexibility.

F. Error Rate:- Error rate is defined as the number of requests that results in error, such as time outs or a filed connection. A high error rates indicates a problem in the load balancer. High error rates decreases the service quality of the load balancers. To minimise error rate in a load balancer it is important to choose a load balancer which is designed for reliability and for error handling.

G. Flexibility:- Flexibility in load balancers refers to the ability of the load balancer to adapt and change as the infrastructure grows over time. It includes the ability to handle the changes in the traffic patterns, being able to add or remove resource or being able to with between many load balancing algorithms. To achieve good flexibility it is important to choose a load balancer which is designed for growth and adaptability.

Contribution To the Literature :-

Our paper can be summarized as this is a summary or the study of many of the research papers already published. We have also included the present system on which the work is being done and also about the future trends, that what can be done after this. The results of many of the load balancers are also included in this paper. However, we have focused on the working of a particular load balancer about its functionalities. We have researched on the existing load-balancers that how the balancers manage their work by the use of the VMs, and also included our point of view based on our research, during research we came across various types of load balancers, read and understand each type of the balancers and also concluded with their pros and cons. We found that direct access load balancer is the most efficient and the fastest load balancer as it completes its process very fastly [13]. While researching we came across many of the facts and information about the load balancers but we compile the most important points and the most important information from our sources and presented them in this particular paper. We also took help of our mentor in gaining some of the information which we require and he also helped us in the research purpose. In this review paper we also mentioned about the types of load balancers, which load balancers should be used when we require a certain functionality.

3. Working of a Load Balancer :-

The whole business running on particular instance so, we need to secure the instances and we have to give better uptime for those instances. If any website connected with many users so there we have to balance the load of the application. So, we need to setup or balance the instances. Load balancer will do the same thing for us. It will basically balance the requests coming from different users.All the EC2 instances(Target nodes) will be launched in different subnets and AWS will make a replica of the instance launched in that particular subnet [14]. So if one subnet collapse then load balancer will connect the user's request to other target nodes. (While shifting the user's request from one EC2 instance to another EC2 instance, users might feel downtime in accessing the application deployed in another instances but this time duration is negligible). Load Balancing is a concept in which all the instances IP's (instances where application are deployed) are stored in the LB & the IP of LB or domain name is given to the client/users. So clients does not have the direct access to connect with the software deployed in the instances running behind. They have to first connect with the IP or domain name of the LB then load balancer use reverse proxy to get the application content and revert back the content to users/clients [15].

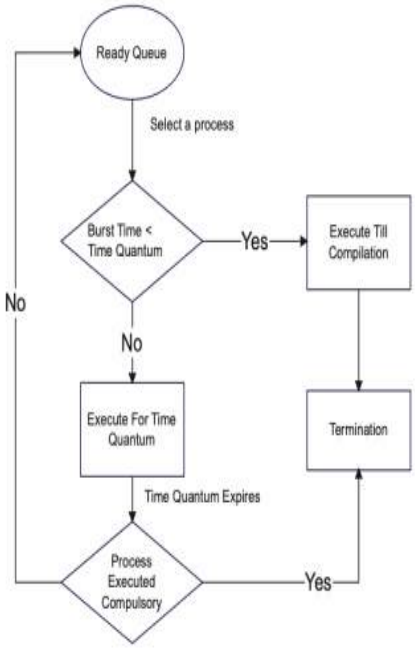
But if users/clients are increasing day by day then load balancer will launch a separate instance behind the scene and that instance will get registered into the load balancer which helps the users/client to easily access the website or application deployed in the backend servers. Load Balancer also keep on checking the health of backend servers. If any of the server is dead, the load balancer de-register the dead server and launch a new instance and register it to the load balancer. So, if any of the instance goes down it should be seamless, means users/clients doesn't have any trouble in accessing the applications [15]. A load balancer distributes incoming network traffic across multiple servers to improve performance, increase availability, and prevent downtime. It does this by receiving incoming requests and directing them to one of several backend servers, based on factors such as server health, current server load, and user-defined rules. Load balancers can be hardware-based, software-based, or a combination of both, and they can use different algorithms to distribute traffic, such as round-robin, least connections, IP hash, and more. Load balancers can also perform additional functions, such as SSL termination, session persistence, and content caching, depending on the specific product and configuration [16].

3.2 Types of Load Balancers based on Functionalities:-

While researching on the load balancers we came across many types of the load balancers which are being used now days and also that were used in the past. All the details of the type of load balancers are basically explained in this particular section

A. Working of Application Load Balancer:-

This LB works on the application layer of the OSI model, it usually takes routing decisions on (HTTPS/HTTP). This type of LB supports dynamic host port means it chooses the port dynamically. It is also known as layer 7 load balancers [20]. It distributes the request on multiple parameters among the application layer. In application load balancer the addition and removal of the tags can be done according to the user needs [16]. The performance of the load balancers increases in the application load balancers. When a user makes a request to an application served by an application LB the application LB checks the request & forwards it to the appropriate back-end server based on the content of the request. It allows the application load balancer to intelligently route traffic to different application components, such as servers running different micro services or APIs, based on the specific needs of the application. The application LB can also perform additional functions, such as SSL termination, content-based routing, health checks, and session persistence, depending on the specific configuration. Additionally, ALBs can integrate with another AMAZON WEB services, such as Auto Scaling and ECS, to automatically adjust capacity and optimize performance as demand changes [14].



Round Robin Scheduling

B. Working of Network Load Balancer:-

This type of LB make routing in the transport layer (TCP/SSL). These types of load balancers can handle the millions of requests per second. It supports or follows dynamic host port mapping. It is also known as layer 4 load balancers [17]. It does not consider any of the parameter at the application level such as the type of content, cookies, location, behaviour of the application etc. These type of load balancers cares only about the network layer information and manages the traffic on this basis only. When a user makes a request to an application served by an NLB, the NLB examines the IP protocol data, such as the source IP address and port, and forwards the request to the appropriate backend server based on the configured rules [19]. This allows the NLB to intelligently route traffic to different backend servers based on factors such as server health, capacity, and availability. Unlike Application Load Balancers, network load balancers support Transmission Control Protocol (TCP), User Datagram Protocol (UDP), and Transport Layer Security (TLS) protocols, which makes them suitable for applications that require high-throughput and low-latency network connections, such as video streaming, gaming, and messaging[20].

C. Working of Classic Load Balancer:-

This type of LB performs routing either on the transport layer of OSI) or the (HTTP/HTTPS). These type of LB required a relationship between the LB port & the container instance port. This LB supports EC2-Classic [18]. It also performs the health checks of the load balancers time to time. When a user makes a request to an application served by a CLB, the CLB examines the IP protocol data, such as the source IP address and port, and forwards the request to the appropriate backend server based on the configured rules. This allows the CLB to intelligently route traffic to different backend servers based on factors such as server health, capacity, and availability. “Classic load balancers support Transmission Control Protocol (TCP), Secure Sockets Layer (SSL), and Transport Layer Security (TLS) protocols”, and provide additional functions such as session stickiness, connection draining, and cross-zone load balancing, which helps improve application availability and scalability [21]. However, classic load balancers lack some of the advanced routing and traffic management capabilities provided by Application Load Balancers and Network Load Balancers, such as content-based routing and support for non-HTTP **protocols.z**

D. Working of Gateway Load Balancer:-

This type of load balancer allows us to set-up and manage the virtual applications, for example firewalls, prevention of the systems. It combines a transparent network gateway. It operates at the 3rd layer of the OSI model. Which is the network layer, It is also known as the layer 3 load balancer. Gateway load balancer supports the maximum transmission unit size of 8500 bytes. Gateway load balancers had many of the use-cases including setting - up the network services centrally as-a -service for everyone internal and external users, which results in the reducing of time, reducing of costs and also the reduction of the risk [22]. GWLBs operate at the edge of the AWS network, acting as a high-throughput gateway for inbound and outbound network traffic. GWLBs provide features such as protocol validation, rate limiting, and content-based routing to improve security and reduce the risk of Distributed Denial of Service (DDoS) attacks.When a user makes a request to an application served by a gateway LB, the Gateway load balancer examines the request and forwards it to the appropriate backend server based on the configured rules [23]. Gateway load balancer can route traffic to different application components running on different servers, containers, or AWS services based on the specific needs of the application. Gateway load balancers can also perform additional functions, such as SSL/TLS termination, network address translation and integration with AWS security services, such as AWS Firewall Manager and AWS WAF. Additionally, "Gateway load balancers can integrate with other AWS services, such as Auto Scaling and Elastic Container Service (ECS), to automatically adjust capacity and optimize performance as demand changes". Overall, Gateway load balancers provide a scalable and secure gateway for network traffic in complex AWS environments, helping improve application availability, performance, and security [24].

3.2 **Types of Load Balancers Based on Configuration:-**

A. Hardware Load Balancer:-

So, as the name suggests , this a physical, on- assertion, hardware gadgetry to distribute the traffic on the various servers. So hardware load balancers can handle the huge amount of traffic at a time, but they are not capable or are limited in terms of the flexibility. Also Hardware load balancers are very high in price [25].

B. Software Load Balancer:-

Software load balancers are basically the computer software/application that needs to be installed in the system and the functions of the software load balancers are same as that of the hardware load balancers. Software load balancers are of the two types- Commercial and open source. They are very less in cost so they can be a perfect alternative of the hardware counterparts [25].

C. Virtual Load Balancer:-

This type of load balancers are totally different from the hardware and the software load balancer, as it Is one of the combinations of the hardware and software load balancers. As, it is program or a function of the hardware load balancer and is working on a virtual machine.The application program of hardware gadgetry is executed on a virtual machine to allow the traffic redirected accordingly [25].

3.3 Best Load Balancing Algorithms:-

A. Round-Robin Algorithm:-

It is one of the most simplest and most used load balancing algorithm. It works on a very simple principle, the requests made by client is rotated to the servers using the rotation method, simple rotation is used. The initial request is sent to the initial application server and second to the second & so on. In this way the whole load of the application is managed. It is also very easy to implement and it can also be deployed very easily in a very efficient manner. But there is a demerit of this algorithm, as when there will every high traffic

FIG 1.1:- Round Robin Scheduling

This data flow diagram shows the steps involved in the round robin algorithm how the algorithm basically works. At the first step the round robin scheduling will get a request from the user and according to the priorities of the tasks the process will get automated an the scheduling will be done according to that only. At fist the task will enter into the ready queue and the process will get selected and if the task will attain burst time quantum, then the process will execute till it will get completed and then it will terminate, but if process does not reaches burst time quantum then it will reach to process execution part and runs till the process gets completed and after the completion the process will be terminated, but if the process will fail to complete by the load balancer then again the process will reach to the queue and again the same steps will be followed.

B. Weighted Round-Robin Algorithm:-

It is most similar to the round-robin LB algorithm, but the basic difference in this algorithm is that, it adds a weight which simple round-robin does not contain. With the help of this weight the algorithm becomes more dynamic and adds a value, as it is more better than the simple round robin algorithm. For example if a sever named X will contain 5 weight and a sever Y will contain 2 weight, then the load balancer will assign the 5 weight request to the server X and for every 2 weight, it will assign to the Y server. So, in the way it is somehow better than the simple round robin algorithm. It contains some weight and works as round robin, so it is named as Weighted-Round-Robin algorithm [20].

C. Least Connection Algorithm:-

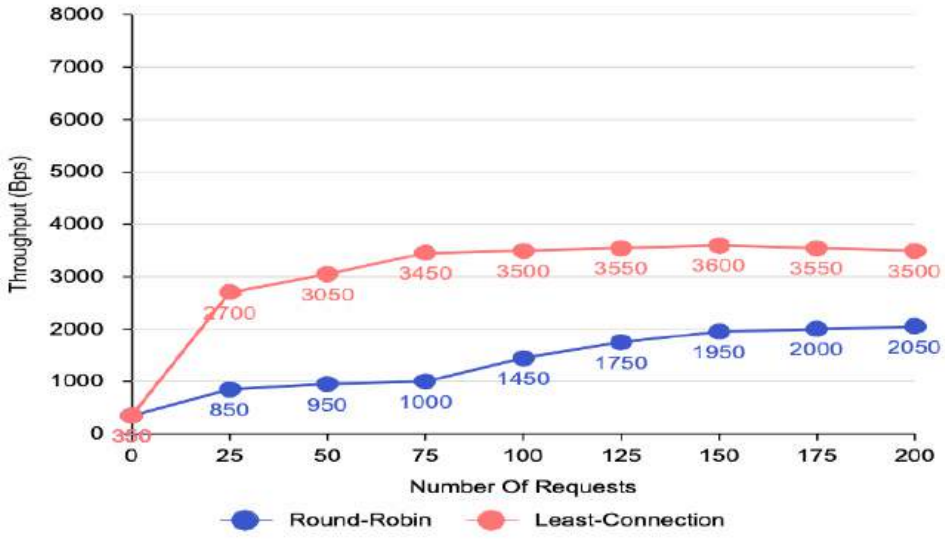
It is one of the best LB algorithm because it works on the active amount of load on a particular server, according to the current load the load is being distributed in this algorithm. So, it is little bit faster than than the above stated load balancing algorithms. It's working is in this way that if a server X is working upon 1,2,3 work loads provided by client and Y is working on 4,5,6 and Z is woking on 7,8,9, after some times X completed with loads 1 and 2 and only 3 is left, and Y had completed 4 and 5 and 6 is left and Z is working on all the three loads, then if a new work comes, then it will be allocated to X, the sever which has less work to do at that active time, so this will decrease the time of the execution. It is a dynamic type of load balancing

algorithm [27].

D. Least Response Time Algorithm:-
It is one of the most advanced version of the lest connection load balancing algorithm. It forwards the request to the servers which has active connection and which has very less average of responding, means whose average response time is very less. In backend, the server which is very fast, first of all it will receive the most requests, and as early as possible it will try to resolve the task. It is the fastest load balancing algorithm available. It also considers the number of active connections based on that only the whole process runs in the background [28].

4. Proposed Classification of Load Balancers

1. Static vs. Dynamic: Static load balancing algorithms distribute incoming requests based on predetermined rules or configurations, while dynamic load balancing algorithms adjust the distribution of requests based on real-time information such as the current load on each server.
2. Centralized vs. Distributed: Centralized load balancing algorithms use a central control entity to distribute requests, while distributed load balancing algorithms use distributed control and decision-making among the servers.
3. Deterministic vs. Probabilistic: Deterministic load balancing algorithms use predetermined rules to distribute requests, while probabilistic algorithms use random or probabilistic methods to distribute requests.
4. Application-level vs. Network-level: Application-level load balancing algorithms operate at the application layer of the network stack and are specific to a particular application or service, while network-level load balancing algorithms operate at the network layer and can be used to distribute traffic for any type of application or service [29].
5. Passive vs. Active: Passive load balancing algorithms do not actively monitor the servers and simply distribute requests based on predetermined rules, while active load balancing algorithms actively monitor the servers and adjust the distribution of requests based on the current state of the servers.
6. Hybrid load balancers: Hybrid load balancers are load balancers that combine elements of both hardware-based & software-based load balancers. They are designed to provide the benefits of both types of load balancers while minimizing their respective drawbacks. Hardware-based load balancers are typically faster and more reliable than software-based load balancers, but they can be expensive and inflexible. Software-based load balancers are generally more flexible and easier to configure than hardware-based load balancers, but they may not be able to handle as much traffic and may not perform as well. Hybrid load balancers attempt to strike a balance between these two approaches by using specialized hardware to handle the high-speed network traffic, while using software to provide flexibility and customization. This allows hybrid load balancers to offer the best of both worlds: high performance and flexibility. Some examples of hybrid load balancers include F5 BIG-IP and Citrix NetScaler. These products combine hardware-based load balancers with software-based features such as traffic management, security, and application acceleration [30].



5. Solution For Current Existing Load Balancers

A load balancer is a network device that distributes incoming traffic across a group of servers or resources in order to improve the performance and availability of a system. There are several solutions for implementing a load balancer, including:

1. Hardware load balancers: These are specialized physical devices that are designed specifically for load balancing. They are typically high-performance and can handle a large volume of traffic, but they can be expensive and inflexible [32].
2. Software load balancers: These are software-based solutions that run on regular servers or virtual machines. They are generally less expensive and more flexible than hardware load balancers, but they may not have the same level of performance.
3. Cloud-based load balancers: These are load balancing solutions that are provided as a service by cloud service providers. They are typically easy to set up and scale, but they can be more expensive than self-hosted solutions.
4. Open source load balancers: There are several open source load balancing solutions available, such as HAProxy, Nginx, and Pound. These solutions can be downloaded and installed on regular servers or virtual machines.

The best solution for a load balancer will depend on the specific requirements and constraints of the system. Factors to consider when choosing a load balancer include performance, cost, flexibility, and ease of use. Popular load balancer solutions include:

- HAProxy
- NGINX
- F5 BIG-IP
- AWS Elastic Load Balancer (ELB)
- Azure Load Balancer
- Google Cloud Load Balancer

6. Result

This section includes the comparative result which is obtained by the comparative analysis of this paper, so according to the needs of an individual one can select which load balancing algorithms they can use. A good load balancer is that load balancer which has the higher throughput, low latency rate, low resource utilization, less error rate, higher availability and good scalability and flexibility. These factors should be available so that the load balancer will not face any difficulty. These factors are very much important for a good load balancer [33]. For example if user wants that the load balancer which are used by them should have highest throughput then they should use NGINX load balancer, while a user want that that the load balancer should have lowest latency then they should use Gateway load balancer or HAProxy load balancer. In the similar way for every different performance there are different load balancer, one load balancer could not contain all the techniques. For example hardware load balancers has the lowest error rate but their costs are very high and also the

Fig 2:- Comparison Graph For Throughput of Round Robin and Least-Connection Algorithm by comparing the throughputs of both the algorithms.

This graph basically compares the speed of the load balancing algorithms i.e round robin algorithm and least-connection algorithm.

So to calculate the speed of the load balancing algorithm we will need to calculate the throughput of both the algorithms.

For Round-Robin algorithm:-

*First we will need to calculate the number of requests that is n and we will store it.

*Then we will calculate the current server to send the requests, for this the mathematical equation is:-

$i=(i+1) \bmod n,$

here, i is the index of the current server and mod is the modulo operator, the whole equation indicates that the index is always within the range.

*Then it sends requests to the server at index i.

For example:- If there are three servers available & the servers had sent the requests to the server 1 and server 2 then the next request will be sent to the server 3 and if a new request comes then it will be transferred to server 1. This is how, round-robin algorithm works.

For Least-Connection Algorithm:-

* The speed of least connection algorithm can be calculated in the terms of number of requests it received. throughput rate is low. So one needs to very much clear that which load balancer they should use as if they will not choose the correct load balancer then their work will not be adequate.
* This algorithm has a complexity of O(1).

* The speed of this algorithm depends on the configuration of hardware and the software.

* For example-

If the load balancer has a processing capacity of 10000 requests, and the average response time is 10 ms. The algorithm can handle upto 100,000 requests.

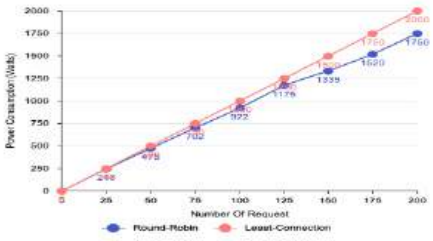


Fig 3:- Comparison Graph For Power Consumption by the comparison of both the load balancers.

So, speed of least connection algorithm is good than speed of round-robin algorithm. The same result can also be seen in the above graph that in the same time frame the speed of the least connection algorithm is better than round-robin algorithm compared to the number of requests.

This graph compares the power consumption of round robin and least connection algorithms.

Mathematical equation for power consumption is calculated as :-

$Ec=E_{idle}+(E_{max}-E_{idle})\times U$

Here E_c is the total energy consumption, E_{idle} is the energy consumption during the load balancer is idle, Emax is the energy consumption during the peak, and U denotes the the utilization of the resources of the host.

For Round-robin algorithm:-

Power consumption of round-robin algorithm depends on certain factors such as the number of servers, time slice of each server, the network bandwidth etc. Power consumption also depends on the context switching, context switching refers to the process of saving the state of a running process, restoring the state of next process and then resuming the execution.

For Least-Connection Algorithm:-

In this algorithm also power consumption depends on certain factors such as number of servers, network bandwidth, load distribution, etc. The power consumption of least-connection algorithm can be optimized by selecting an appropriate number of servers, ensuring a balanced load distribution. This algorithm can be more power efficient as compared to the round robin as it avoids overloading to an extent and can also distribute load more efficiently.

The same can be observed in the above graph, as the number of requests increased power consumption of the round-robin load balancer also increases. The power consumption of the least-connection load balancer remains less than round robin as compared with the number of requests because its ability to distribute the loads.

This table compares the types of load balancing algorithms and provides the details about the type of load balancer, advantages of the load balancer, disadvantages of the load balancer and the use cases of the load balancers. By the comparison of all the types of the load balancers we can find the best load balancer for a certain type of work. For example- if we are have to handle a very large web application, then we will require a least connection load balancer, it can easily handle a large web application, for every kind of work we cannot use same load balancer, according to the needs we will have to use load balancers, for a high or complex traffic application we will have to use a content-based algorithm. Using Dynamic type of load balancers is also advisable. So, with the help of this comparison table it is suggested that using Dynamic type of load balancers is advisable, so among this type of algorithms it is better to use least-connection algorithm because it's power consumption is low, it is having very less delay time and it is also faster than the algorithms stated here. It also distributes, load in an efficient manner, so by taking care of all the factors it is concluded that least-connection algorithm is the best algorithm of the load balancers which are stated in the above comparison table.

7. Limitations of This Review Paper:-

The major aspect which this review paper could not acknowledge is that what is the most advanced load balancing techniques, as according to the type of work we can choose the type of load balancer which we should use, but there must be a kind of load balancer which can be used for every purpose, but it is not possible as it will increase the cost of the load balancers, if we integrate different algorithms of a load balancer. I am adding some more points for defining the same:-

1. Applicability to specific contexts: Load balancing algorithms & techniques can work differently depending on the specific application, network architecture, workload, and user behavior. Therefore, the findings and recommendations of the study may not generalize to other contexts, and may require further testing and customization.
2. Availability of data and resources: Load balancing research often depends on simulations, experiments and real-world data, which may have limitations in terms of accuracy, availability, or representativeness. Therefore, the results & conclusions of the study may be subject to sampling errors, bias, or uncertainty.
3. Rapidly evolving technology: Load balancing techniques and algorithms are constantly evolving and adapting to new hardware, software, and network technologies. Therefore, the findings and recommendations of the study may become outdated or obsolete in a short period of time, and may require regular updates and revisions.
4. Interactions with other components: Load balancing is often part of a larger system or infrastructure, which may include other components such as servers, databases, caches, firewalls, and routers. Therefore, the performance and effectiveness of load balancing may depend on the interactions and dependencies between these components, which may be complex and difficult to model or control.

Table1:-Comparison Table For Algorithms According to This Review Paper.

Algorithm	Type	Pros	Cons	Use Cases
Round Robin	Static	Simple,fair, easy to implement	Not adaptive to workload, may cause overloading or underutilization	Low-Latency,non-critical application
Weighted Round Robin	Static	Allows different weights for different servers, more flexible	Still not adaptive, may require manual turning	Hetrogeneous applications with varying resource demands
Least Connection	Dynamic	Balances load based on the number of active connections	Not suitable for short-lived connections	Large-scale web applications
Random	Static	Simple and efficient can handle high traffic.	Non optimal for specific requirement of quality of service	Non-critical or low priority application
Least Time	Dynamic	Measures server response time and assigns task to the fastest server.	Requires up-to-date performance metrics	Latency-sensitive and resource-intensive application
Content-based	Dynamic	Routes traffic based on the content or type of requests	Requires deep packet inspection, may increase complexity	Application with variable workloads and complex traffic patterns

8. Conclusion

For our proposed system we had reviewed many research papers & proposed our algorithms that are based on many

types of load balancers such as software load balancer, hardware load balancer, hybrid load balancer. Load balancing is a very major area in the field of the cloud computing so it is a very important aspect in this field & it also reduces the main compile time of the whole process going on behind the scene. So, there are several approaches or rather algorithms that has been advised in this paper so that the jobs scheduled migrating of the resources & the resource utilisation, & many more would becomes very easy.. According to the needs of an individual one can select which combination of load balancers they can use.
For instance ,

* Layer 4 and Layer 7, these are best combination of load balancers, layer 4 can manage the simple traffic distribution, while layer 7 can manage the advanced applications for the same.
* Software and Hardware Load Balancers, these are also one of the best combinations of load balancers as hardware load balancers can be used for higher performance and a software load balancer can be used for more flexibility and to lower the cost.
* Active-Passive load balancers, these are best load balancing algorithms as one of the load balancers is on active mode and one is on standby, so the process never fails, if required the standby load balancer will also take over, when there will be more traffic. So according to the research paper advised by us it will become very easy for the users to choose which load balancing algorithms or techniques they want to use, which is more beneficial for them and also cheaper. This review paper would also provide a great understanding for the researchers to research upon various facts and algorithms of the load balancers.

9. Future work

In the proposed paper we basically researched and did the review of the paper in the three major algorithms and categories:
Hardware, Software & Virtual load balancer. After that the researchers can further research upon the algorithms to make the algorithms more advance & more clever. The fault tolerance of the algorithms should also be improved so that the issues can be solved and the algorithms should take less time to be executed. Without the cloud computing techniques all the algorithms of load balancing are not valid, so later with the advancement of the cloud computing the algorithms of the load balancing will also be increased.

10. References

[1] Agarwal, M., Srivastava, D.G.M.S., 2017. Cloud Computing: A Paradigm Shift in the Way of Computing. Int. J. Mod. Educ. Comput. Sci. 9 (12), 38–48. https://doi.org/ 10.5815/ijmecs.2017.12.05.

[2] Ali, S. et al., 2020. Towards pattern-based change verification framework for cloud- enabled healthcare component-based. IEEE Access 8, 148007–148020. https://doi.org/10.1109/ACCESS.2020.3014671.

[3] Mala, Y., Prasad, J.S., 2018. A review on load balancing algorithms in cloud computing environment. Int. J. Comput. Sci. Eng. 6 (8), 771–778. https://doi.org/ 10.26483/ijarcs.v9i2.5837.

[4] Agarwal, M., Srivastava, G.M.S., 2019. A PSO algorithm based task scheduling in cloud computing. Int. J. Appl. Metaheuristic Comput. 10 (4), 1–17.

[5] Alhammadi, A.S.A., Vasanthi, V., 2020. Multiple regression particle swarm optimization for host overload and under-load detection. Test Eng. Manage. 82, 10253–10261.

[6] Gomathi, B., Krishnasamy, K., 2013. Task scheduling algorithm based on hybrid particle swarm optimization in cloud computing environment. J. Theor. Appl. Inf. Technol. 55 (1), 33–38.

[7] Jena, U.K., Das, P.K., Kabat, M.R., 2020. Hybridization of meta-heuristic algorithm for
load balancing in cloud computing environment. J. King Saud.

[8] Kennedy, J., Eberhart, R.C., 1995. Particle swarm optimization. In: Proceedings of the IEEE International Conference on Neural Networks. IEEE, pp. 1942–1948.

[9] Kumar, M., Sharma, S.C., 2017a. Deadline constrained based dynamic load balancing algorithm with elasticity in cloud environment. Comput. Electr. Eng., 1–17 https://doi.org/10.1016/j.compeleceng.2017.11.018.

[10] Mishra, S.K., Sahoo, B., Parida, P.P., 2018. Load balancing in cloud computing: a big picture. King Saud Univ.– Comput. Inf. Sci. 32 (2), 149–158.

[11] Zhong, Z., Chen, K., Zhai, X., Zhou, S., 2016. Virtual machine-based task scheduling algorithm in a cloud computing environment. Tsinghua Sci. Technol. 21 (6), 660–667.

[12] Ramezani, F., Lu, J., Hussain, F.K., 2013. Task-based system load balancing in cloud computing using particle swarm optimization. Int. J. Parallel Prog. 42 (50), 1–16 .

[13] Rimal BP, Choi E, Lumb I (2009, August) A taxonomy and survey of cloud computing systems. In: Fifth international joint conference on INC, IMS and IDC, 2009. NCM'09, pp 44–51.

[14] Borovskiy V, Wust J, Schwarz C, Koch W, Zeier A (2011) A linear programming approach for optimizing workload distribution in a cloud. Cloud Comp:127–132.

[15] Dave A, Patel B, Bhatt G (2016, October) Load balancing in cloud computing using optimization techniques: a study. In: International Conference on Communication and Electronics Systems (ICCES), pp 1–6.

[16] Jamshidi P, Ahmad A, Pahl C (2013) Cloud migration research: a systematic review. IEEE Trans Cloud Comp 1(2):142–157

[17] Raviteja S, Atmakuri R, Vengaiah C (2017) A review on cloud computing migration and issues

[18] Milani AS, Navimipour NJ (2016) Load balancing mechanisms and techniques in the cloud environments: systematic literature review and future trends. J Netw Comput Appl 71:86–98

[19] Kalra M, Singh S (2015) A review of metaheuristic scheduling techniques in cloud computing. Egypt Inform J 16(3):275–295

[20] Adhikari M, Amgoth T (2018) Heuristic-based load-balancing algorithm for IaaS cloud. Futur Gener Comput Syst 81:156–165.

[21] Gulati A, Chopra RK (2013) Dynamic round robin for load balancing in a cloud computing. IJCSMC 2(6):274–278.

[22] Uk Jena, P.k Das (2022) Hybridization of meta-heuristic algorithm for load balancing in cloud computing environment 1(2):165-157

[23] Borst S. (2022). Load balancing in large-scale heterogeneous systems. Queueing Systems: Theory and Applications. **100**:3-4. (397-399).
[24] der Boor M, Borst S, Van Leeuwaarden J and Mukherjee D. (2022). Scalable Load Balancing in Networked Systems. SIAM Review. **64**:3. (554-622).
[25] X. Liu and L. Ying. Steady-state analysis of load balancing algorithms in the sub-Halfin-Whitt regime. J. Appl. Probab., 57(2):578 -- 596, June 2020.
[26] Daniel Appel(2023) A narrow band-based dynamic load balancing scheme for the level-set ghost-fluid method, January 2023.
[27] Fangcheng Tang A hybrid multi-objective approach for real-time flexible production scheduling and rescheduling under dynamic environment in Industry 4.0 context, August 2021.
[28] Subhranshu Sekhar Tripathy (2023) State-of-the-Art Load Balancing Algorithms for Mist-Fog-Cloud Assisted Paradigm: A Review and Future Directions.
[29] P.A Laplante, Zhang Jia and J. Voas, "What's in a Name? Distinguishing between SaaS and SOA", *IT Professional*, vol. 10, no. 3, pp. 6-50, May-June 2008.
[30] Gutierrez-Garcia J, Trejo-Sánchez J and Fajardo-Delgado D. (2023). Agent coalitions for load balancing in cloud data centers. Journal of Parallel and Distributed Computing. **172**:C. (1-17). Online publication date: 1-Feb-2023.
[31] Server and data center energy efficiency, Final Report to Congress, U.S. Environmental Protection Agency, 2007.
[32] V. K. Adhikari, S. Jain, and Z.-L. Zhang. YouTube traffic dynamics and its interplay with a tier-1 ISP: An ISP perspective. In ACM IMC, pages 431--443, 2010.
[33] S. Dhakal, M. M. Hayat, J.E.Pezoa, C. Yang, and D. Bader, "Dynamic Load Balancing in Distributed System in the Presence of Delays: A Regeneration-Theory Approach,", IEEE Transactions on Parallel and Distributed Systems, vol. 18, no. 4, April 2007.
[34] Wickremasinghe, Bhathiya, Rodrigo N. Calheiros and Rajkumar Buyya, "Cloudanalyst: A cloudsim-based visual modeller for analysing cloud computing environments and applications".