FUDMA Journal of Sciences (FJS)
ISSN online: 2616-1370
ISSN print: 2645 - 2944
Vol. 9 No. 12, December, 2025, pp 35 – 44
DOI: https://doi.org/10.33003/fjs-2025-0912-3939

EFFICIENT LOAD BALANCING TECHNIQUES FOR MOBILE FOG NETWORKS IN THE FOG LAYER: A REVIEW

*[1]Agber Selumun, [2]Abdullahi Muhammad Bashir, [2]Bashir Sulaimon Adebayo and [2]Ojerinde Oluwaseun Adeniyi

[1]Department of Mathematics and Computer Science, Rev Fr Moses Orshio Adasu University Makurdi (Formerly Benue State University, Makurdi), Benue State.
[2]Department of Computer Science, Federal University of Technology Minna, Niger State.

*Corresponding authors' email: seseagber@gmail.com

ABSTRACT
The proliferation of Internet of Things (IoT) devices has significantly increased the demand for cloud services, leading to the emergence of Mobile Fog Networks (FogNets) as an effective paradigm for edge processing. This paper reviews efficient load-balancing techniques within the Fog layer, focusing on their ability to handle increasing workloads and ensure optimal resource utilization. A total of forty (40) peer-reviewed articles published between 2020-2025 were systematically reviewed using a descriptive and comparative analysis approach. The findings reveal that heuristic-based, hybrid, and machine learning driven load-balancing methods demonstrate superior performance in terms of response time, energy efficiency, and latency reduction. However, most existing studies rely on simulation-based evaluation and lack real-world deployment validation. The review concludes that future research should emphasize adaptive, context-aware, and energy-efficient load-balancing strategies to enhance real-time service delivery in FogNets.

Keywords: Internet of Things (IoT), Cloud Computing, Fog Networks (FogNets), Load Balancing, Fog Layer, Mobile Fog Networks and Fog Computing Environments

INTRODUCTION
In recent years, various smart devices, including wearable technology, smartphones, and industrial components, have been equipped with sensors to monitor and collect real-time physical data from their surroundings. Wireless sensor networks offer an effective framework for organizing and managing these sensor nodes (Vijarania et al., 2023). However, the significant traffic generated by Internet of Things (IoT) nodes often leads to network congestion and server overload. To address this challenge, an intermediary fog layer has been introduced as a bridge between IoT devices and the cloud. This layer helps minimize response delays and latency. By offloading some of the heavy computational workloads, the fog layer alleviates the burden on lower network layers, ensuring more efficient data processing and communication (Vijarania et al., 2023).

Fog computing complements cloud computing by enabling support for delay-sensitive IoT applications while enhancing mobility, geo-distribution, and location awareness. It reduces network bandwidth usage by minimizing traffic between end devices and the cloud (Gu et al., 2018). Additionally, it strengthens security and privacy by pre-processing and encrypting data closer to its source. Fog resources span the entire continuum from the edge of the network to the cloud, whereas edge computing confines these resources within a single-hop distance from the devices. As a result, fog computing is more complex than edge computing, with the edge functioning as a subset of the fog (Ebrahim & Hafid, 2023). Distributed fog deployments are feasible for modular applications, such as workflows and bag-of-tasks, which can be divided into pipelined workflows. Conversely, monolithic applications cannot be partitioned into separate logical modules, requiring them to operate as a single module on a single computing entity (Ebrahim & Hafid, 2023).

According to Gupta & Singh (2022), the Internet of Things (IoT) can enable numerous data transfers between numerous servers. Fog servers are used in the Internet of Things to provide efficient data transfers from dynamic devices. However, researchers continue to place a strong emphasis on minimizing the load balancing problem. A faulty scheduling policy could cause some virtual machines to be overwhelmed while other virtual machines are idle. A novel hybrid Grey Wolf Optimization (GWO) with the Modified Moth Flame method was used in the researchers' suggested model, which was based on dynamic load balancing in a fog-IOT environment (MMFA). Additionally, the GMFA significantly contributed to the improvement of Deep Reinforcement Learning (DRL). The GMFA algorithm was used to improve the performance of the actor-critic based deep reinforcement learning (DRL) approach, and the combined approach was given the moniker GMFA-DRL. When it comes to resource allocation problems, RL has a number of advantages, and simulations show that it outperforms reactive approaches. The Python-based platform Jupyter was used to implement the suggested GMFA-DRL methodology. Performance matrices including Throughput, Latency, Makespan, Load Balancing Level (LBL), and Energy Consumption were used to assess the performance. The simulation results show that the suggested model produces outcomes for load balancing, low energy usage, low latency, and good throughput. As a result, it can be demonstrated that the suggested methodology is more effective than the current one.

By utilizing hybrid load balancing and a distributed environment, Kadhim & Manaa (2022) presented a fog-to-server architecture based on the IoT to address the issue of packet loss in fog and servers. The suggested approach was based on a hybrid form of load balancing that incorporated least connection and weighted round robin algorithms in fog nodes to account for the workload and the amount of time needed to distribute requests to the active servers. The results show the proposed system improved network evaluation parameters such as total response time of 131.48 ms, total packet loss rate of 15.670%, average total channel idle of 99.55%, total channel utilization of 77.44%, average file transfer protocol (FTP) file transfer speed (256 KB to 15 MB files) of 260.77 KB/sec, and average time (256 KB to 15 MB) of 19.27 sec.

According to Vijarania et al. (2023), the Internet of Things (IoT) and cloud computing have significantly transformed the technological landscape, impacting various aspects of daily life. However, the rapid growth of IoT and cloud applications has exposed limitations in the traditional cloud model, such as network instability, reduced bandwidth, and high latency. To address these challenges, fog computing has emerged as a solution that bridges IoT devices and cloud computing by deploying fog nodes closer to the data sources. This approach helps reduce network traffic, latency, and the load on cloud servers. In their study, the researchers highlighted various IoT–fog–cloud models designed to distribute the load uniformly across the network. They proposed an efficient load balancing mechanism utilizing fog computing to optimize the distribution of tasks among fog devices. A performance evaluation of the proposed solution against existing techniques demonstrated improvements in performance metrics such as energy consumption, throughput, response time, and resource utilization. These findings underline the effectiveness of the proposed strategy in overcoming the limitations of the traditional cloud model and enhancing overall system efficiency.

According to Ebrahim & Hafid (2023), the proliferation of IoT applications for smart environments has been significantly enabled by cloud computing. However, the inherent distance of cloud resources from end devices makes them unsuitable for delay-sensitive applications. To address this limitation, fog computing has emerged as a solution by deploying distributed resources closer to end devices. These resources can collaborate to support distributed IoT application workflows through the use of stateless micro fog service replicas, which enhance resiliency and maintain service availability despite failures. Load balancing plays a critical role in this collaboration by optimally distributing workloads among fog nodes. This ensures fair utilization of compute and network resources while minimizing execution delays. They proposed using ELECTRE, a Multi-Criteria Decision Analysis (MCDA) approach, to achieve efficient load balancing in fog environments. Their method incorporates multiple objectives, such as compute and network load information, to guide service selection decisions. The proposed approach was evaluated in a realistic unbalanced topology with heterogeneous workload requirements. To the best of their knowledge, this is the first use of ELECTRE-based methods for load balancing in fog environments. Through simulations, the researchers compared their approach with traditional methods, including random selection, Round-Robin, nearest node, and fastest service algorithms. The results demonstrated that their approach achieved up to a 67% improvement in overall system performance, outperforming the baseline methods significantly.

Keshari et al. (2022), Intelligent Traffic System (ITS) is improved by vehicular fog computing (VFC), which computes real-time traffic data for accident alarms, route routing, and other purposes. Instead of using the main cloud for processing, it uses stationary and moving cars as fog nodes. Due to sort distance, VFC uses less bandwidth, responds more quickly, and causes less core cloud congestion. In accordance with the computational resources that are available, VFC executes resource allocation to choose the ideal vehicle for computation. Because direct communication is constrained by the transmission range of the vehicle, it executes data retrieval procedures to convey traffic data from car to vehicular fog. When allocating resources and retrieving data, VFC encounters a number of problems due to the high mobility of the vehicles, dynamic topology, and other factors. Due to ineffective resource allocation and impractical data retrieval, this results in uneven resource distribution and delayed data delivery. Due to sharing in vehicular fog, the security and privacy of traffic data are threatened to leak. We give in-depth exploration at each stage of the VFC procedure to address these problems. As a result, we introduce a new classification of VFC operation that categorizes the difficulties in each phase, including resource allocation in task division, scheduling, and load balancing; result retrieval in phase, node selector, node selection, and path recovery; and secure data sharing in authentication, encipherment, auditing, and data privacy. With this state-of-the-art, future directions for handling computing at VFC efficiently and open research questions are better understood.

According to Keshari et al. (2022), In order to balance the load among various servers, a process known as load balancing (LB) is used. The fog server is managing the bulk of the cloud server's data to expedite user queries. Data accessibility has increased because to fog computing and the exponential growth in data requests. Many issues are overcome by fog computing in accordance with user requests, while some still call for further development. The issue with fog computing is LB as a result of a rise in network layer traffic. Numerous LB approaches have already been put forth in the cloud layer, but they have only ever been used in the fog layer up until this point. Service quality issues, including as delays in response and processing times, security breaches, and other issues, may be brought on by ineffective LB. They examined a number of algorithms in their survey that are based on LB and address the problem of network overloaded data. Other authors have concentrated on LB aspects such as latency, bandwidth, deadlines, cost, security, execution time, and reaction time. The quality parameter table and algorithm for additional parameters based on fault tolerance are also explained.

Kanbar & Faraj (2022), in their research stated that a new technology for a large-scale setting is cloud computing. Thus, it encounters a variety of difficulties, the main one being load balancing, which reduces the efficiency of the computer resources. To address these issues, they proposed the RADISH (Region Aware Dynamic Scheduling) model, which consists of five sequential processes, the first of which uses a task nature-based bi-class neural network to classify incoming tasks as sensitive or non-sensitive while taking into account factors such as login information, email addresses, passwords, service types, and QoS parameters to shorten scheduling latency. The second step uses a multi-criteria based improved moth flame optimization (QoS aware AMFO) to schedule the classified jobs while taking sensitive, non-sensitive, energy, priority, completion time, and work load into account. Due to this algorithm's strong convergence, scheduling lag was decreased. They perform load balancing in the third process by recommending SAC with a prospective field clustering approach. The VM clustering is taken into account while calculating local potential, energy, and density. To balance the server load and increase the effectiveness of the procedure, three repositories are created. In order to reduce allocation latency and enhance QoS, they introduced the VM state-aware Hopcroft-Karp algorithm in our task allocation proposal. We accomplish good SLA and QoS in an IoT fog multi-cloud system in this way. The CloudSim simulation program runs the simulation and assesses performance in terms of latency, bandwidth, completion time, throughput, energy consumption, CPU and memory utilization, SLA violations, and overhead.

According to Hameed et al. (2021), IoT-enabled clusters of automobiles offer a robust source of computational resources while enabling efficient collaboration through vehicle-to-vehicle and vehicle-to-infrastructure communication. This is facilitated by vehicular fog computing, where vehicles function as fog nodes to provide cloud-like services to IoT applications and integrate with traditional cloud systems to collaboratively complete tasks. However, managing loads efficiently in vehicular fog computing presents challenges due to the dynamic and decentralized nature of vehicular ad-hoc networks (VANETs). To address these challenges, the researchers proposed a cluster-enabled, capacity-based load-balancing approach designed to enhance energy efficiency and performance in vehicular fog distributed computing for processing IoT jobs. The approach incorporates a dynamic clustering mechanism that considers vehicle position, speed, and direction to form clusters, which serve as pools of computing resources. It also includes a predictive mechanism to estimate a vehicle's departure time from a cluster, facilitating the anticipation of its future position within the dynamic network. Additionally, the study introduced a capacity-based load distribution strategy to balance workloads both within individual clusters (intra-cluster) and across multiple clusters (inter-cluster) in the vehicular fog network. Using the NS2 network simulation environment, the proposed approach was evaluated and found to achieve balanced energy consumption, reduced network delays, and improved network utilization, thereby demonstrating its effectiveness in managing vehicular fog computing tasks.

Mondragón-Ruiz et al. (2021), in their research showed the development of a distributed fog computing architecture for the deployment of IoT applications. Their study shows how these architectures optimize the distribution of resources throughout the entire deployed platform, in addition to considerably reducing latency. On the one hand, regarding resource distribution, they observed that by deploying critical data analysis and decision-making applications (CEP and the MQTT Broker, in our case), the values of the evaluated metrics are reduced considerably (CPU consumption, RAM memory and power consumption) on the cloud server, with the consequent savings for the cloud provider. Specifically, the fog computing approach enables a reduction of RAM consumption up to 35% and energy up to 69% at the core level, since it fully exploits the computational resources of fog nodes. On the other hand, regarding latency, the work highlights how a fog computing architecture considerably reduces latency with respect to cloud computing, up to 35% better.

According to Kashani & Mahdipour (2023), fog computing has emerged as a modern distributed paradigm that complements cloud computing by providing services closer to the edge of the network. This extension addresses challenges in delay-sensitive applications by enabling location awareness and mobility support. Additionally, fog computing plays a crucial role in enhancing service computing by reducing latency. A key aspect of fog networks is load balancing, which ensures that fog nodes are neither underutilized nor overloaded, thereby optimizing system performance. Effective load balancing improves quality of service (QoS) parameters, including resource utilization, throughput, cost efficiency, response time, performance, and energy consumption. While significant research has been conducted on load-balancing algorithms in fog networks, a systematic consolidation of these efforts has been lacking. They provided a comprehensive review of load-balancing algorithms, classifying them into four categories: approximate, exact, fundamental, and hybrid algorithms. The authors also analyzed the metrics used in load balancing, highlighting the advantages and disadvantages associated with each type of algorithm. Furthermore, the study explored evaluation techniques and tools applied in previous research and identified critical open challenges and future trends in load-balancing algorithms for fog networks. By systematically investigating these aspects, the study contributes to the development of efficient load-balancing strategies in fog computing environments.

According to (J. Singh et al., 2022), the proliferation of client applications on the fog computing layer, driven by advancements in the Internet of Things (IoT), has underscored the importance of fog computing in reducing latency and optimizing resource utilization for IoT tasks. Despite its numerous benefits, fog computing faces several challenges, including resource overloading, security, node placement, scheduling, and energy consumption. Among these, load balancing remains a critical challenge due to the growing number of IoT devices and requests, necessitating an even distribution of workloads across available resources. To address these challenges, the researchers proposed a secure and energy-aware fog computing architecture integrated with a Software-Defined Networking (SDN)-enabled fog environment. The study introduced a load-balancing technique aimed at optimizing resource utilization and reducing delays. Additionally, a Deep Belief Network (DBN)-based intrusion detection system was implemented to enhance security and minimize workload communication delays within the fog layer. The simulation results demonstrated the efficacy of the proposed approach. The technique achieved significant improvements in key performance metrics, including a 15% reduction in average response time, a 23% reduction in average energy consumption, and a 10% reduction in communication delays compared to existing methods. These findings highlight the potential of the proposed architecture to enhance the performance and efficiency of fog computing environments.

Abdussami & Farooqui (2020) compiled a list of similarities and differences between cloud computing and fog computing, which may differ in terms of design, tools, configurations, and services offered to associations and customers. As a consequence of this comparison, FC was shown to perform satisfactorily in terms of data processing service and low bandwidth use.

According to Pereira et al. (2020), the rapid growth and heterogeneity of Internet of Things (IoT) devices present significant challenges for Fog Computing, particularly in maintaining efficiency as fog nodes frequently become overloaded. This overload adversely affects response times, which is a critical performance metric in fog environments. To address this issue, Pereira et al. (2020) proposed an innovative architecture model for Fog Computing, accompanied by a Priority Load Balancer designed to enhance the efficiency of fog nodes. The proposed load balancer utilizes task-specific information and computational dynamics to optimize the allocation of resources and reduce response times effectively. Simulation results demonstrated that the proposed solution significantly outperformed traditional strategies, such as direct and round-robin approaches. Notably, the Priority Load Balancer achieved a reduction of over 56% in the response time for high-priority tasks compared to existing load-balancing methods. These findings underscore the effectiveness of the proposed architecture in addressing the challenges of IoT heterogeneity and workload management in Fog Computing environments.

To ensure the overall efficiency of a fog network, it is essential to manage available resources such as memory, processors, and bandwidth effectively. Proper resource management prevents scenarios where some devices become overloaded while others remain underutilized. Such imbalances can lead to bottlenecks, increasing communication latency across the network. To address this, resource allocation algorithms are employed to optimize network performance and maintain a high quality of service (QoS) for users. These algorithms ensure the efficient utilization of available resources by facilitating effective data and process exchanges among the devices in the network, thereby enhancing overall system performance.

Hence The foundation of both the cloud and the fog is made up of elastic resources, such as compute, storage, and networking (Das & Inuwa, 2023). Fog computing is a non-trivial expansion of the cloud due to a number of distinctive features.

i. ***Edge location:*** The fog has the ability to support latency-sensitive applications that need real-time data processing because of its proximity to end users.
ii. ***Location awareness:*** The services provided by the fog are widely dispersed, in sharp contrast to the centralized cloud. In order to support mobility, the geographically dispersed fog nodes can determine their locations and monitor the devices of end users.
iii. ***Real-time interactions***: as opposed to bulk processing, are a feature of fog applications.
iv. ***Edge analytics***: In the age of big data, local analysis of sensitive data via fog computing provides an alternative to transferring such data to the cloud. v. Scalability: If data created by end devices is constantly sent to the cloud, it may become the bottleneck. In order to overcome the scalability issue brought on by the proliferation of end devices in the IoT, fog computing helps reduce the strain of centralized processing (Bajaj et al., 2022).

The motivation for this review arises from the growing need to optimize workloads and resource allocation in mobile fog environments. Previous studies have demonstrated that poor load distribution can lead to inefficient resource utilization and energy wastage, ultimately causing fog node failures. Hence, there is an urgent need to identify and evaluate effective load-balancing techniques that can enhance energy efficiency and service reliability in FogNets.

This paper makes several key contributions. First, it conducts an extensive review of forty (40) peer-reviewed articles on load-balancing approaches in fog computing, comparing them based on the algorithms, tools, and simulation platforms used, as well as their respective strengths and limitations. Second, it presents a taxonomy of existing load-balancing strategies within the fog environment. Third, it provides a comparative summary of recent results, identifies existing research gaps, and highlights emerging trends in fog load balancing. Finally, a conceptual load-balancing architecture is proposed to address identified limitations and guide future research directions.

The remainder of this paper is organized as follows: Section 2 presents the research methodology adopted for this review, including the selection criteria and analysis approach used in evaluating the reviewed studies. Section 3 reviews existing surveys and discusses various load-balancing strategies employed in fog computing environments. Section 4 provides a comparative analysis and taxonomy of the identified techniques. Section 5 highlights key research challenges, open issues, and future directions, while Section 6 concludes the paper with final remarks and recommendations.

MATERIALS AND METHODS

Numerous research publications have been read, research queries have been created, and various databases have been searched as part of this review process. In order to find relevant papers for this study, a variety of sources were searched, including IEEE Explore, Science Direct, Springer, Google Scholar, the ACM Digital Library, and Elsevier. In order to respond to the questions raised by this review strategy, these papers are examined. This article details the peer review of the research articles that was completed. Initial research has revealed areas for improvement in this article. Table 1 below represents the total number of Search Results on Digital Libraries

Table 1: Search Results on Digital Libraries

No	Academic Database	Result
1	Scopus	470
2	IEEE	231
3	ScienceDirect	190
4	ACM	711
5	Springer	90
6	Google Scholar	785
	Total	2,477

Search Parameters

To locate the articles in the databases covered in the review plan, we employed a few keywords. The search was done using the terms “Load balancing in fog networks”, "Load balancing approaches", "Fog computing" and "Fog load balancing". The time-consuming and widespread way of searching in databases is database searching. Also included in our search were the terms "resource utilization," "task migration in VM," "workload on fog," and "load distribution," which are some additional commonly used terms for load balancing in fog computing. This approach was used for the research paper's title and abstract. We took into account studies from 2020 to 2025. Only those papers that met our assessment criteria were taken into consideration after we completed a quality assessment check on the papers that were searched.

Quality Assessment Check

To ensure the credibility and relevance of the reviewed studies, a quality assessment process was conducted following the criteria outlined by Kaur & Aron (2021). Initially, a total of 2,477 articles were retrieved from six major digital libraries, as presented in Table 1. The reduction to 300 articles in the first stage was achieved by screening the titles and keywords to eliminate duplicates, non-English publications, and papers that were clearly unrelated to load balancing or fog computing.

In the second stage, abstracts and conclusions were reviewed to determine whether the studies explicitly addressed load-balancing techniques or resource management strategies within fog or edge computing environments. This refinement reduced the number of relevant articles to 200.

Subsequently, full-text evaluations were performed to assess the methodological rigor, relevance to fog computing, and the presence of performance metrics (such as response time, latency, energy efficiency, and resource utilization). Based on this detailed screening, 100 articles were retained.

A final quality assessment was then applied using the following criteria, adapted from Kaur & Aron (2021):

i. ***Relevance***: The study directly focuses on load balancing or resource allocation in fog/edge computing environments.
ii. ***Scientific Contribution:*** The paper presents an algorithm, model, framework, or comparative analysis relevant to fog computing.
iii. ***Methodological Soundness:*** The research demonstrates a clear methodology, including evaluation or experimental validation.
iv. ***Publication Quality:*** The article is peer-reviewed and published in a recognized journal or conference.
v. ***Recency:*** The publication falls within the selected review period of 2020-2024.

After applying these criteria, 50 studies were found to be of sufficient quality and relevance. Following a final round of detailed analysis and exclusion of duplicates or incomplete studies, 40 papers were retained for this systematic review and synthesis.

RESULTS AND DISCUSSION

Fog computing, often referred to as fogging, differs from traditional cloud computing by emphasizing the decentralization of computational structures. Positioned between data-generating devices and the cloud, fog computing shifts computational capabilities closer to the network's edge. This approach enables faster services for users, including communication and software solutions. Fog computing is particularly advantageous for high-mobility technologies like the Internet of Things (IoT) and vehicular ad hoc networks (VANETs). Instead of relying on complex network infrastructures, devices in a fog computing setup are typically connected directly to their destination. This direct connection reduces latency and enhances the overall quality of service (Alwakeel, 2021). Fog computing environments use nodes that are dispersed across the environment and can gather information from sensors placed nearby. The fog layer itself processes and stores the real-time data that fog nodes receive. After processing data at the fog layer, responses are given to end users. The cloud is used to store and process data that will only sometimes be used**.**

Table 2: Inclusion /Exclusion Criteria

Inclusion	Exclusion
1. Research articles that present techniques or innovative solutions on load balancing mechanisms in fog computing	1. Review articles, editorial articles, short articles (less than six pages), white articles, and non-English articles
2. Peer-reviewed articles in conferences and JCR-indexed journal	2. Research articles that do not mention solutions and methods to
3. Articles published between 2020 and December 2025	3. Improve load balancing in fog computing explicitly
	4. Books, book chapters, and theses

Description

Fog computing is a decentralized computing infrastructure or process in which computing resources are located between the data source and the cloud or any other data center. Fog computing is a layered model for enabling ubiquitous access to a shared continuum of scalable computing resources. The model facilitates the deployment of distributed, latency-aware applications and services, and consists of fog nodes (physical or virtual), residing between smart end-devices and centralized (cloud) services.

Several definitions have been put out by various studies, however Cisco first used the phrase in 2012. Fog, in Cisco's opinion, moves cloud services closer to edge hardware. Instead of storing IoT device data in cloud data centers, fog computing is an architectural deployment of computing resources that allows for the exchange and transfer of data across distinct nodes (Atieh, 2021).

Fog computing has revolutionized modern communication by addressing the significant technical challenges and complexities associated with cloud computing. However, it remains vulnerable to numerous security and privacy threats related to both data and services. Due to unique characteristics such as geo-distribution, mobility, and heterogeneity, traditional security and privacy measures designed for cloud computing are often inadequate for fog computing environments. Consequently, advanced, state-of-the-art security mechanisms are essential to address these challenges effectively. While fog computing offers several advantages over traditional cloud systems, certain security issues can hinder its widespread deployment in modern applications. Various research works have explored these security concerns, with some focusing on overarching issues, while others delve into specific aspects of fog computing applications or architecture security (Alwakeel, 2021).

Architecture of Fog Network

Fog computing architecture combines hardware, software, and logical and physical network components to create a vast network of interconnected devices. The topology and protocols utilized, as well as the distribution of fog nodes (physical and geographical), are important architectural components of a fog architecture see figure 1 below. Fog architecture takes into account the division of tasks among multiple layers, the kinds and numbers of protocols employed, and the limitations set at various layers. The IoT layer, which houses various smart devices, is the initial layer of the traditional architecture for fog computing. The fog computing layer, which is the second layer, has fog nodes with limited computational and storage power. The highest layer, called the cloud layer, houses enormous data centers. The fog layer processes the data generated by IoT devices, and when the data has been processed, users receive an immediate response. Since the fog layer is set up close to IoT users, users of real-time applications can receive a prompt answer. The data that is received from the fog layer is kept for a long period in cloud data centers.

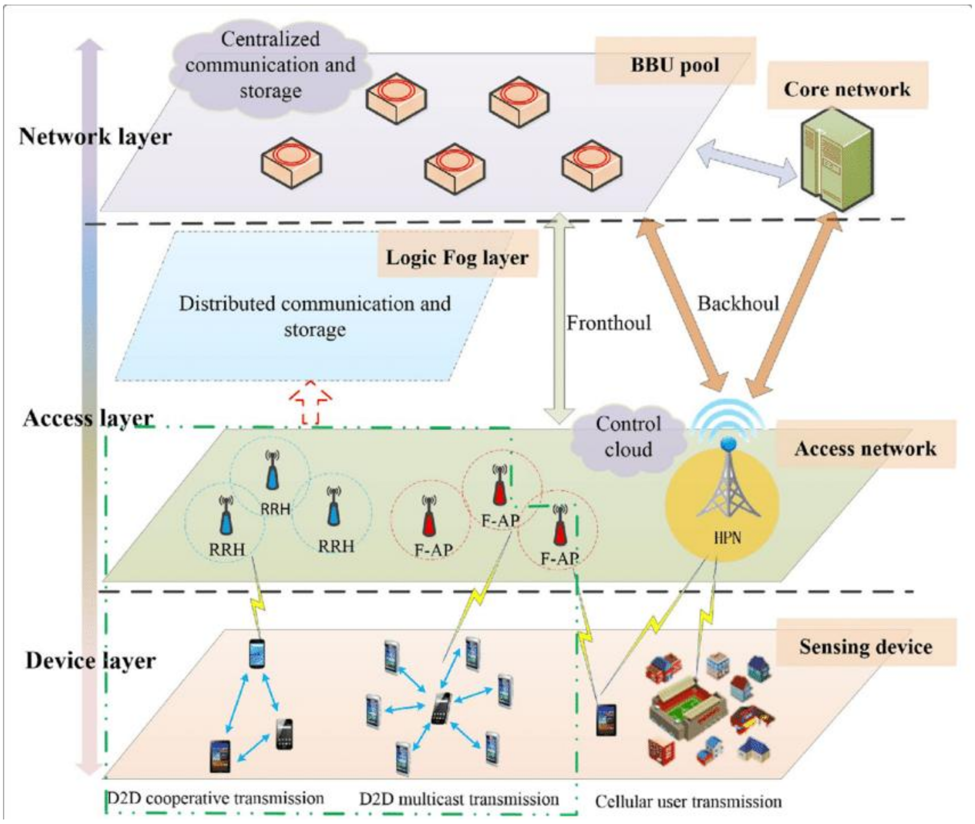

Figure 1: Fog Computing Network Architecture (Yan et al., 2018)

Load Balancing

By provisioning and de-provisioning instances of applications and optimizing resource usage, load balancing enables the fog system to distribute workload among resources equitably, intending to continue providing services even if the service component fails. In order to improve the performance of applications and network utilization in fog computing, an adequate system of load balancing is required because data centers procure differences between hosts and demonstrate unique aspects of traffic. Load balancing, as a mechanism, distributes the workload among multiple resources to prevent any overload or under-load on those resources. Load balancing, which divides the workload across several resources, is implemented in both hardware and software (Kaur et al., 2024).

Throughput maximization, reaction time minimization, and traffic optimization are some of the objectives of load balancing. Other goals of the load-balancing technique include reducing request processing time, increasing scalability in dispersed environments, and optimizing server-side resource use. The methods for load balancing in fog networks might be static, dynamic, or a combination of the two. Because user behavior is unpredictable and static load balancing methods are not always effective in the network, the rule should be coded in the load balancer when using static methods and primary system information as an essential feature. Furthermore, because of the dynamic load distribution based on the pattern pre-programmed in the load-balancer, dynamic approaches outperform static methods (Mishra & Majhi, 2020).

Types of Load Balancers

There are basically two types of load balancers in fog networks. This includes the static and dynamic load-balancer (Kanellopoulos & Sharma, 2022).

i. Static Load-balancer: In fog networks, static load balancing techniques distribute jobs without taking into account the system's state or metrics like the processor load level (Kashani & Mahdipour, 2023). Static algorithms are designed for systems with very little variation in load and distribute traffic equally among servers or in accordance with other principles that are not sensitive to system variables (Kanellopoulos & Sharma, 2022). For improved processor performance, static load balancing methods necessitate thorough knowledge of server resources at the time of implementation (Aldossary et al., 2025). See figure 2 below for a static load balancer.

Figure 2: Static Load Balancer (Aldossary et al., 2025)

ii. Real-time network communication is necessary for dynamic load balancing algorithms since they identify and favor the lightest server. Based on the system's current condition, the dynamic algorithm manages the load and shifts traffic from heavily used machines to underutilized machines in real-time (Kadry et al., 2022). See figure 3 below for a dynamic load balancer.

Figure 3: Dynamic Load Balancer (Kadry et al., 2022)

Dynamic load-balancing mechanisms apply the state of the systems today and employ unique policies, such as:

i. ***Transfer:*** It outlines the requirements for sending a job from one node to another. The processing or transfer of tasks that arrive and enter the transfer policy is determined by a rule that takes into account the workload of each node. The policy addresses task rescheduling and migration (Seraj et al., 2024).
ii. ***Selection***: This step determines whether a task should be transmitted or not, as well as how to choose a task based on factors such migration overhead, task execution time, and total number of non-local system calls (Ebrahim & Hafid, 2023).
iii. ***Location:*** Tasks are sent to under-loaded nodes that are defined by the system. The readiness of critical services for job rescheduling or migration is examined in targeted nodes.
iv. ***Information:*** This policy collects all available data, taking into account system nodes, and other policies use this data to make decisions. The timing of the information collection is determined by this policy (P. Singh et al., 2022).

The transfer policy is responsible for intercepting incoming tasks and determining whether they should be offloaded to a remote node to balance the workload or executed locally (Kavitha et al., 2023). Eligible tasks are transferred, while tasks that do not satisfy offload criteria are processed locally (Seraj et al., 2024). Once the transfer decision is made, the location policy identifies a suitable destination node by evaluating current load, availability, and proximity (Ebrahim & Hafid, 2022). If no optimal remote node is found, tasks remain local. Both the transfer and location policies rely on the information policy, which continuously gathers system state information (e.g. resource utilization, latency, bandwidth) to support dynamic and informed load-balancing decisions (Kashani et al., 2022).

Metrics for Balancing Loads

Several metrics are needed to assess a mechanism of load balancing and weigh it in comparison with mechanisms before to show which mechanism is better and to recognize the pros and cons related. The metrics need some qualitative paradigms. Various qualitative metrics are used in articles, like response time, cost, energy, etc. The essential metrics for load balancing in fog computing are stated below:

i. ***Response Time:*** This issue is described by the interval starting from the acceptance of a request (or task) to the response to a request for a server in fog environment (Shafiq et al., 2022).
ii. ***Cost:*** The payment of money to ask for an action that is required to do (Kashani et al., 2022).
iii. ***Energy Consumption:*** It refers to the energy consumption amount in a fog network. Energy consumption can be decreased by an effective load balancing mechanism (Kashani et al., 2022).
iv. ***Scalability:*** It shows how the system is capable of accomplishing a load balancing mechanism with a couple of hosts or machines (Kashani et al., 2022).
v. ***Security:*** It is the quality side of service that procures non-repudiation and confidentiality via authentication involving parties and message encryption (Kashani et al., 2022).
vi. ***Flexibility:*** In fog networks, nodes may join or leave the fog environment dynamically due to mobility, intermittent connectivity, power constraints, or other factors. A flexible load balancing mechanism must account for this by detecting newly joined nodes (or volunteer nodes) and removed/revoked nodes, updating the system's view of available resources, and adjusting load distribution accordingly (e.g., by accepting or rejecting volunteer nodes based on mobility, energy, or availability). For example, in LIMO: Load-balanced Offloading with MAPE and PSO in Mobile Fog Networks (Seraj et al., 2024), the authors account for node mobility and task migration to adapt to changes in node presence. Similarly, the mobility-aware hierarchical fog framework in IIoT allows end devices to volunteer as fog nodes, and includes mobility and energy criteria in accepting or rejecting such nodes.
vii. ***Resource Utilization:*** It represents the maximum utilization of the resources available in a cloud system (Mall et al., 2024).
viii. ***Deadline:*** The latest time when a service request in the fog system can be completed (Sirjani et al., 2025).
ix. ***Processing Time:*** The duration in which a service request in the fog system is executed entirely (Misirli & Casalicchio, 2024).
x. ***Reliability:*** The ability of a fog network to perform its required requests in a defined time and a specified condition (Rateb et al., 2025).
xi. ***Throughput:*** We can refer to the maximum requested service rate might be processed in the fog system as throughput (Rathi et al., 2022).
xii. ***Availability***: A rise in resource application or service requests can maintain the system performance that shows the capability of a computing system (Bachiega et al., 2023).

Significance of Load Balancing in Fog Network

Equal workload allocation in the fog environment has become more important as a result of the growth of IoT in the digital world and a progressive increase in real-time applications. High resource utilization and increased user satisfaction are both made possible by load balancing. As a result, the system's overall performance and resource utility will both increase. It makes sure that no resource is either overburdened or underused (Sarma et al., 2023; J. Singh et al., 2022). Users' demands can be optimally balanced and the system's overall operational cost can be minimized by distributing the burden among all processors equally. Users continuously travel through the architecture based on cloud fog.

Load Balancing with Dynamic Resource Allocation in a Fog Environment

Fog computing shifts the processing of IoT applications from centralized cloud platforms to the network's edge. In a fog environment, routers serve as potential physical servers, providing resources for fog services at the network's edge (Angel et al., 2022). These routers can enhance computational and storage capabilities, allowing them to function as efficient computing nodes.

This approach is particularly beneficial for real-time IoT applications, which often have unique performance requirements in the era of big data. Such applications prioritize edge computing nodes for hosting to meet their stringent demands. In a fog computing environment, users can access and utilize computing, storage, and network resources much like they would in a cloud setting. Additionally, virtualization technology plays a crucial role in dynamically provisioning resources on demand, ensuring flexibility and scalability (Yakubu & Murali, 2023).

The physical resources deployed in the remote cloud data center and the fog computing nodes could be used to run IoT applications. The resource schedulers and managers should select the appropriate computing nodes to host the fog services combined in the IoT applications through the design of resource allocation strategies. The resource allocation for the IoT applications should take into account both the centralized and the geo-distributed computing nodes. The main technologies for managing data centers are resource allocation and resource scheduling, which provide significant contributions to reducing carbon emissions, increasing resource utilization, and achieving load balancing for data centers (Mijuskovic et al., 2021)

Open Issues and Research Challenges

A distributed computing environment is provided by fog computing. The cloud's services are extended to the network's edge by fog computing. Fog computing uses a variety of computer devices termed fog nodes, such as routers, switches, and gateways, which are dispersed throughout a wide geographic region. Fog nodes are linked together by networking technologies including WiFi, 3G, and 4G VOLTE. At the network's edges, fog computing offers storage services to handle user demands. The fog server has to lower maintenance costs while providing features like data management.

Thus the open issues, future trends, and challenges of load balancing in fog computing include:

i. ***Energy Consumption and Green Fog:*** The difficulties of greenhouse gas and carbon emissions were not taken into account by researchers for the majority of the strategies examined. Green fog computing and energy usage can be significant in the context of load balancing and fog computing (Dahiya et al., 2021); nevertheless, only 17% of the examined articles made an energy-related point, which could have increased the acceptance and effectiveness of existing load balancing strategies. Therefore, load balancing strategies that take into account energy utilization, greenhouse gas emissions, and carbon emissions in a foggy environment are emerging and may provide a solution to open problems.

ii. ***Multi-Objective Optimization*:** It is clear that there is no specific mechanism for defining the majority of QoS criteria to choose load balancing in a fog network. Some of the systems, for instance, only take into account factors like energy use, cost, or reaction time while ignoring others like scalability, dependability, and security. In order to take some QoS criteria into consideration, multi-objective optimization in load balancing decision making needs to be increased, and determining how to trade off different characteristics may be a significant unresolved topic.

iii. ***Optimal Solutions:*** Most fog-based load balancing techniques described in literature, including scheduling, resource allocation, etc., fall into the NP-hard and NP-complete complexity categories. Other optimization techniques like bacterial colony optimization, memetic, Artificial Immune System, Imperialist Competitive Algorithm, grey wolf Optimizer, simulated annealing, firefly algorithm, lion optimizer algorithm, and glowworm swarm are good directions for future works. Some meta-heuristic and heuristic algorithms have been used to solve them (Kashani et al., 2022).

iv. ***Implementation Challenges:*** In relation to the fact that fog computing is being studied, the majority of researchers do not yet have access to the real testbed, and it was discovered that 85% of the publications employed simulator-based tools for their evaluations. The application of the stated methods in the real testbed is quite difficult since the outcomes of scenarios like scheduling in the real environment can differ from those in the simulated environment.

v. ***Context-Aware Computing:*** Context-aware computing helps fog computing's load balancing by providing fresh data that may be used for cutting-edge applications and to gather information from the smart objects. Additionally, context-aware computing, which continues to be a suspended case legally connecting with the load balancing in fog computing smart objects, recovers the first application of ontologies as sources of learning (Kashani et al., 2022). Future trends may include the intriguing application of context-aware computing to the development of fog network load balancing strategies.

vi. ***Scalability:*** Some fog computing methods need to be able to operate on massive sizes. Some nodes, devices, and associated processes are not guaranteed by the small-scale validation of these methodologies. Despite its significance, only 8% of the literature has specifically addressed the scalability factor, and the papers that have dealt with it have been defined in small-scale settings. This presents an open research problem.

vii. ***Social Networks Analytics:*** The growth of social networks has produced a solid foundation for their use in fog networks. Big data is produced by the social media content that is shared. Social big data analytics can be used to forecast the distribution of fog resources. In fog networks, it can also anticipate resource and service needs. Massive networks and data have been produced by social networks like Facebook, Twitter, and Instagram (Kashani et al., 2022). Therefore, a promising area for future research might be the use of social network analytics to fog computing's load balancing methods.

viii. ***Interoperability:*** Interoperability can be seen as a crucial success factor in the load balancing in IoT/fog domains due to the variety and dispersal of fog nodes and sources. As consumers want to use many service providers, they frequently compare their favorites and other crucial aspects like price, functionality, etc. Consumers can use interoperability to switch between IoT/fog-based products or apply a combination of services and products to create smart environments in their own unique ways while using load balancing. Therefore, considering interoperability as a crucial component in combining the load balancing in IoT/fog-based services, it will be a fascinating feature for future studies (Kashani et al., 2022).

CONCLUSION

This study reviewed 40 peer-reviewed articles published between 2020 and 2025 that focused on load-balancing techniques in fog computing environments. The findings indicate that load balancing remains a critical challenge in ensuring low latency, efficient resource utilization, and high Quality of Service (QoS) for Internet of Things (IoT) applications. Most of the reviewed works emphasize dynamic and adaptive load-balancing mechanisms, which make decisions based on real-time network status, node workload, and mobility conditions, rather than static or pre-defined allocations.

Across the reviewed literature, metaheuristic algorithms including Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), and Genetic Algorithms (GA) were the most frequently adopted techniques for achieving balanced workload distribution, due to their ability to converge toward near-optimal solutions quickly. Similarly, machine learning-based models, especially reinforcement learning and deep neural network approaches, are increasingly being employed to predict task loads and optimize offloading decisions dynamically. However, these models require significant computational resources, which may limit their deployment in lightweight fog nodes.

A notable trend identified among recent studies is the growing interest in energy-aware and mobility-aware load-balancing frameworks. Many studies recognize that balancing energy consumption against response time remains a major trade-off in fog systems. Hybrid approaches that integrate heuristic optimization with predictive learning have shown potential to improve both energy efficiency and latency performance. Furthermore, studies highlight the need for fault-tolerant and flexible architectures capable of adapting to node churn where fog nodes frequently join and leave the network. Future research should prioritize the development of hybrid intelligent load-balancing models that combine optimization algorithms with learning-based prediction to enhance adaptability and reliability. More experimental validation using real-world IoT datasets and simulators such as iFogSim, EdgeCloudSim, and CloudAnalyst is recommended to assess scalability and performance under varying workloads. Additionally, upcoming studies should integrate security, mobility management, and energy efficiency considerations into unified frameworks to enable sustainable, resilient, and high-performance fog computing environments.

REFERENCES

Abdussami, A. A., & Farooqui, F. (2020). A Systematic Literature Review on Fog Computing Cite this paper. International Journal of Advanced Science and Technology, 29(7), 12755–12769

Aldossary, D., Aldahasi, E., Balharith, T., & Helmy, T. (2025). A Systematic Literature Review on Load-Balancing Techniques in Fog Computing: Architectures, Strategies, and Emerging Trends. In Computers (Vol. 14, Issue 6). Multidisciplinary Digital Publishing Institute (MDPI). https://doi.org/10.3390/computers14060217

Alwakeel, A. M. (2021). An overview of fog computing and edge computing security and privacy issues. Sensors, 21(24). https://doi.org/10.3390/s21248226

Angel, N. A., Ravindran, D., Vincent, P. M. D. R., Srinivasan, K., & Hu, Y. C. (2022). Recent advances in evolving computing paradigms: Cloud, edge, and fog technologies. In Sensors (Vol. 22, Issue 1). https://doi.org/10.3390/s22010196

Atieh, A. T. (2021). The Next Generation Cloud technologies: A Review On Distributed Cloud, Fog And Edge Computing and Their Opportunities and Challenges. In ResearchBerg Review of Science and Technology (Vol. 1, Issue 1)

Bachiega, J., Costa, B., Carvalho, L. R., Rosa, M. J. F., & Araujo, A. (2023). Computational Resource Allocation in Fog Computing: A Comprehensive Survey. ACM Computing Surveys, 55(14 S). https://doi.org/10.1145/3586181

Bajaj, K., Sharma, B., & Singh, R. (2022). Implementation analysis of IoT-based offloading frameworks on cloud/edge computing for sensor generated big data. Complex and Intelligent Systems, 8(5). https://doi.org/10.1007/s40747-021-00434-6

Dahiya, N., Dalal, S., & Jaglan, V. (2021). Efficient Green Solution for a Balanced Energy Consumption and Delay in the IoT-Fog-Cloud Computing. In Green Internet of Things for Smart Cities. https://doi.org/10.1201/9781003032397-7

Das, R., & Inuwa, M. M. (2023). A review on fog computing: Issues, characteristics, challenges, and potential applications. In Telematics and Informatics Reports (Vol. 10). https://doi.org/10.1016/j.teler.2023.100049

Ebrahim, M., & Hafid, A. (2023). Resilience and load balancing in Fog networks: A Multi-Criteria Decision Analysis approach. Microprocessors and Microsystems, 101. https://doi.org/10.1016/j.micpro.2023.104893

Gu, Y., Chang, Z., Pan, M., Song, L., & Han, Z. (2018). Joint Radio and Computational Resource Allocation in IoT Fog Computing. IEEE Transactions on Vehicular Technology, 67(8), 7475–7484. https://doi.org/10.1109/TVT.2018.2820838

Gupta, S., & Singh, N. (2022). Fog-GMFA-DRL: Enhanced deep reinforcement learning with hybrid grey wolf and modified moth flame optimization to enhance the load balancing in the fog-IoT environment. Advances in Engineering Software, 174. https://doi.org/10.1016/j.advengsoft.2022.103295

Hameed, A. R., ul Islam, S., Ahmad, I., & Munir, K. (2021). Energy- and performance-aware load-balancing in vehicular fog computing. Sustainable Computing: Informatics and Systems, 30. https://doi.org/10.1016/j.suscom.2020.100454

Kadhim, A. S., & Manaa, M. E. (2022). Hybrid load-balancing algorithm for distributed fog computing in internet of things environment. Bulletin of Electrical Engineering and Informatics, 11(6), 3462–3470. https://doi.org/10.11591/eei.v11i6.4127

Kadry, S., Rauf, H. T., Khalid, A., Khattak, H. A., Abbas, A., & Awaisi, K. S. (2022). A dynamic load balancing mechanism for fog computing environment. International Journal of Web and Grid Services, 18(3). https://doi.org/10.1504/ijwgs.2022.10045721

Kanbar, A. B., & Faraj, K. (2022). Region aware dynamic task scheduling and resource virtualization for load balancing in IoT–

fog multi-cloud environment. In Future Generation Computer Systems (Vol. 137). https://doi.org/10.1016/j.future.2022.06.005

Kanellopoulos, D., & Sharma, V. K. (2022). Dynamic Load Balancing Techniques in the IoT: A Review. In Symmetry (Vol. 14, Issue 12). https://doi.org/10.3390/sym14122554

Kashani, M. H., Ahmadzadeh, A., & Mahdipour, E. (2022). Load balancing mechanisms in fog computing: A systematic review. IEEE Transactions on Services Computing. https://doi.org/10.1109/TSC.2022.3174475

Kashani, M. H., & Mahdipour, E. (2023). Load Balancing Algorithms in Fog Computing. IEEE Transactions on Services Computing, 16(2). https://doi.org/10.1109/TSC.2022.3174475

Kaur, H., Harnal, S., Baggan, V., & Tiwari, R. (2024). Resource Management Solutions for IoT Devices in Fog Computing Architecture. In S. Harnal, R. Tiwari, L. Garg, & A. Mathur (Eds.), Cloud and Fog Optimization-based Solutions for Sustainable Developments. CRC Press. https://doi.org/10.1201/9781003494430

Kaur, M., & Aron, R. (2021). A systematic study of load balancing approaches in the fog computing environment. Journal of Supercomputing, 77(8). https://doi.org/10.1007/s11227-020-03600-8

Kavitha, M. S., Sadashiv, N., & Dilip Kumar, S. M. (2023). Load Balancing in Fog Computing: A Detailed Survey. International Journal of Computing and Digital Systems, 13(1). https://doi.org/10.12785/ijcds/130158

Keshari, N., Singh, D., & Maurya, A. K. (2022). A survey on Vehicular Fog Computing: Current state-of-the-art and future directions. Vehicular Communications, 38. https://doi.org/10.1016/j.vehcom.2022.100512

Mall, L., Vashisht, P., & Narang, A. (2024). Quality of Service Based on Scheduling in Cloud/Fog Environment. International Journal of Innovative Research in Engineering and Management, 11(3), 65–70. https://doi.org/10.55524/ijirem.2024.11.3.10

Mijuskovic, A., Chiumento, A., Bemthuis, R., Aldea, A., & Havinga, P. (2021). Resource Management Techniques for Cloud / Fog and Edge Computing : An Evaluation Framework and Classification. Sensors, 1–22. https://doi.org/10.3390/s21051832

Mishra, K., & Majhi, S. K. (2020). A state-of-art on cloud load balancing algorithms. International Journal of Computing and Digital Systems, 9(2). https://doi.org/10.12785/IJCDS/090206

Misirli, J., & Casalicchio, E. (2024). An Analysis of Methods and Metrics for Task Scheduling in Fog Computing. In Future Internet (Vol. 16, Issue 1). https://doi.org/10.3390/fi16010016

Mondragón-Ruiz, G., Tenorio-Trigoso, A., Castillo-Cara, M., Caminero, B., & Carrión, C. (2021). An experimental study of fog and cloud computing in CEP-based Real-Time IoT applications. Journal of Cloud Computing, 10(1). https://doi.org/10.1186/s13677-021-00245-7

Pereira, E. P., Padoin, E. L., Medina, R. D., & Mehaut, J. F. (2020). Increasing the efficiency of Fog Nodes through of Priority-based Load Balancing. Proceedings - IEEE Symposium on Computers and Communications, 2020-July. https://doi.org/10.1109/ISCC50000.2020.9219576

Rateb, R., Hadi, A. A., Tamanampudi, V. M., Abualigah, L., Ezugwu, A. E., Alzahrani, A. I., Alblehai, F., & Jia, H. (2025). An optimal workflow scheduling in IoT-fog-cloud system for minimizing time and energy. Scientific Reports, 15(1). https://doi.org/10.1038/s41598-025-86814-1

Rathi, S., Nagpal, R., Mehrotra, D., & Srivastava, G. (2022). A metric focused performance assessment of fog computing environments: A critical review. Computers and Electrical Engineering, 103. https://doi.org/10.1016/j.compeleceng.2022.108350

Sarma, B., Kumar, R., & Tuithung, T. (2023). A Dynamic Load Balancing Architecture for Fog Computing using Tree Base Resource Arrangement and Flexible Task Prioritization A Dynamic Load Balancing Architecture for Fog Computing using Tree Base Journal of Information Technology Management, 15.

Seraj, Y., Fadaei, S., Safaei, B., Javadi, A., Monazzah, A. M. H., & Hemmatyar, A. M. A. (2024). LIMO: Load-balanced Offloading with MAPE and Particle Swarm Optimization in Mobile Fog Networks, 2024 5th CPSSI International Symposium on Cyber-Physical Systems (Applications and Theory) (CPSAT), pp. 1-8, https://doi.org/10.1109/CPSAT64082.2024.10745455

Shafiq, D. A., Jhanjhi, N. Z., & Abdullah, A. (2022). Load balancing techniques in cloud computing environment: A review. In Journal of King Saud University - Computer and Information Sciences (Vol. 34, Issue 7). https://doi.org/10.1016/j.jksuci.2021.02.007

Singh, J., Singh, P., Amhoud, E. M., & Hedabou, M. (2022). Energy-Efficient and Secure Load Balancing Technique for SDN-Enabled Fog Computing. Sustainability (Switzerland), 14(19). https://doi.org/10.3390/su141912951

Singh, P., Kaur, R., Rashid, J., Juneja, S., Dhiman, G., Kim, J., & Ouaissa, M. (2022). A Fog-Cluster Based Load-Balancing Technique. Sustainability (Switzerland), 14(13). https://doi.org/10.3390/su14137961

Sirjani, M. S., Ahmad, M., & Sobati-Moghadam, S. (2025). Optimizing Task Scheduling in Fog Computing with Deadline Awareness. http://arxiv.org/abs/2509.07378

Vijarania, M., Gupta, S., Agrawal, A., Adigun, M. O., Ajagbe, S. A., & Awotunde, J. B. (2023). Energy Efficient Load-Balancing Mechanism in Integrated IoT–Fog–Cloud Environment. Electronics (Switzerland), 12(11). https://doi.org/10.3390/electronics12112543

Yakubu, I. Z., & Murali, M. (2023). An efficient meta-heuristic resource allocation with load balancing in IoT-Fog-cloud computing environment. Journal of Ambient Intelligence and Humanized Computing, 14(3). https://doi.org/10.1007/s12652-023-04544-6

Yan, J., Wu, D., Zhang, C., Wang, H., & Wang, R. (2018). Socially aware D2D cooperative communications for enhancing Internet of Things application. Eurasip Journal on Wireless Communications and Networking, 2018(1). https://doi.org/10.1186/s13638-018-1127-0