NextGen Scientific Publication
American International Journal of Computer Science and Technology

Volume 8 Issue 3, Pg. No. 61-72, AIJCST-V8I3P106, 2026
https://doi.org/10.63282/3117-5481/AIJCST-V8I3P106

Original Article

Cloud and Edge-Based Distributed Computing: A Survey of Architectures and Resource Management Approaches

***Anath Bandhu Chatterjee**
Staff Software Engineer, PayPal Inc.

Abstract:

The swift development of data-intensive applications and Internet of Things (IoT) systems has increased the rate of change in the distributed computing paradigms, especially cloud and edge computing. Although cloud computing provides scalable and on-demand services with centralized infrastructures, it has limitations in terms of high latency, bandwidth, and supporting real-time applications. To address these challenges, edge and fog computing have become complementary paradigms that can allow processing data nearer to the source, thus, decreasing the latency and enhancing responsiveness. This paper provides a wide overview of the architectures and resource management strategies in cloud and edge-based distributed computing systems. It discusses the centralized cloud architecture, edge, and fog computing models, the cloud-edge continuum that incorporates these paradigms into a single framework. Moreover, it discusses important resource management methods, including resource allocation and provisioning, task scheduling and load balancing, and QoS- and SLA-aware optimization. Data placement, communication protocols, network optimization and real-time constraints are also discussed in the paper as the data and network management strategies. Finally, it identifies the current challenges, research gaps and future directions for developing efficient, scalable and intelligent distributed computing systems.

Keywords:

Cloud Computing, Edge Computing, Distributed Systems, Resource Management, Task Scheduling, Qos and SLA Optimization, Data and Network Management.

Article History:

Received: 24.03.2026

Revised: 27.04.2026

Accepted: 04.05.2026

Published: 12.05.2026

1. Introduction

The distributed computing and data processing models have become essential in the context of processing the amount of data that is being generated in the modern digital ecosystem. The conventional centralized systems cannot cope with the amount, speed, and types of data generated by businesses, Internet of Things (IoT) devices, and online sites anymore[1]. Distributed computing allows the distribution of computing tasks across multiple nodes, enhancing scalability, performance and fault tolerance. Simultaneously, enhanced data processing models, such as batch and stream processing, enable organizations to process data effectively both offline and online. These developments have formed the basis of modern Computing paradigms, such as cloud and edge computing, which further facilitate dispersed data processing.

Cloud computing enables organizations to deliver computing services through internet-based delivery, which includes servers, storage, databases, networking and software[2][3]. The system provides main benefits through its capacity to scale operations, its ability to adapt and its economic performance, which allows several virtual computers to run on a single physical server thanks to virtualization technology. The essential characteristics of cloud computing are defined by the National Institute of Standards and Technology (NIST), which include self-service access on demand, network access across multiple areas, resource sharing and rapid system expansion and

service usage measurement[4][5][6]. The advantages of cloud-centric architectures are limited by challenges with latency, bandwidth usage and real-time data processing, which have led to the emergence of edge computing as a solution.

IoT systems face various challenges that traditional cloud computing resources solve, yet these systems depend on centralized cloud infrastructures, which create problems through their increased latency and excessive bandwidth requirements, especially for applications requiring immediate processing[7]. Edge Computing has resolved these challenges, which enables organizations to process and store their data at locations nearest to their data origin points[8]. Edge computing improves operational efficiency through its network edge computation model which decreases latency and bandwidth needs while keeping sensitive information safe through local data processing. The system facilitates effective allocation of responsibilities since it can distribute processing functions among various components of the system[9]. These features of edge computing make it an appropriate tool for real-time applications, including industrial automation systems, smart city operations and emergency response operations that require real-time data analysis and timely decision-making.

Resource-efficient use is also an important factor in cloud and edge computing setups, especially because of the dynamic and heterogeneous nature of application demands[10]. Resource allocation is process of allocating computing resources (processing power, memory, storage, network bandwidth) to tasks in the system. Strategies of proper allocation enhance performance, lower the cost of operation and delivery of services[11]. There are broadly two resource allocation methods namely: the dynamic and the static methods. The resources allocated by Static allocation are preset and determined by a set of requirements, which has the advantage of being simple but offers limited flexibility to workload changes. Dynamic allocation, on the other hand, optimizes resources based on system conditions dynamically and thus offers a higher degree of flexibility and better utilization. The decision between the two methods is dictated by the needs of the application as well as the limitations of the system.

1.1. Structure of the Paper

The paper is organized as follows: The paper is structured in the following way: Section II introduces cloud and edge-based architectures, such as centralized, edge, fog models and hybrid models. In section III, the resource management is mentioned and is addressed in terms of allocation, scheduling, load balancing, and optimization of the QoS/SLA. Section IV discusses data and network management, such as data placement, communication, and latency. Section V discusses the recent literature and identifies research gaps, whereas Section VI presents the paper's conclusion and future directions.

2. Architectures of Cloud and Edge-Based Distributed Computing

The current computing systems are based on cloud and edge-based distributed computing architectures that allow scalable, efficient, and flexible use of resources[12]. Conventional centralized cloud systems offer efficient processing and storage services, by bringing processing much closer to data origins, edge and fog systems lower latency and bandwidth consumption[13]. To overcome the limitations of individual models, cloud-edge continuum leverages these paradigms to create a single model that dynamically distributes workload across multiple layers. This architectural evolution supports emerging applications requiring real-time processing, high scalability, and efficient data management in distributed environments.

2.1. Cloud Computing Architecture

The concepts and tools of virtualization and distributed computing form the foundation of cloud computing[14]. Mell and Grance define cloud computing as a model that permits straightforward, on-demand network access to a pool of configurable computing resources, including servers, networks, storage, apps, and services, which can be quickly provisioned and released with little management work or interaction with service providers[15][16][17]. Large data centers with thousands of servers are used to construct centralized clouds. These data centers are the hub for all application processing and resources. They are run and owned by a single company. Figure 1 illustrates a centralized cloud architecture. The existing commercial systems are made up of centralized clouds. Amazon EC2, Google App Engine, Windows Azure, etc.

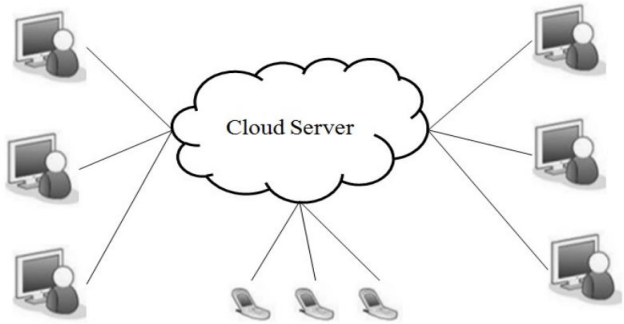

Figure 1. Overview of Cloud Architecture

Cloud computing design combines EDA (Event Driven design) with SOA (Service Oriented Architecture)[18]. Front-end and back-end systems are the two main parts of the environment.

- Front-End Components: The client devices, web browser interfaces, and apps that communicate with the cloud platform are examples of user-side components. These APIs provide users access to cloud resources.
- Back-End Components: The service provider uses the cloud itself. This covers the cloud provider-managed infrastructure layer, which consists of servers, storage systems, databases, and application software. The back end is in charge of resource management, security, and service delivery.

2.1.1. Virtualization and Hypervisors

Hypervisor-driven virtualization is another technology that lies at the heart of cloud computing[19]. It lets several virtual machines (VMs) run on a single physical server, but they are not connected to each other. Hypervisors (e.g., VMware ESXi, Microsoft Hyper-V, KVM) are a software layer that controls these virtual machines, ensuring that they are allocated and used efficiently.

2.1.2. Resource Orchestration and Automation

The process of resource provisioning, scaling and management in either private or public clouds is done by cloud orchestrators [20] like Kubernetes, OpenStack and Terraform[21]. They ensure that service is available, control containers and optimize workloads in real time. The orchestration simplifies complex cloud operations by integrating automation into workflows. Cloud automation is the use of software to manage and streamline such operations, reduce human labor, and increase efficiency, responsiveness and reliability in cloud environment management.

2.1.3. Edge and Fog Computing Architectures

In contrast to centralized data centers, edge computing aims to process data closer to point of production. The technique reduces latency and bandwidth consumption and permits real-time application execution.

By putting processing, storage, and networking resources closer to data source, fog computing expands on the ideas of edge computing[22]. Fog computing typically involves intermediate layers of computing infrastructure, such as edge gateways or fog nodes, situated between edge devices and centralized data centers.

2.1.4. Architecture of Edge Computing

Figure 2: The Enterprise layer, Edge cloud or Edge network, and the Edge layer share the workload in multi-layered distributed systems.

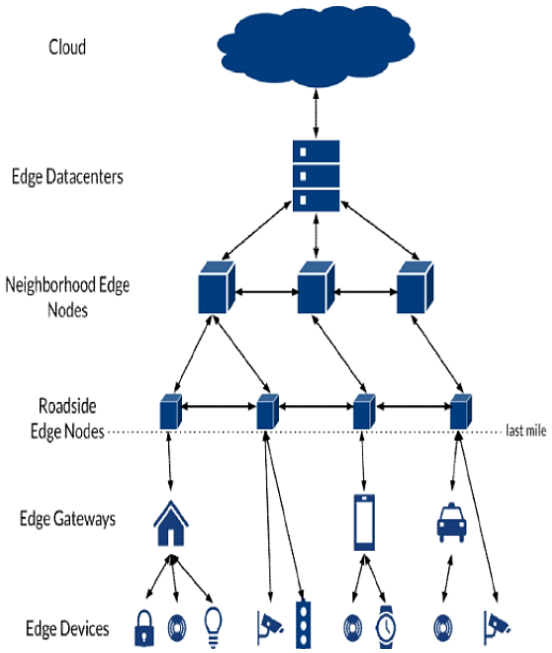

Figure 2. Edge Computing Architecture

Edge computing encompasses various architectural models that dictate how computing resources are distributed and organized within a network.

2.2. Fog Computing Architecture

A dispersed computer infrastructure known as fog computing places applications, computation, storage, and statistics between the cloud and the information supply [23]. It serves as an intermediary between stop devices (clients) and conventional cloud servers, forming a layer of a designated community. The components of this tiered architecture are:

- Clients: These end-users, which generate statistics and include IoT devices like sensors, actuaries, cellphones, and more.
- Edge Devices: They make up community's immediate layer, which is where information is first digested. This comprises switches, routers, and other devices that can provide community connectivity and some record processing.
- Fog Nodes: These are geographically distributed, decentralized nodes that offer compute resources nearer the network's threshold. They carry out a great deal of administration, size, configuration, communication, and storage work.
- Cloud Services: The cloud is a representation of the centralized statistical centers that give birth to long-term storage and significant information processing.

Understanding fog computing's layered architecture is crucial to comprehending how well it manages the massive volume of data produced by IoT devices. The decentralized shape of the cloud layer to the buyer layer in Figure 3 below illustrates the hierarchical structure of fog computing in an infographic. By understanding the allocation of computer resources closer to the information source, this graphical depiction minimizes latency and maximizes fact processing.

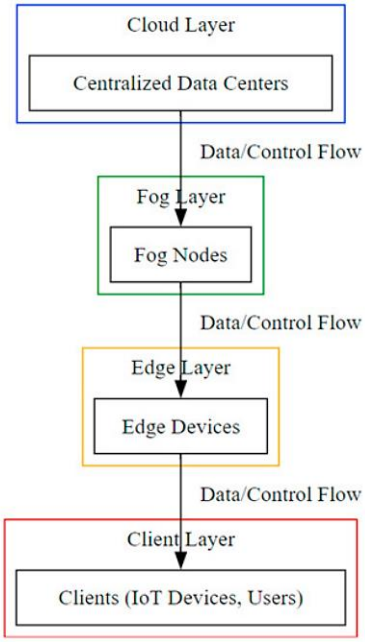

Figure 3. Fog Computing Architecture

2.3. Edge-Cloud Continuum and Hybrid Models

The edge cloud continuum is a single, hierarchical model where the computation, storage, and control are distributed between devices, edge servers, fog nodes, and cloud infrastructure[24]. Instead of viewing edge and cloud as discrete platforms, the continuum unites them into a unified platform that dynamically coordinates the allocation of resources, data flow, and placement of workloads[25]. This continuum of heterogeneity is characterized by heterogeneity in hardware capabilities, geographical distribution of nodes and adaptive orchestration such that, depending on the latency requirements, jobs are carried out at the best location, energy and performance.

The continuum enables applications to consider the edge nodes to implement time-sensitive functions and the cloud to implement large-scale training, synchronization and persistent storage. This hierarchical synergy is already underlying new systems, such as autonomous transportation systems and industrial digital twins.

The computation of hybrid cloud-edge continuum model is spread across multiple layers, such as the end devices, edge nodes and cloud infrastructure[26][27]. The layers possess distinctive characteristics in terms of computational power, energy constraints and communication delay. In this way, one can wisely allocate workloads to such heterogeneous resources so that the performance of a system becomes efficient.

3. Resource Management Approaches

Resource management is arguably among the most critical aspects of cloud and edge-based distributed computing systems because it ensures an efficient use of heterogeneous resources and fulfillment of the requirements of the application. It addresses such key functions as resource allocation and provisioning, task scheduling, and load balancing to ensure that systems deliver maximum performance and scalability. Furthermore, QoS- and SLA-aware optimization schemes play a vital role in reliability of services and delivery of performance guarantees. All these combined strategies allow managing resources dynamically and efficiently in complex cloud-edge environments.

3.1. Workload-Aware Resource Allocation and Optimization

Resource allocation on the cloud-edge dynamically provisions and operates distributed resources (e.g., virtual machines, containers, and edge nodes). These systems need to support diversified types of workloads, such as latency-sensitive (e.g., real-time analytics and autonomous systems), compute-intensive (e.g., AI/ML inference), and data-intensive (e.g., video processing and IoT streams) workloads[28]. Lightweight virtualization, and in particular containers, is a widespread practice at the edge due to their low overhead and fast deployment.

To manage such a heterogeneous workload, resource allocation policies are designed with optimization mechanisms that take into account the needs of applications, the nature of the workload and the limitations of the system. The allocation process typically includes the analysis of the received requests, the selection of suitable resources and the intelligent coordination of the resources between the cloud and edge layers that result in an effective execution.

Key workload-aware optimization strategies include:

- Workload-Aware Scheduling: Allocates tasks based on workload type and priority to minimize latency and maximize performance efficiency.
- Adaptive Resource Scaling: Automatically increases resources when the workload requires it, and this leads to efficient use of resources and reducing the cost of operation.
- AI-Driven Decision Making: Utilizes ML to predict work schedules, optimize resource utilization, and support proactive resource management.
- Energy-Aware Optimization: Controls the resource distribution to minimize the energy consumption without affecting the performance requirement.

Additionally, variability in the workloads and demand trends requires real-time changes in resource allocation policies. The inefficiency of the distribution of workloads can either underutilize or overload the resources and lead to SLA breaches and poor performance. Therefore, intelligent and active optimization policies are essential in ensuring effective, trustworthy and scalable operations of cloud-edge distributed systems.

3.2. Task Scheduling and Load Balancing

A three-layer model with end devices, a fog layer, and a centralized cloud server serves as the foundation for the fog computing architecture. The fog layer consists of multiple fog nodes or servers that provide processing, communication and storage capabilities, often implemented through virtualization technology to enable the creation of many virtual machines (VMs). These fog nodes are placed close to end users in order to be able to communicate easily through wireless networks such as Wi-Fi and Bluetooth, thereby reducing the latency and establishing real-time services.

This architecture and load balancing of the system are essential to effective system performance. After the requests sent by the user arrive at the fog layer, they may be split into smaller tasks and assigned to various virtual machines to be processed. The task of scheduling algorithms is to decide how to allocate the tasks in the available resources in the best way to achieve minimal execution time, memory usage, and resource consumption[29]. Nonetheless, the problem of scheduling tasks in a fog environment is a computationally hard problem, usually NP-hard (because there are no deterministic solutions in the poly time).

Moreover, the workload might not be evenly distributed to fog nodes, as certain servers may become overloaded, and others may be underutilized, especially during the peak demand period. Hence, there is a need to establish proper load balancing system to balance the workloads among the fog nodes and avoid performance degradation and enhance system reliability. Effective scheduling and dynamic load balancing are essential to the effective implementation of fog and edge computing systems.

3.3. QoS and SLA-Aware Optimization

The quality or degree of the availability, performance, and reliability of an application, as well as underlying infrastructure in which it is housed, is known as quality of service (QoS). QoS is inherent to cloud users, who demand service providers to achieve predetermined performance levels, and providers, who need to trade off quality of provided services with operational expenses. Such a balance is a complicated decision-making task because it implies Service Level Agreements (SLAs), it defines QoS targets, as well as penalties in case of SLA breaches. SLAs are legally binding agreements between service providers and consumers, which establish the anticipated performance levels and the related economic consequences. Compliance with SLAs is essential, and their breach directly affects the revenue and the reliability of the services[30]. In order to combat this, monitoring mechanisms are used to monitor the performance of the system and identify possible SLA violations on the fly.

However, it turns out to be increasingly challenging to manage QoS in cloud and edge environments due to the following reasons: heterogeneity of resources, unstable loads, and changes in performance. These challenges have led to the development of QoS and SLA-aware optimization techniques which dynamically adjust resource allocation, task and workload allocation schedules. Such optimization techniques are meant to minimize the cases of SLA breach and improve efficiency of the system to facilitate cost-effective and reliable delivery of services in distributed computing systems.

4. Data and Network Management in Distributed Environments

Data and network management are incomplete without cloud-edge distributed systems, which are needed to process large volumes of data and enable seamless interaction in a heterogeneous environment. This includes information-positioning and information-processing schemes to minimize latency and make maximum use of resources, as well as communication and network-optimization schemes to ensure reliable and efficient data exchange. Furthermore, problems like latency, bandwidth, and real-time constraints significantly affect the system's performance and the applications' responsiveness. These points combined are vital in realizing scalable, efficient and high-performance distributed computing systems.

4.1. Data Placement and Processing Strategies

The positioning and processing techniques of data in cloud and edge-based distributed computing systems are critical to performance, scalability, and efficiency of system[31]. Data placement is the process of choosing where to store data across system layers, such as fog layers, edge nodes, end devices, and centralized cloud data centers. The placement of data in a distributed environment must be efficient to reduce data transfer costs, lower latency and increase overall resource utilization.

The latest cloud-edge platforms support data-intensive applications that generate large volumes of heterogeneous data requiring real-time processing. Conventional cloud-based solutions are based on the centralized storage and processing of data, and they tend to cause more latency and consumption of bandwidth. Edge-enabled strategies overcome these drawbacks by moving latency-sensitive and frequently accessed data to the edge, thereby improving response times and reducing network congestion.

A number of data placement methods have been suggested to improve performance of system:

- Replication-based techniques improve availability and fault tolerance by creating copies of data across several nodes.
- Caching devices are used to store information that is accessed frequently at the edge nodes to avoid transmission of the same data many times to the cloud.
- Locality-aware placement strategies take into account the geographical position of users and data sources to reduce the delays in communication.
- Placement Workflow-based placement methods maximize the positioning of intermediate data within data-intensive applications, minimizing the time and resource usage of the application.

Data processing plans go hand in hand with data location choices and dictate the distribution of computation throughout the system:

- Edge processing is the time-sensitive and latency-sensitive processing that is performed closer to the data source.
- Compute-intensive workloads like large-scale analytics, long-term storage, and model training are done on clouds.
- Hybrid processing models balance the workloads on both edge and cloud layers and take the strengths of each platform to enhance efficiency.

Despite these advancements, data placement and processing in cloud–edge systems remain challenging due to resource heterogeneity, dynamic workloads, limited storage at edge nodes, and varying network conditions. Therefore, intelligent and adaptive strategies are required to continuously optimize data placement and processing decisions. These strategies aim to balance latency, bandwidth consumption, energy efficiency, and system performance, making them essential for next-generation distributed computing systems.

4.2. Communication Protocols and Network Optimization

The efficient data interchange between endpoints, edge nodes, and cloud infrastructure is made possible by communication protocols in cloud-edge distributed systems[32]. These systems primarily rely on two interaction models: request–response and publish–subscribe. Several communication protocols and models are widely used:

- The request–response model is a fundamental communication paradigm in which A server receives a request from a client, processes it, and provides a response. In web-based and centralized systems, this concept is frequently applied.
- HTTP and CoAP are based on the request–response model, where HTTP is widely used for web communication, CoAP is intended for IoT devices with limited resources.
- Asynchronous communication is possible with the publish-subscribe approach, where publishers send messages to a broker, and subscribers receive updates without direct communication.
- MQTT and AMQP follow the publish–subscribe model and are suitable for large-scale distributed systems with multiple users and devices.
- Communication protocols provide different levels of Quality of Service (QoS), MQTT and AMQP support multiple QoS levels for reliable message delivery, while CoAP offers limited QoS through confirmable and non-confirmable messages.

- The DDS protocol provides an extensive set of QoS policies, allowing fine-grained control over reliability, latency, and data delivery.

Network optimization in cloud-edge systems is expected to improve the performance of the system by lowering latency, bandwidth usage, and energy consumption. Methods like edge caching minimize unnecessary data transfer by storing high-access data at edge of network, which is also called caching[33], and offloading of computation between edge and cloud devices to enhance efficiency and reduce waiting times. Also, there is a growing use of AI-based solutions to optimize the use of resources, forecast network conditions, and dynamically manage traffic. These solutions greatly enhance the scalability, responsiveness and performance of distributed computing systems.

4.3. Latency, Bandwidth, and Real-Time Constraints

Latency is one of the key characteristics that sets edge computing solutions apart from clouds. Conventional cloud-based systems require data to be moved from devices along the network path to centralized data centers and back, which, in the event of overloaded or geographically dispersed networks, can result in round-trip delays of tens to hundreds of milliseconds, or even seconds[34]. Time-sensitive applications, such as autonomous vehicles and augmented reality, and industrial control systems are not suited for this delay, where responses must be received in milliseconds or microseconds.

Edge computing helps alleviate this issue by keeping computational power nearer to the data, allowing it to be processed with minimal network access [35]. Empirical experiments have found that the latency is reduced by about 80-95% when typical IoT workloads are performed on the edge. Another important difference is the bandwidth usage. Cloud-based solutions need raw data to be constantly transmitted, which consumes a lot of bandwidth and may congest the network. An example is that one self-driving vehicle is capable of creating 4-5 terabytes of data in a day, rendering complete cloud transmission unfeasible. The major distinctions between cloud and edge computing can be summed up as follows:

- Latency: Edge computing is a much better solution in terms of latency because data is processed close to the source, but cloud computing causes increased delays because of the lengthy network routes.
- Bandwidth Utilization: Edge computing reduces bandwidth consumption by performing data processing locally, whereas cloud computing needs mass data transmission.
- Data Processing Approach: The foundation of edge computing is distributed, localized processing, whereas the foundation of cloud computing is centralized.
- Scalability Impact: Large-scale IoT deployments can also affect cloud models by network bottlenecks, which is mitigated with edge computing through decentralized operations.

The importance of edge computing also applies to modern distributed systems, time constraints underscore it. Real-time video analytics, industrial automation, and smart healthcare applications all require quick decision-making and real-time data processing. These may be integrated with low latency and efficient bandwidth utilization to ensure that edge computing systems can adequately address these demanding requirements.

5. Literature Review

Recent articles point out the advances in edge-cloud computing and edge-resource control, particularly on optimization techniques, QoS-aware plans, and hybrid designs, and show deficiencies in integrated, real-time, and scalable resource control models.

J. Liu et al. (2026) provide a tutorial on the underlying architectures, communication protocols, and network-conscious computing structures, as well as critically examine the convergence of model optimization and distributed intelligence in edge-cloud systems. AI-based resource management strategies that may balance performance, energy consumption, and communication overhead across heterogeneous networks are being thoroughly studied, as are model optimization strategies including compression, adaptability, and neural architecture search. To create some criteria of assessment of these complex distributed systems, the performance measurement and benchmarking techniques are also thoroughly examined[36].

A. A. Vali et al. (2026) provide a summary of the most advanced resource management techniques in edge computing and microservices-based fog with an emphasis on energy-efficient solutions. They systematically review and classify over 136 publications (2020-2024) into five broad subdomains: resource allocation, instance selection, task scheduling and offloading, service placement, and resource provisioning. This is categorized into the optimization methods, goal to be met and strength and weaknesses of each method[37].

N. E. H. Boubaker et al. (2025) provide an up-to-date critical overview of resource management techniques in Edge-Fog-Cloud ecosystems. They adopt a holistic perspective that integrates the methods of combinatorial optimization with the methods of learning and their roles in the context of resource management in different aspects. The survey examines five key aspects that are, scheduling of tasks, placement of resources, allocation, offloading and migration of service. They suggest a systematic taxonomy of more than 70 works (2017-2025) depending on their goals, methods, QoS systems for decision-making, as well as external elements like user movement and network volatility. The role of the synergy between combinatorial and ML-based approaches is given special consideration, and in which cases either of them is more efficient, and in which cases their combination can provide more robust solutions[38].

N. Rasouli, C. Klein, and E. Elmroth (2025) explore in-depth the intricacies of deploying mission-critical applications in an edge environment, dissecting their technical needs. A comprehensive literature review based on a conventional review methodology is used in the investigation. They examine why quality metrics for cornerstone service metrics that are crucial to such critical applications in edge contexts are sought after to deliver services efficiently. Additionally, they found that there were some significant gaps in existing resource management strategies[39].

H. M. Zangana, A. K. Mohammed, and S. R. M. Zeebaree (2024) explore the changing nature of cloud architecture of distributed edge computing, particularly of collaborative and decentralized computing paradigms. This systematic review focuses on synthesizing the latest developments in decentralization methods, blockchain integration with edge computing, federated learning, and cooperative scheduling. With edge computing gaining increasing importance as a means of supporting IoT and other distributed systems, new approaches are required to mitigate the challenges associated with latency, resource consumption, and data protection. The benefits of autonomous serverless systems, cooperative sharing of computing resources, and latency-conscious task management systems are the main conclusions[40].

F. C. Andriulo et al. (2024) aim to systematically examine and compare cloud computing, edge computing, and hybrid architectures with an emphasis on their use in IoT environments. The current approach presents recent developments in computing technology for IoT applications following a thorough search and analysis of peer-reviewed publications, conference proceedings, and industry reports. The highlights indicate that edge computing is best at minimizing latency and improving data privacy due to local processing, while cloud computing is the most flexible and scalable. Among hybrid technologies, fog and mist computing is the most common choice, blending the best aspects of edge and cloud computing systems. In the IoT ecosystem, SE hybrid models are effective for bandwidth utilization and low-latency, privacy-sensitive applications. Hybrid architectures are also found to be especially suitable in cases where there is a need to manage bandwidth efficiently, and low-latency processing is needed[9].

X. Zhang and S. Debroy (2023) provide a general overview of the issues and solutions to them that can be found in MEC resource management. They begin by explaining the key issues in the resource allocation of MEC in the situation where the user applications possess different performance requirements. They explain the peculiarities of issues associated with the dynamism of the environments and provide the cases of the use where MEC is implemented[41].

Table I summarizes the recent research on edge-cloud computing and resource management, highlighting their focus, discoveries, limitations, challenges, and gaps in the research in unified frameworks, real-time optimization, adaptive resource management, and large-scale implementation of distributed systems.

Table 1. Summary of Recent Studies in Edge–Cloud Computing and Resource Management Systems

Authors	Focus	Key Findings	Challenges	Limitations	Future Work
J. Liu et al. (2026)	Model optimization, distributed intelligence, and edge-cloud continuum computing (ECCC)	Identified architectures, communication protocols, and optimization techniques (compression, NAS, adaptation); emphasized AI-driven resource management	Managing heterogeneous networks, balancing energy efficiency, performance, and communication overhead	Limited real-world implementation validation across large-scale deployments	Develop scalable, real-time optimized ECCC frameworks with stronger benchmarking and deployment validation

		and benchmarking standards			
A. A. Vali et al. (2026)	Resource management in fog and edge computing using microservices	Categorized 136 studies into service placement, provisioning, scheduling, allocation, and instance selection; highlighted energy-efficient optimization techniques	Complexity in integrating multiple optimization objectives and dynamic microservice environments	Primarily survey-based; lacks experimental validation and unified framework	Design unified frameworks integrating multiple resource management strategies with real-world validation
N. E. H. Boubaker et al. (2025)	Resource management in Edge-Fog-Cloud environments	Developed taxonomy of 70+ works; highlighted synergy between combinatorial optimization and ML-based approaches for QoS-aware decisions	Handling user mobility, network variability, and multi-objective optimization	Limited practical implementation of hybrid optimization models	Develop hybrid ML + optimization frameworks for adaptive and scalable resource management
N. Rasouli et al. (2025)	Mission-critical applications in edge environments	Identified QoS metrics and gaps in current resource management strategies for critical applications	Ensuring reliability, low latency, and fault tolerance in edge systems	Focus limited to analysis; lacks concrete implementation strategies	Develop robust, fault-tolerant resource management frameworks for mission-critical edge systems
H. M. Zangana et al. (2024)	Decentralized and collaborative edge-cloud architectures	Highlighted benefits of federated learning, blockchain, serverless frameworks, and collaborative scheduling	Addressing latency, security, and efficient distributed coordination	Limited performance evaluation across diverse real-world scenarios	Enhance decentralized frameworks with improved security, scalability, and real-time coordination
F. C. Andriulo et al. (2024)	Comparative study of IoT edge, cloud, and hybrid architectures	Found edge reduces latency and improves privacy, cloud offers scalability, hybrid models balance both	Efficient bandwidth management and integration of hybrid systems	Limited focus on standardized orchestration and management frameworks	Develop intelligent orchestration systems for hybrid architectures in IoT
X. Zhang and S. Debroy (2023)	Mobile Edge Computing (MEC) resource management	Identified challenges in resource allocation for diverse applications; categorized optimization and deep learning approaches	Dynamic environments and varying QoS requirements	Limited exploration of integrated AI-driven real-time solutions	Develop adaptive AI-based resource allocation systems for dynamic MEC environments

6. Conclusion and Future Work

The growing need for real-time and data-intensive applications has made cloud and edge-based distributed computing a paradigm in the current computing environments. In this paper, a detailed overview of architectures and resource management strategies in these

systems was provided. It discussed centralized cloud designs, edge computing and fog computing designs, and cloud-edge continuum, highlighting how these designs enable efficient workload allocation and scalable system designs. The primary methods of resource management comprised load balancing, employment scheduling, and resource allocation, and quality of service/service level agreement-conscious optimization were explored in their impacts on the system performance, efficiency and reliability. Moreover, data and network management policies, such as data location, communication policies and latency-sensitive optimization, were also discussed as crucial factors in the system's responsiveness and scalability. Despite these successes, multiple challenges exist, including resource heterogeneity, adjustment of workload, scalability, and the need to make effective real-time decisions. Further research is necessary to come up with single and multi-purpose resource management systems that can be effective to operate across cloud-edge continuum. This tendency shows integration of AI and ML approaches to offer the opportunity of dynamic decision-making, optimization of the processes in real-time and prediction in resource allocation. In addition, the challenges of achieving scalability, interoperability, security and energy efficiency are some of the critical problems that the development of distributed computing systems to support large-scale applications requires resolving.

References

[1] N. Perera and J. Mathew, "A Review of Distributed Computing and Data Processing Models," vol. 6, 2019.
[2] C. Tayal, "Designing Hybrid ETL Pipelines for Multi-Cloud Integration," *Int. J. Emerg. Trends Comput. Sci. Inf. Technol.*, vol. 4, no. 4, pp. 129–134, 2023, doi: 10.63282/3050-9246.IJETCSIT-V4I4P114.
[3] B. P. Singh and H. Singh, "Using LLMs for Autonomous Cloud Infrastructure Entitlement Management to Prevent Overprivileged Access," *J. Eng. Comput. Sci.*, vol. 5, no. 4, pp. 1–14, April, 2026, doi: https://doi.org/10.5281/zenodo.19488212.
[4] A. S. Omar and F. Mwakondo, "Evolution of Cloud Computing: Trends, Issues, and Future Directions: A Systematic Literature Review," *Int. J. Comput. Sci. Trends Technol.*, vol. 12, no. 3, pp. 102–111, 2024.
[5] A. Parupalli and H. Kali, "An In-Depth Review of Cost Optimization Tactics in Multi-Cloud Frameworks," *Int. J. Adv. Res. Sci. Commun. Technol.*, vol. 3, no. 5, June, pp. 1043–1052, 2023, doi: 10.48175/IJARSCT-11937Q.
[6] Y. Patel, "Self-Adaptive Al-Based Orchestration for Multi-Cloud Interoperability and Performance Optimization," in *SoutheastCon 2026*, Huntsville, AL, USA: IEEE, 2026, pp. 01–08, April. doi: 10.1109/SoutheastCon63549.2026.11476031.
[7] S. Hamdan, M. Ayyash, and S. Almajali, "Edge-Computing Architectures for Internet of Things Applications: A Survey," *Sensors*, vol. 20, no. 22, p. 6441, Nov. 2020, doi: 10.3390/s20226441.
[8] S. Singamsetty, "EdgeNexus: Bridging AI and Data Engineering for Seamless Edge Computing," *Turkish Online J. Qual. Inq.*, vol. 13, no. 1, pp. 2343–2351, 2022.
[9] F. C. Andriulo, M. Fiore, M. Mongiello, E. Traversa, and V. Zizzo, "Edge Computing and Cloud Computing for Internet of Things: A Review," *Informatics*, vol. 11, no. 4, 2024, doi: 10.3390/informatics11040071.
[10] C. Patel, "A Review of Multi-Channel CRM Strategies Using Big Data and Cloud Integration," *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.*, vol. 8, no. 1, January-February, pp. 577–588, 2022, doi: https://doi.org/10.32628/IJSRCSEIT.
[11] M. A. Mary, "Survey on Resource Management Technique in Cloud Computing," *Int. J. Eng. Res. Technol.*, vol. 2, no. 12, pp. 232–235, 2013, doi: 10.21275/v5i4.nov162511.
[12] N. Radhasharan, "Real-Time Edge-To-Cloud Intelligence Architecture For Autonomous Drilling Systems," *J. Int. Cris. RISK Commun. Res.*, vol. 9, no. 1, pp. 90–102, 2026, doi: 10.63278/jicrcr.vi.3577.
[13] A. Warrier and A. K. S, "Hybrid Edge-Cloud AI Gateway with 1D-CNN for Real-Time Anomaly Detection and Temporal Fusion Transformer for Healthcare Data Streams," in *2025 7th International Conference on Innovative Data Communication Technologies and Application (ICIDCA)*, IEEE, Oct. 2025, pp. 204–211. doi: 10.1109/ICIDCA66325.2025.11280437.
[14] V. K. Sharma, "Cloud Computing IoT: 5G Focused IoT with Cloud Solutions," *Int. J. AI, BigData, Comput. Manag. Stud.*, vol. 6, no. 3, pp. 21–25, July, 2025, doi: 10.63282/3050-9416.IJAIBDCMS-V6I3P103.
[15] S. P. S. and S. S., "A Review on Cloud Computing Architectures," *Int. J. Comput. Appl.*, vol. 152, no. 7, pp. 1–4, Oct. 2016, doi: 10.5120/ijca2016911879.
[16] S. R. Sirikonda, "Reducing SRE Toil via Safe Autonomous Remediation in Cloud-Native Systems," *Am. J. Technol.*, vol. 5, no. 3, pp. 30–49, 2026.
[17] R. K. Gadiraju, "Artificial Intelligence for Resource Optimization in Cloud Computing Environments," *J. Electr. Syst.*, vol. 20, no. 6, pp. 3164–3174, March, 2024.
[18] J. A. Mahatme and M. R. Satpute, "A Comprehensive Review on Cloud Computing: Challenges, Architectures, and Future Directions," *IARJSET*, vol. 12, no. 7, Jul. 2025, doi: 10.17148/IARJSET.2025.12736.
[19] A. K. Padhy, T. P. Patel, V. Soni, A. K. Elengovan, G. B. Thokala, and N. Seshagiri, "Cloud-Native Multimodal Semantic Search and Recommendation for Large-Scale Digital Commerce," in *2026 4th Odisha International Conference on Electrical Power Engineering, Communication and Computing Technology (ODICON)*, 2026, pp. 1–6, February. doi: 10.1109/ODICON66687.2026.11470613.
[20] A. Warrier, "iPaaS Solutions for Healthcare Enterprise Integration: Cloud-Native Integration Platforms for Multi-System Orchestration," *Int. J. Lead. Res. Publ.*, vol. 3, no. 1, pp. 1–9, Jan, Jan. 2022, doi: 10.70528/IJLRP.v3.i1.1770.
[21] P. Naayini and S. Kamatala, "Automating Infrastructure Platforms with Cloud, Kubernetes, and Site Reliability Engineering," *Int. J. Comput. Tech.*, vol. 8, no. 6, pp. 1–9, November, 2021.
[22] D. Penglin and M. L. Ali, "A Survey of Emerging Trends in Edge Computing," 2024. doi: 10.13140/RG.2.2.22183.76962.

[23] Z. R. Ahmed, S. Askar, D. H. Hussein, and M. A. Ibrahim, "Fog Computing Challenges and Opportunities in IoT Networks: A Review," *Procedia Comput. Sci.*, vol. 259, pp. 1749–1764, 2025, doi: 10.1016/j.procs.2025.04.130.
[24] M. R. C. Mukkolakkal, "InfraLLM: A Generic Large Language Model Framework for Production-Grade Microservice Auto-Scaling in Cloud Infrastructure," *Int. J. Sci. Res. Mod. Technol.*, vol. 4, no. 11, pp. 113–123, 2025, doi: 10.38124/ijsrmt.v4i11.1023.
[25] R. Sannapureddy, V. M. Nadella, and S. Nelavelli, "Edge–Cloud Continuums for Latency-Sensitive Tasks," *Int. J. AI, BigData, Comput. Manag. Stud.*, vol. 5, pp. 189–201, 2024, doi: 10.63282/3050-9416.IJAIBDCMS-V5I4P121.
[26] A. Al-Dulaimy *et al.*, "The computing continuum: From IoT to the cloud," *Internet of Things*, vol. 27, p. 101272, Oct. 2024, doi: 10.1016/j.iot.2024.101272.
[27] A. Warrier, "Securing and Scaling API Gateways in Hybrid Environments," *J. Artif. Intell. Mach. Learn. Data Sci.*, vol. 3, no. 3, pp. 2914–2920, Sep. 2025, doi: 10.51219/JAIMLD/Arjun-warrier/607.
[28] S. Durga, E. Daniel, J. A. Onesimu, and Y. Sei, "Resource Provisioning Techniques in Multi-Access Edge Computing Environments: Outlook, Expression, and Beyond," *Mob. Inf. Syst.*, vol. 2022, pp. 1–24, Dec. 2022, doi: 10.1155/2022/7283516.
[29] J. Bisht and V. Subrahmanyam, "Survey on Load Balancing and Scheduling Algorithms in Cloud Integrated Fog Environment," 2021, doi: 10.4108/eai.27-2-2020.2303123.
[30] M. Jelassi, C. Ghazel, and L. A. Saidane, "A survey on quality of service in cloud computing," *2017 3rd Int. Conf. Front. Signal Process. ICFSP 2017*, vol. 27, no. 1, pp. 63–67, 2017, doi: 10.1109/ICFSP.2017.8097142.
[31] L. Singh and J. Malhotra, "A Survey on Data Placement Strategies for Cloud based Scientific Workflows," *Int. J. Comput. Appl.*, vol. 141, no. 6, pp. 30–33, May 2016, doi: 10.5120/ijca2016909651.
[32] A. Katal and V. Sethi, "Communication Protocols in Fog Computing: A Survey and Challenges," in *Fog Computing*, Boca Raton: Chapman and Hall/CRC, 2022, pp. 153–170. doi: 10.1201/9781003188230-11.
[33] M. Pooyandeh and I. Sohn, "Edge Network Optimization Based on AI Techniques: A Survey," *Electronics*, vol. 10, no. 22, p. 2830, Nov. 2021, doi: 10.3390/electronics10222830.
[34] V. K. Sharma and K. S. Abhilash, "Latency-Aware Edge-Cloud Architecture for 5G IoT Integration," in *2025 6th International Conference on Electronics and Sustainable Communication Systems (ICESC)*, IEEE, Sep. 2025, pp. 1398–1405. doi: 10.1109/ICESC65114.2025.11212232.
[35] S. Min and C. Wei, "Comparative Analysis of Filter-based Feature Selection Methods for High-Dimensional Data in Classification Tasks," *J. Adv. Comput. Syst.*, vol. 3, no. 8, pp. 25–38, Aug. 2023, doi: 10.69987/JACS.2025.50103.
[36] J. Liu *et al.*, "Edge-Cloud Collaborative Computing on Distributed Intelligence and Model Optimization: A Survey," *IEEE Commun. Surv. Tutorials*, vol. 28, pp. 5049–5080, 2026, doi: 10.1109/COMST.2026.3669216.
[37] A. A. Vali, S. Azizi, M. Shojafar, and R. Buyya, "Energy-Efficient Resource Management in Microservices-Based Fog and Edge Computing: State-of-the-Art and Future Directions," *ACM Comput. Surv.*, Apr. 2026, doi: 10.1145/3797911.
[38] N. E. H. Boubaker, K. Zarour, N. Guermouche, and D. Benmerzoug, "A Comprehensive Survey on Resource Management for IoT Applications in Edge-Fog-Cloud Environments," *IEEE Access*, vol. 13, pp. 111892–111925, 2025, doi: 10.1109/ACCESS.2025.3583584.
[39] N. Rasouli, C. Klein, and E. Elmroth, "Resource Management for Mission-Critical Applications in Edge Computing: Systematic Review on Recent Research and Open Issues," *ACM Comput. Surv.*, vol. 58, no. 3, pp. 1–37, Feb. 2026, doi: 10.1145/3762181.
[40] H. M. Zangana, A. K. Mohammed, and S. R. M. Zeebaree, "Systematic Review of Decentralized and Collaborative Computing Models in Cloud Architectures for Distributed Edge Computing," *SISTEMASI*, vol. 13, no. 4, p. 1501, Jul. 2024, doi: 10.32520/stmsi.v13i4.4169.
[41] X. Zhang and S. Debroy, "Resource Management in Mobile Edge Computing: A Comprehensive Survey," *ACM Comput. Surv.*, vol. 55, no. 13s, pp. 1–37, Dec. 2023, doi: 10.1145/3589639.