



ARTICLE

Empowering Edge Computing: Public Edge as a Service for Performance and Cost Optimization

Ateeqa Jalal^{1,*}, Umar Farooq^{1,4,5}, Ihsan Rabbi^{1,4}, Afzal Badshah², Aurangzeb Khan^{1,4},
Muhammad Mansoor Alam^{3,4} and Mazliham Mohd Su'ud^{4,*}

¹Institute of Computer Science & IT, University of Science and Technology, Bannu, 28200, Pakistan

²Department of Software Engineering, University of Sargodha, Sargodha, 40162, Pakistan

³Faculty of Computing, Riphah International University, Islamabad, 44000, Pakistan

⁴Faculty of Computing, Multimedia University, Cyberjaya, 63100, Malaysia

⁵International Heriot Watt Faculty, K. Zhubanov University, Aktobe, 030001, Kazakhstan

*Corresponding Authors: Ateeqa Jalal. Email: ateeqajalal@gmail.com; Mazliham Mohd Su'ud. Email: mazliham@mmu.edu.my

Received: 24 May 2025; Accepted: 05 August 2025; Published: 09 December 2025

ABSTRACT: The exponential growth of Internet of Things (IoT) devices, autonomous systems, and digital services is generating massive volumes of big data, projected to exceed 291 zettabytes by 2027. Conventional cloud computing, despite its high processing and storage capacity, suffers from increased network latency, network congestion, and high operational costs, making it unsuitable for latency-sensitive applications. Edge computing addresses these issues by processing data near the source but faces scalability challenges and elevated Total Cost of Ownership (TCO). Hybrid solutions, such as fog computing, cloudlets, and Mobile Edge Computing (MEC), attempt to balance cost and performance; however, they still struggle with limited resource sharing and high deployment expenses. This paper proposes Public Edge as a Service (PEaaS), a novel paradigm that utilizes idle resources contributed by universities, enterprises, cellular operators, and individuals under a collaborative service model. By decentralizing computation and enabling multi-tenant resource sharing, PEaaS reduces reliance on centralized cloud infrastructure, minimizes communication costs, and enhances scalability. The proposed framework is evaluated using EdgeCloudSim under varying workloads, for key metrics such as latency, communication cost, server utilization, and task failure rate. Results reveal that while cloud has a task failure rate rising sharply to 12.3% at 2000 devices, PEaaS maintains a low rate of 2.5%, closely matching edge computing. Furthermore, communication costs remain 25% lower than cloud and latency remains below 0.3, even under peak load. These findings demonstrate that PEaaS achieves near-edge performance with reduced costs and enhanced scalability, offering a sustainable and economically viable solution for next-generation computing environments.

KEYWORDS: Big data; edge as a service; edge computing

1 Introduction

The rapid advancement of digital technologies has led to an unprecedented surge in data generation from diverse sources, including social media, IoT devices, and autonomous systems, known as big data [1]. Big data is characterized by the three Vs (volume, variety, and velocity) which describe the scale, diversity, and speed at which data is produced [2]. By 2027, global data generation is projected to reach 291 zettabytes (as shown in Fig. 1), necessitating scalable and efficient solutions to manage this unexceptional volume of information [3]. Migrating this massive volume of data to the cloud introduces significant network



congestion and transmission delays, further escalating the challenges of real-time data processing and responsiveness [4,5].

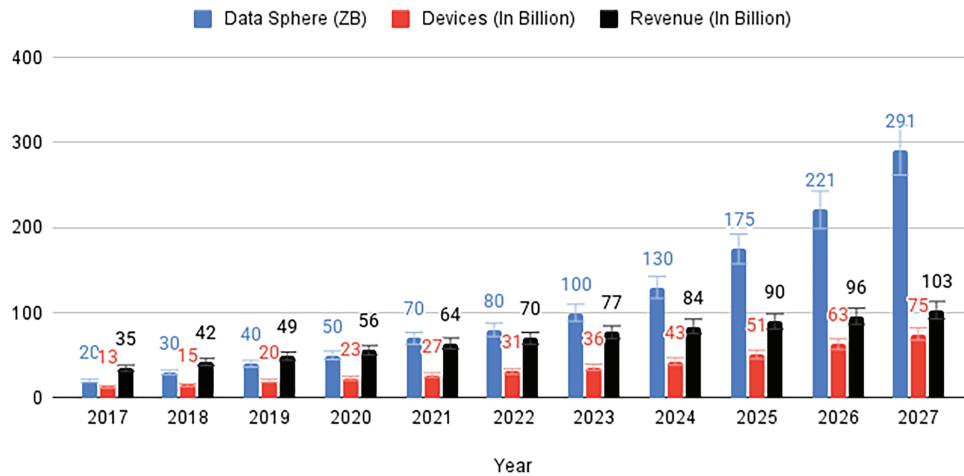


Figure 1: Devices, data and revenue forecast from 2017 to 2027

Cloud computing inherently suffers from delay issues due to its centralized infrastructure, which results in high transmission latency and network congestion when processing large volumes of data [6]. The reliance on long distance data transfer further increases delays, making cloud unsuitable for time sensitive applications.

Edge computing mitigates latency by processing data closer to its source, reducing the need for extensive cloud communication. However, edge computing is constrained by its limited scalability and high TCO, making its deployment challenging [6]. Establishing a widespread edge network requires significant financial and operational investments, having economic barriers for organizations [7]. While edge computing offers a promising alternative, its isolated deployment model lacks the resource sharing capabilities necessary for efficient scaling. The trade off between cloud and edge computing necessitates an optimized approach that effectively balances cost, scalability, and performance to address modern data management challenges [8].

Several data offloading techniques have been developed to enhance processing efficiency in cloud and edge computing environments. However, they continue to face challenges related to scalability, cost, and network congestion [9]. Fog computing extends cloud capabilities to the network's edge, enabling localized processing to reduce latency and bandwidth constraints. Similarly, cloudlets provide small scale data centers near mobile users, facilitating low latency computation for applications [10]. Another approach, Multi-access Edge Computing (MEC), integrates cloud functionalities within mobile networks, allowing computational tasks to be offloaded to cellular base stations. This reduces congestion and improves service responsiveness [11]. While these techniques improve performance, they lack an integrated framework for dynamic resource allocation and scalability, limiting their broader adoption [12]. Furthermore, they often face challenges related to scalability, deployment costs, and resource management, with a significant financial burden placed on service providers, who must invest heavily in infrastructure deployment and maintenance [13]. This underscores the need for a more flexible and collaborative approach to data offloading.

This paper introduces PEaaS as a novel framework designed to bridge the gap between cloud and edge computing. PEaaS decentralizes computing by encouraging companies, organizations, and individuals to establish local edge servers, thereby reducing data transfer costs, minimizing delays, and enhancing network reliability during peak hours. Furthermore, PEaaS offers a cost effective solution to the issue of TCO, allowing

organizations to share and optimize their computing resources more efficiently, transforming the traditional cloud-edge paradigm into a more flexible and scalable system.

This scenario can be illustrated as follows:

“Consider a security camera company providing surveillance services globally. While the company’s devices are equipped with edge computing capabilities to process data locally, deploying dedicated computing resources worldwide significantly raises device costs, imposing a financial burden on the provider. This, in turn, drives up prices, making the service less accessible to customers and limiting market reach, posing a serious financial challenge. To mitigate this issue, companies can utilize local and regional computing resources, reducing the total cost of ownership while enhancing service availability. This approach not only attracts more customers but also enables efficient local processing of the vast amounts of data generated by security cameras, particularly during peak hours, minimizing the need to migrate the data to the cloud.”

The motivation behind this research stems from the urgent need to address the inefficiencies of existing cloud and edge computing architectures. While data offloading techniques such as fog computing, cloudlets, and MEC have alleviated some challenges, they remain limited by scalability constraints, high deployment costs, and inefficient resource utilization [14]. These limitations highlight the necessity for an innovative framework that not only optimizes performance but also reduces the economic burden of edge infrastructure. By exploring the concept of PEaaS, this research aims to revolutionize the way data is processed and managed, enabling faster, more efficient services while simultaneously creating new opportunities for businesses to participate in edge computing.

Compared to existing models like MEC, Fog, and Cloudlets, PEaaS uniquely adopts a public resource-sharing approach where edge servers are contributed by diverse providers such as universities and organizations [15]. Unlike traditional models that rely on operator-managed infrastructure, PEaaS enables multi-tenant resource sharing with lightweight isolation using virtualization and containerization [16]. Service Level Agreement (SLA) ensure performance and security guarantees, while trust management mechanisms support safe participation. This collaborative model reduces infrastructure costs and enhances scalability, laying the foundation for the contributions outlined below.

Considering the above, the contributions of this paper are as follows:

- Proposing PEaaS as a solution to mitigate cloud-related delays and address the scalability limitations of edge computing.
- Reducing the total cost of ownership by eliminating the need for service providers to deploy dedicated edge infrastructure, as PEaaS enables resource sharing.
- Creating new business opportunities for companies, organizations, and individuals by allowing them to outsource their computational resources through PEaaS.

The rest of the paper is organized as follows:

[Section 2](#) reviews related literature and existing projects. [Section 3](#) presents the proposed methodology to address the current challenges in big data offloading. [Section 4](#) evaluates the proposed methodology and discusses the results. Finally, [Section 5](#) concludes the study.

2 Related Work

The increasing demand for real time data processing has led researchers to explore alternative solutions [17]. Various studies have focused on enhancing cloud computing, optimizing edge computing, and integrating both paradigms for improved efficiency [18]. This section categorizes existing research into cloud computing, edge computing, hybrid models, and cost optimization techniques. [Table 1](#) shows the detailed summary of the literature.

2.1 Cloud Computing

Cloud computing has been a fundamental approach to big data processing and storage, providing centralized resources that facilitate high-performance computing [19]. A hierarchical edge-cloud architecture proposed in [20] introduces cloud servers at the edge to improve workload distribution and minimize delays. Similarly, reference [21] integrates private and public cloud systems to balance computational loads, improving system performance.

Further, resource-sharing techniques have been explored to mitigate cloud inefficiencies. Authors in [22,23] suggest forming federations among cloud providers to optimise resource allocation, improving performance and minimising congestion. Similarly, authors in [24] propose federated management of cloud data centres to ensure balanced energy consumption and maintain service quality.

Despite these advancements, cloud computing faces many challenges. The reliance on a centralised architecture results in significant transmission delays, making it unsuitable for time-sensitive applications [25,26].

2.2 Edge Computing

Edge computing has emerged as a promising alternative to reduce data transmission delays by bringing computational resources closer to the data source. Various approaches have been proposed to enhance edge computing performance, including resource optimisation and workload distribution strategies [27]. Edge as a Service (EaaS), introduced in [28,29], allows service providers to share virtualised resources, significantly reducing infrastructure costs. Authors in [9] present an Infrastructure as a Service (IaaS) model that enables dynamic allocation of virtual machines for edge computing, improving response times and efficiency.

Optimising edge computing also involves improved resource management and decentralised architectures. Authors in [30,31] suggest forming an Edge Service Provider (ESP) federation to enhance workload distribution and ensure cost-effective edge deployments. Additionally, regional computing, discussed in [11], proposes processing big data at intermediate servers, reducing dependency on centralized cloud servers and improving efficiency.

Though edge computing provides advantages in reducing latency, it introduces challenges related to scalability and deployment cost [32]. The limited computational capacity of edge nodes restricts their ability to handle increasing workloads, making its deployment difficult [33]. Furthermore, maintaining edge infrastructure incurs high ownership costs [34].

Table 1: Summary of selected literature in edge and hybrid computing

Ref.	Main contribution	Limitation	Method
[12]	Suggests service function chaining (SFC) to improve EC routing and cost-effectiveness.	Integration with standard EC protocols presents challenges.	Mathematical model
[20]	Proposes hierarchical edge-cloud architecture to reduce latency and manage peak-hour load.	Integration with legacy infrastructure is technically challenging.	Mathematical model
[22]	Develops a federation model for cooperation across edge/cloud providers.	Limited to one geographic area (Toronto), reducing generalizability.	Prototype

(Continued)

Table 1 (continued)

Ref.	Main contribution	Limitation	Method
[31]	Explores power-aware cooperative EC system minimizing cost while balancing delay.	Cannot guarantee delay/throughput constraints in all cases.	Simulation
[35]	Proposes Hybrid Edge model extending private edge to public edge for improved processing efficiency.	Raises ownership costs by extending private services to public edge environments.	Mathematical model
[36]	Suggests Regional Computing to minimize social big data latency and network congestion.	Vague provider model; cost structure is not well-defined.	CloudSim
[34]	Studies CPU frequency and power tuning in edge devices to reduce cost and delay.	Queuing and offloading implementation complexity is high.	Simulation
[37]	Introduces Edge-as-a-Service (EaaS) for deploying edge nodes to manage workload near users.	Global server deployment is costly and hard to maintain.	Prototype
[38]	Proposes PMMRA auction for optimal edge resource sharing among devices.	Pricing fairness and practical implementation remain open issues.	Prototype
[39]	Introduces dual auction system to improve MEC economic efficiency and resource allocation.	Stakeholder acceptance and scalability are concerns.	Simulation

2.3 Hybrid Models

To utilize the benefits of both cloud and edge computing, hybrid models are proposed [21,24]. Hybrid cloud edge architectures aim to balance scalability and latency by dynamically allocating tasks between cloud and edge servers [26]. Authors in [20,29] utilize a workload distribution mechanism to optimize efficiency. Similarly, references [9,35] introduce Hybrid Edge, which integrates private and public edge resources, ensuring effective load balancing.

Federated hybrid models have also been explored, focusing on collaborative resource management [31]. Authors in [7,11,22] advocate for joint cloud edge frameworks that dynamically allocate tasks based on real time network conditions and resource availability. These models contribute to enhancing system performance while maintaining cost efficiency [32].

Although hybrid models provide a solution to the limitations of both cloud and edge computing, their implementation presents challenges. Managing the interoperability between cloud and edge environments is difficult. Migrating some tasks to the cloud leads to the same delay [33]. Additionally, workload allocation in hybrid models must be optimized to ensure that latency sensitive tasks are prioritized while balancing computational resources effectively [34].

2.4 Cost Optimization

Cost remains a critical factor in the adoption of cloud and edge computing solutions [40]. Several economic models have been introduced to optimize resource utilization and reduce operational expenses [41]. Authors in [42] examine cost trade-offs by adjusting CPU frequency and transmission power to achieve

efficient resource usage. Additionally, references [43,44] propose an integrated programming approach to minimize the cost of data distribution in edge computing.

To further reduce expenses, partial service hosting has been investigated in [45,46], where selected user requests are processed locally while others are offloaded to cloud servers, optimizing bandwidth usage. Another approach, discussed in [47], introduces decentralized storage solutions for edge computing, ensuring better resource utilization and improved performance.

Cost efficient traffic routing and resource provisioning have also been explored [48]. Service Function Chaining (SFC), presented in [12,39], optimizes network efficiency by streamlining resource management. Furthermore, references [38,49] introduce the Profit Maximization Multi-round Auction (PMMRA), an incentive-based allocation system that dynamically distributes edge resources. In the context of Mobile Edge Computing (MEC), reference [50] explores Break-even-based Double Auction (BDA) and Dynamic Pricing-based Double Auction (DPDA), which refine pricing strategies and resource allocation.

While various cost reduction strategies have been proposed, financial barriers remain a significant obstacle to widespread cloud and edge computing adoption [51]. Deploying and maintaining edge infrastructure is expensive, requiring sustainable revenue models to ensure long term viability [52]. Furthermore, optimizing resource allocation without compromising performance remains a key challenge in cost efficient cloud and edge deployments.

3 Proposed Framework

The proposed framework is structured across three distinct layers: (i) Users' devices, (ii) PEaaS and (iii) Cloud servers. The Users' devices layer includes all computing, smart, and IoT devices connected to the system. PEaaS represent resources offered by local entities (e.g., universities, institutions or cellular companies) under a service level agreement, with the system's algorithm allocating user workloads to these servers up to a predefined threshold. Finally, Cloud Servers, owned by cloud service providers, handle permanent storage and processing of user data. Fig. 2 shows the structure of the proposed framework.

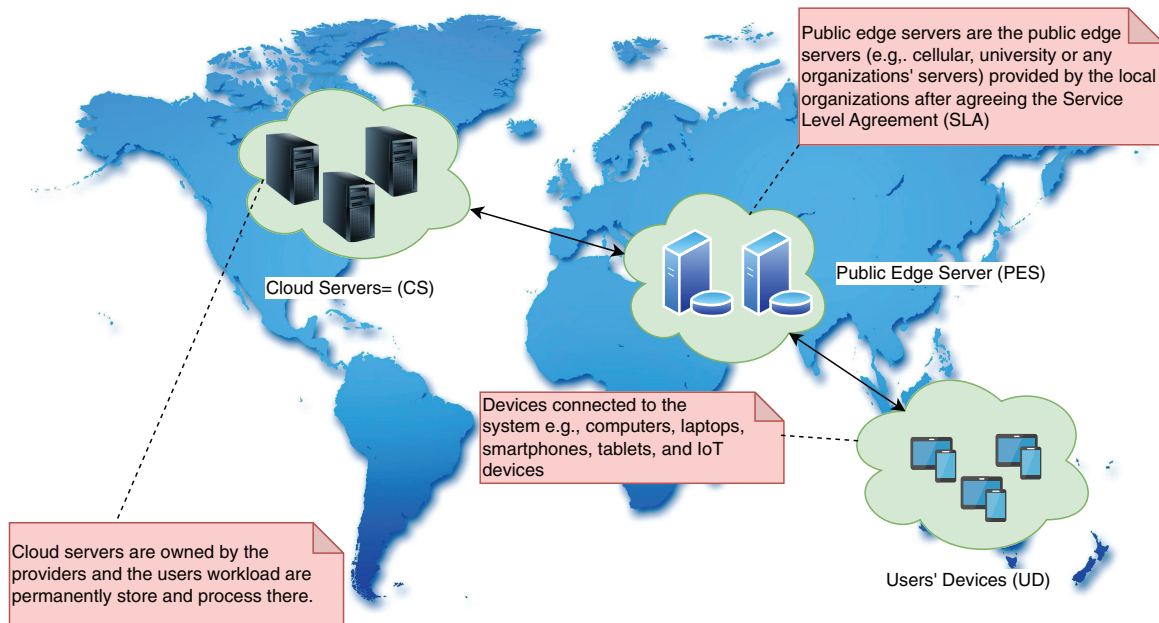


Figure 2: Proposed structure of public edge as a service

3.1 Devices Layer

The Devices Layer consists of users' devices (e.g., computers, smartphones, and IoT gadgets) all of which communicate with the system through the modern 5G technologies, ensuring minimal data transmission delay between devices and computing servers [46]. The delay incurred at this layer depends on both the propagation and transmission delay, which can be represented by the following equations:

$$Del_{prop} = \frac{Dis}{tr_s} \quad (1)$$

where Del_{prop} shows the propagation delay, Dis shows the distance and tr_s shows the transmission speed.

$$Del_{trans} = \frac{W}{T_s} \quad (2)$$

where Del_{tran} shows the transmission delay, W shows the workload and T_s shows the channel capacity.

The congestion in the Devices Layer can be expressed as:

$$Del \propto \sum_{i=1}^n \left(\frac{Dev_{act_i}}{BW_i} \right) \quad (3)$$

Here, Del is the delay, Dev_{act} refers to the number of active devices per base station or any other internet service provider, and BW is the bandwidth available for communication. As the number of active devices increases, congestion and delays also rise. Additionally, the relationship between congestion and workload is represented by:

$$Con \propto \frac{W_i}{BW_i \cdot t_s} \quad (4)$$

In this equation, Con is the congestion, W is the workload, and BW is the bandwidth at a given time t_s . Finally, congestion can also be modeled based on available bandwidth as:

$$Con \propto \left(\frac{1}{BW^\alpha} \right) \quad (5)$$

In this last equation, α reflects the bandwidth elasticity.

3.2 Public Edge Layer

The Public Edge Layer forms the core of the framework. It utilizes idle computational resources provided by local entities, reducing the cost of edge infrastructure for service providers or the edge provided by the cellular companies on the base station. At the public edge, delay factors include both processing delay and queuing delay, represented by:

$$Del_{process} = \frac{S}{P_r} \quad (6)$$

where $Del_{process}$ is the processing delay, S represents the data size, and P_r is the processing rate of the public edge server.

$$Del_{queue} = \frac{L \times a}{R} \quad (7)$$

Similarly, Del_{queue} is the queuing delay, L is the packet length, a is the packet arrival rate, and R is the processing rate.

Congestion at the public edge depends on computational resources and can be represented as:

$$Con \propto \frac{W_{edge}}{Com^k} \quad (8)$$

Here, W_{edge} is the workload directed toward the edge servers, Com is the computational power of the server, and k reflects the server's efficiency. Higher computational power results in significantly reduced congestion.

3.3 Cloud Layer

The Cloud Layer consists of high performance servers with vast storage and processing capacities. While the cloud stores all data, only a small fraction of the processing occurs here, with the bulk handled at the public edge. During peak periods, a large workload directed toward cloud servers can cause congestion. The relationship between delay and workload in the cloud can be expressed as:

$$Del_{cc} \propto \sum_{i=1}^n \frac{Dis_i}{Cap_i \cdot BW_i} \quad (9)$$

In this equation, Del_{cc} shows the cloud delay, Dis is the distance between devices and the cloud, Cap represents the server's processing capacity, and BW refers to the network bandwidth. As distance and workload increase, delay surges.

The cloud layer inherits the properties of congestion and delay as noted in previous layers, further increases by long distances and massive workloads characteristic of cloud servers.

3.4 Algorithm

Algorithm 1 is designed to efficiently manage big data generated by devices $d = \{d_1, d_2, \dots, d_n\}$ and their corresponding workloads $w = \{w_1, w_2, \dots, w_m\}$. It follows a hierarchical offloading strategy across edge, PES, and cloud servers to ensure efficient task distribution based on latency, congestion, and network conditions.

Algorithm 1: Edge-aware big data offloading

Input: Big data generated by devices and their respective workloads

Output: Decision on data processing and migration to Edge Servers, Public Edge Servers (PES), and Cloud Servers

foreach device generating data **do**

Compute Del_{prop} and Del_{trans} using Eqs. (1) and (2)

Compute Del using Eq. (3)

if ($Del_{prop} + Del_{trans} < \delta_1$) **AND** ($Del < \delta_2$) **then**

Process locally on Edge Server;

// Low latency and congestion

end

else

Compute Con using Eqs. (4) and (5)

if ($Con < \theta_1$) **AND** (Bandwidth $> \theta_2$) **then**

Offload to PES;

// Optimal network condition

(Continued)

Algorithm 1 (continued)

```

    end
  else
    if Offloading request occurs during off-peak hours then
      if Available capacity on PES satisfies Eq. (8) then
        Offload to PES
      end
    else
      Offload to Cloud Servers
    end
  end
else
  Offload to Cloud Servers
end
end
end
end
if Urgent workload requires migration (based on Eq. (9)) then
  Offload to Cloud Servers for further migration
end

```

Initially, the algorithm computes propagation and transmission delays (Eqs. (1), (2) and device-level congestion (Eq. (3)). If the total delay and congestion are below predefined thresholds δ_1 and δ_2 , the task is processed locally on the edge server to ensure fast execution.

If the data is not time-sensitive, the algorithm evaluates the network condition by computing congestion and bandwidth availability using Eqs. (4) and (5). If congestion is below θ_1 and bandwidth exceeds θ_2 , the task is offloaded to the PES, ensuring low-latency and cost-effective processing.

In suboptimal network conditions, the algorithm checks whether the task is being generated during off-peak hours. If so, and if the PES has sufficient processing capacity (as verified using Eq. (8)), the task is offloaded to PES. Otherwise, or during peak traffic, the task is offloaded to the cloud to avoid overloading local resources.

Finally, if an urgent workload demands migration to another location, the algorithm uses Eq. (9) to assess the cloud offloading decision for wider geographic accessibility.

4 Evaluation

To evaluate the effectiveness of the proposed framework, we employed EdgeCloudSim [53], a simulation tool tailored for edge computing analysis. Built on CloudSim [54], it simulates various network conditions, resource management strategies, and workload distributions, making it suitable for assessing edge and cloud performance. The simulation environment was configured using real-world network parameters, as detailed in Table 2, with workloads varying from 100 to 2000 devices. The experiment included one cloud server, one PEaaS, and five edge servers, with each scenario repeated ten times to ensure consistency in delay and cost measurements.

Table 2: Simulation parameters and settings

Parameters	Description
Quantity of cloud servers	1
Quantity of PEaaS	1
Quantity of edge servers	5
Minimum devices	100
Maximum devices	2000
Minimum upload work	20 KB
Maximum upload work	2500 KB
Minimum download work	25 KB
Maximum download work	1250 KB
Simulation time	One hour
Delay	Round trip delay
Cost	Communication cost
Simulation repetition for average calculation	10 times

The evaluation follows an upload and download task per request model, consistent with the methodology used in the initial EdgeCloudSim publication [9]. The upload workload ranged from 20 to 2500 KB, while the download workload varied from 25 to 1250 KB. A round-trip delay metric was used to assess performance, and communication cost was considered as a primary cost factor.

4.1 Experimental Setup

The experimental setup consists of three distinct scenarios to manage big data through edge, public edge, and cloud computing paradigms. First, we evaluate the performance and resource utilization associated with processing data using edge computing, where tasks are handled at the device edge servers. Second, we analyze the efficiency of PES provided by local organizations, assessing their impact on latency and costs. Third, we examine the performance of cloud computing, where data is offloaded to centralized cloud servers for processing. Table 2 presents the simulation parameters used for evaluation.

4.1.1 Edge Context

For the Edge Computing setup, a single edge server was created. The computing specification included 4 cores, 5000 MIPS, 8000 MB of RAM, and 150,000 MB of storage capacity. On this server, 2 virtual machines (VMs) were deployed, each equipped with 2 cores. The setup was designed to handle data from 100 to 20,000 devices. The simulation was repeated ten times, with each session lasting half an hour to ensure reliability. The performance and communication costs were measured to evaluate performance.

4.1.2 Public Edge Context

For the PES scenario, a decentralized edge server was deployed, utilizing resources from various local organizations and service providers. The server consisted of 8 cores, 10,000 MIPS, 16,000 MB of RAM, and 150,000 MB of storage capacity. Four VMs were deployed, each equipped with 2 cores. This setup facilitated distributed task processing, ensuring lower latency and cost efficient operations for up to 20,000 connected devices. The simulation was repeated ten times to evaluate its performance.

4.1.3 Cloud Context

In the cloud computing context, a single cloud data center was established to process big data. The server included 32 cores, 200,000 MIPS, 64,000 MB of RAM, and 200,000 MB of storage capacity. Eight VMs were deployed within this infrastructure. The setup managed up to 20,000 devices, processing 300,000 tasks globally. The simulation was executed ten times, each lasting 30 min.

4.2 Result and Discussion

The experimental results provide a comparative analysis of Edge, PEaaS, and Cloud computing environments in terms of key performance metrics, including task failure rate, service time, processing time, network delay, server utilization and communication cost. The evaluation is conducted across varying numbers of devices, ranging from 100 to 2000, to assess scalability and efficiency. The primary objective is to demonstrate how PEaaS outperforms traditional Cloud computing by reducing failure rates, improving response times, and optimizing resource utilization. Each metric is analyzed separately, with column-wise bar graphs providing a clear visual representation of performance trends. The findings highlight that Edge computing offers the lowest latency, while PEaaS balances reliability and performance, making it a superior alternative for handling computationally intensive and latency sensitive tasks.

4.2.1 Task Failure Rate Analysis

Task failure rates vary significantly across the three computing paradigms; edge, PEaaS, and cloud, as shown in Fig. 3. The failure rate trend increases with the number of connected devices, however, the rate of increase is notably different among these environments.

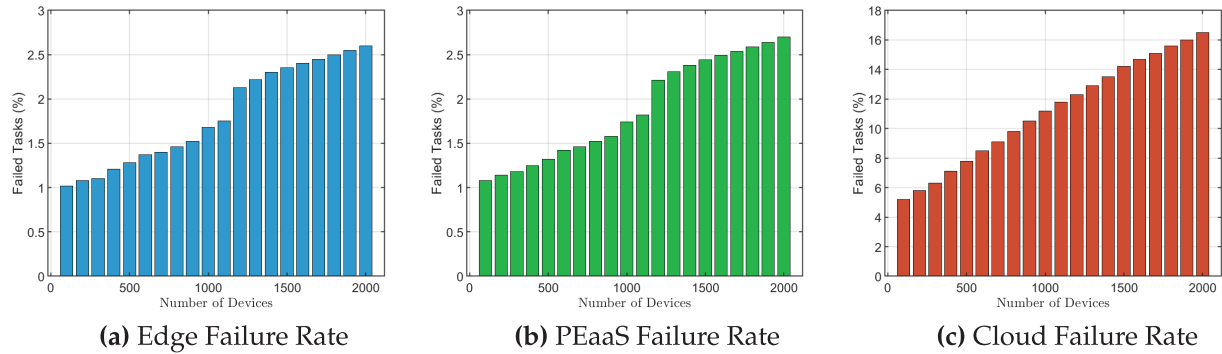


Figure 3: Comparison of task failure (in %) rate across edge, PEaaS, and cloud environments

At lower device counts, all three paradigms show relatively low failure rates. As illustrated in Fig. 3a, edge computing maintains a stable failure rate, starting at 1.02% for 100 devices and gradually rising to 2.6% at 2000 devices. This stability is attributed to localized processing, which reduces network dependencies. However, edge computing has limited resources, making it expensive for providers to deploy and maintain additional edge nodes for scalability.

Fig. 3b shows that PEaaS follows a similar trajectory to edge computing, starting at 1.08% for 100 devices and reaching 2.5% at 2000 devices. The slightly higher initial failure rate compared to edge is due to its distance, however, it remains close to edge performance throughout the scaling process. Unlike edge computing, PEaaS benefits from dynamic resource allocation, reducing the need for extensive physical infrastructure investment while maintaining a low failure rate.

On the other hand, cloud computing exhibits significantly higher failure rates, as shown in Fig. 3c. The failure rate begins at 5.2% for 100 devices and escalates sharply to 12.3% at 2000 devices. This increase is primarily due to distance and centralized resource bottlenecks, which cause delays in processing and task execution.

Overall, PEaaS presents a compelling balance between reliability and scalability. While edge computing ensures the lowest failure rates, its resource constraints make large scale deployment challenging. Cloud, though scalable, suffers from severe performance degradation due to long delays. PEaaS emerges as an optimal solution, closely matching the failure rates of edge computing while leveraging its distributed infrastructure to offer better scalability and cost efficiency.

4.2.2 Service Time

Service time reflects the end-to-end duration from task dispatch to result delivery, combining both processing and network delays. It is a key indicator of system responsiveness and directly impacts the feasibility of latency-sensitive applications.

As depicted in Fig. 4, the PEaaS environment achieves the lowest service time, starting at approximately 0.14 s and increasing steadily to 2.75 s as the number of devices scales to 2000. This highlights its ability to offer rapid processing while avoiding network overhead typically seen in cloud computing. Edge computing, while also delivering low service times due to close proximity to end devices, exhibits slightly higher service time (ranging from 0.15 to 3.24 s) because of its limited computing capacity under growing workloads.

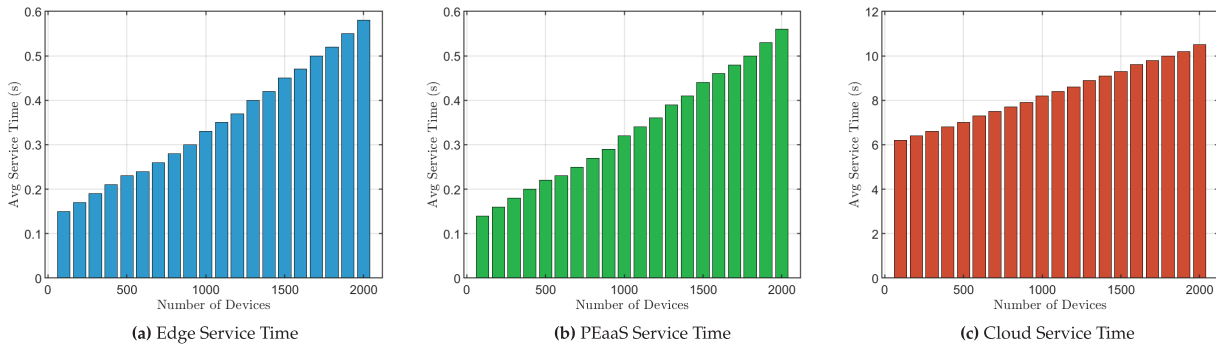


Figure 4: Comparison of service time (ms) across edge, PEaaS, and cloud environments

In contrast, cloud computing shows a significantly higher service time, beginning at 6.2 s and reaching 9.98 s as device count increases. This sharp rise is attributed to network transmission delays, congestion, and centralized resource contention, which become more prominent with larger data volumes.

Overall, while edge excels in responsiveness and cloud in resource availability, PEaaS emerges as the most balanced and efficient option. It combines the low-latency advantage of edge computing with the scalability and resource distribution of cloud systems, making it the most suitable choice for large-scale, time-critical vehicular offloading scenarios.

4.2.3 Processing Time

Processing time determines system efficiency by measuring the duration required to execute tasks. Fig. 5 presents a comparative analysis of processing time across edge, PEaaS, and cloud environments. Edge computing achieves the lowest processing time due to localized task execution, minimizing transmission

overhead. PEaaS follows closely, leveraging optimized resource allocation to maintain near edge performance, while cloud exhibits significantly higher processing times due to centralized workload management and network delays.

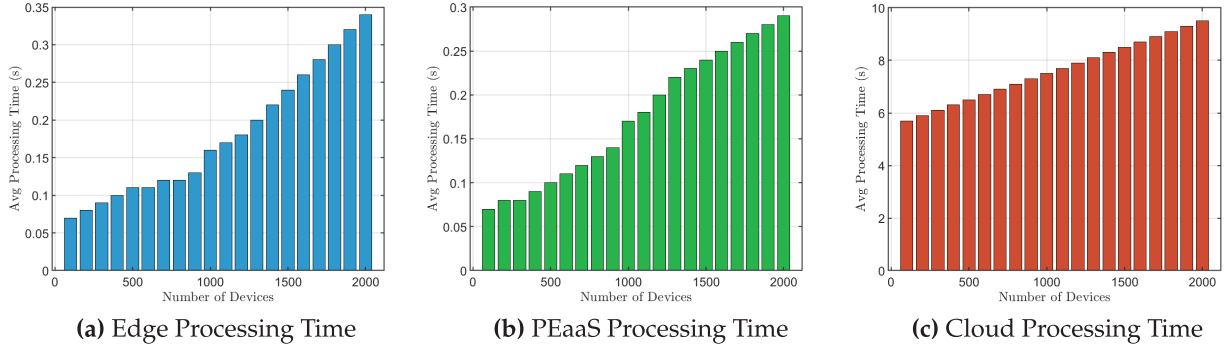


Figure 5: Comparison of processing time (ms) across edge, PEaaS, and cloud environments

At a lower workload of 100 devices, processing times for edge, PEaaS, and cloud are recorded as 0.98, 1.05, and 5.4 s, respectively. As device count scales to 2000, processing time increases across all environments, however, with varying trends. Edge maintains a moderate increase, reaching 2.4 s, while PEaaS reaches 2.6 s, demonstrating its capability to handle increased workloads efficiently. In contrast, cloud processing time surges to 9.8 s due to congestion, server queuing delays, and centralized resource contention.

The results emphasize PEaaS as a viable alternative to edge computing, offering comparable processing time while avoiding the capital-intensive requirement of deploying dedicated edge servers. Unlike cloud, which suffers from performance degradation as workload increases, PEaaS ensures low latency processing through distributed and optimized task execution. This makes PEaaS a scalable and cost effective solution for task offloading, maintaining efficiency across varying workload conditions.

4.2.4 Network Delay

Network delay is an important parameter in data offloading decisions. It directly affects task execution efficiency and overall system responsiveness. Fig. 6 presents the network delay trends across Edge, PEaaS, and cloud environments. Edge computing shows the lowest network delay due to its close locality to end devices, minimizing transmission overhead. PEaaS, utilizing distributed infrastructure, maintains a moderate delay. This ensures a balance between latency and scalability. On the other hand, cloud experiences significantly higher network delays due to data traversing multiple hops over wide area networks, introducing congestion and queuing delays.

At lower workloads of 100 devices, network delays for Edge, PEaaS, and Cloud are recorded at 0.06, 0.06, and 5.6 s, respectively. As the device count increases to 2000, Edge delay increases slightly to 0.33 s, while PEaaS reaches 0.28 s. This shows its capacity to manage increasing workloads efficiently. Cloud delay increases drastically to 9.4 s due to high dependency on centralized servers and network delays.

These results highlight PEaaS as an interesting alternative for task offloading, maintaining significantly lower network delays than Cloud. It also ensures scalability beyond the limitation of Edge computing. By utilizing shared infrastructure and distributed processing at intermediary nodes, PEaaS optimally balances network performance and system responsiveness, making it an efficient choice for time sensitive applications.

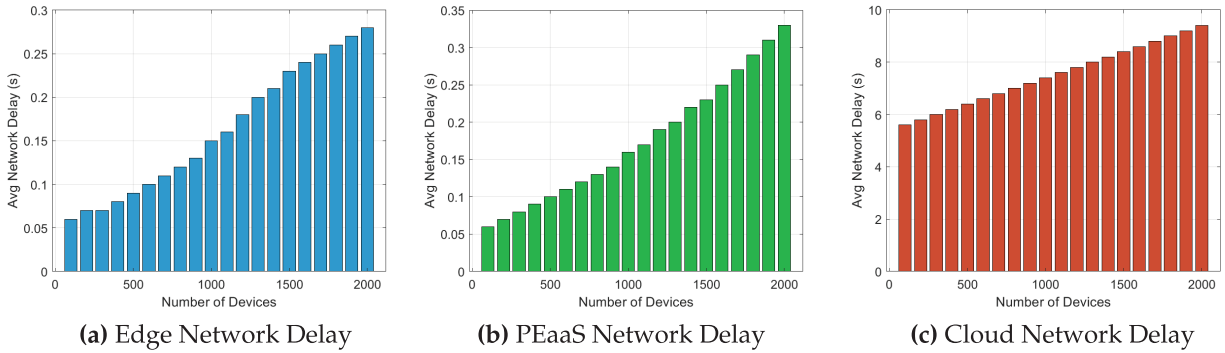


Figure 6: Comparison of network delay (ms) across edge, PEaaS, and cloud environments

4.2.5 Server Utilization

Server utilization is a key metric for evaluating the efficiency of computational resource allocation in Edge, PEaaS, and Cloud paradigms. Fig. 7 illustrates the utilization for each paradigm, highlighting the impact of increasing workloads on computational efficiency.

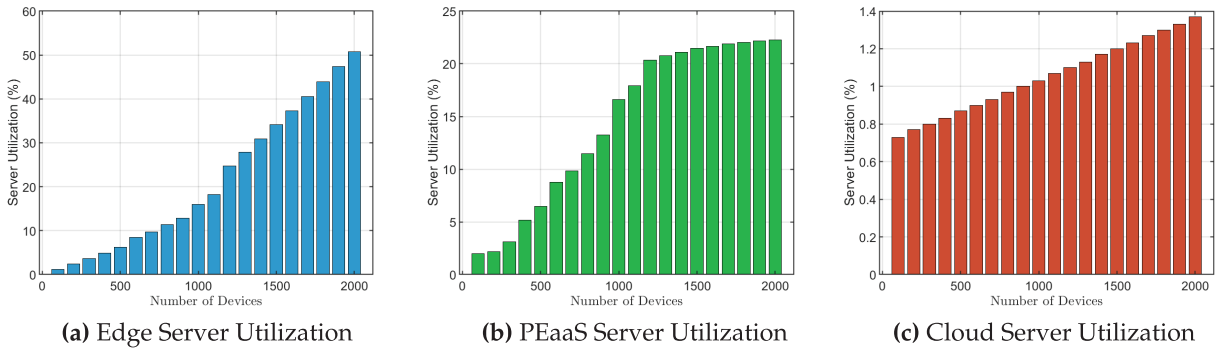


Figure 7: Comparison of server utilization (in %) across edge, PEaaS, and cloud environments

Edge computing shows the highest server utilization due to its constrained resources and limited scalability. As the device count increases, Edge servers reach maximum capacity, with utilization increasing from 1.15% at 100 devices to 50.8% at 2000 devices (7a). This high utilization indicates efficient resource usage, however, also demonstrates the limitations of Edge in handling big data without significant infrastructure expansion.

PEaaS, on the other hand, effectively balances resource allocation by distributing workloads across a broader, shared infrastructure. As shown in Fig. 7b, PEaaS utilization starts at 1.98% with 100 devices and steadily increases to 22.25% at 2000 devices. This moderate utilization ensures that resources are not over-utilized while maintaining efficiency in task execution.

Cloud infrastructure shows the lowest utilization levels, as shown in Fig. 7c, with values ranging from 0.73% to 1.37% across all device counts. Despite its massive computational capabilities, Cloud remains underutilized due to network induced delays and inefficiencies in centralized processing.

The results indicate that PEaaS offers an optimal balance between Edge and Cloud computing. While Edge servers risk overloading and Cloud suffers from underutilization, PEaaS efficiently scales with increasing workloads. This ensures effective resource usage without sacrificing performance. This makes PEaaS a cost effective and scalable solution for handling diverse computational demands in real time applications.

4.2.6 Communication Cost

Communication cost is a critical factor in determining the feasibility of offloading tasks to Edge, PEaaS, or Cloud environments. Fig. 8 presents a comparative analysis of communication costs across the three architectures, illustrating how costs scale with increasing device count.

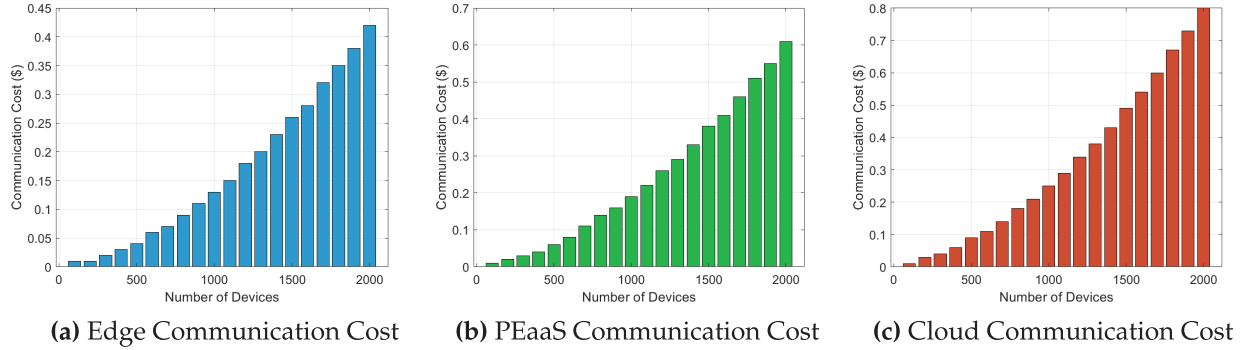


Figure 8: Comparison of communication cost (\$) across edge, PEaaS, and cloud environments

Edge computing maintains the lowest communication cost due to minimal data transmission over the network. As shown in Fig. 8a, the cost remains low, increasing from 0.01 at 100 devices to 0.42 at 2000 devices. The limited network dependency of Edge computing results in minimal overhead costs, making it a cost effective option for latency sensitive applications with minimal offloading requirements.

PEaaS exhibits a moderate cost structure, balancing network usage and efficiency. As seen in Fig. 8b, the cost starts at 0.01 for 100 devices and gradually rises to 0.61 at 2000 devices. This increase is due to the reliance on shared edge servers, where some data must be transmitted over the network for optimal processing. However, PEaaS ensures lower costs compared to Cloud by reducing unnecessary long distance transmissions while maintaining scalability.

Cloud, in contrast, experiences the highest communication cost due to significant data movement across wide networks. Fig. 8c highlights that the cost starts at 0.01 for 100 devices, however, increases sharply to 0.8 at 2000 devices. The centralized nature of Cloud computing requires extensive network usage, leading to higher costs associated with bandwidth consumption, energy and data transfer.

The results demonstrate that PEaaS offers an optimal trade-off between cost and scalability. While Edge provides the lowest cost, however, it suffers from scalability limitations, and Cloud incurs the highest cost due to excessive network utilization, PEaaS efficiently balances network overhead while ensuring cost effective resource allocation. These findings suggest that PEaaS is the preferred choice for applications requiring cost efficiency while maintaining network scalability.

The results demonstrate that PEaaS offers a balanced and efficient approach to task offloading compared to Edge and Cloud environments, as shown in Table 3. While Edge computing provides the lowest failure rate and latency, its limited resources necessitate significant capital investment for infrastructure deployment. Cloud computing, on the other hand, suffers from high failure rates, network congestion, underutilization of resources, and higher communication costs due to extensive data transfers over wide area networks.

Table 3: Comparative analysis of edge computing (EC), PEaaS, and cloud computing (CC) for IoT big data processing

Parameters	Edge computing (EC)	PEaaS	Cloud computing (CC)
Delay	Very low	Low	High
Response time	Very low	Low	High
Latency	Very low	Low	High
Distance from IoT devices	Near to the devices	Regionally distributed	Long distance from devices
Network model	Distributed	Decentralized (Public Edge Nodes)	Centralized
Security	High	Low	Moderate
Server mobility	Possible	No	No
Real-time processing	Highly responsive	Optimized for regional real-time processing	High latency affects real-time processing
Storage capacity	Limited	Moderate	High scalability
Cost	High deployment cost	Cost-effective resource sharing	High operational cost
Scalability	Limited	Flexible	Highly scalable
Capital Cost	High due to dedicated infrastructure	Reduced due to shared infrastructure	High due to large-scale infrastructure

PEaaS bridges this gap by utilizing publicly available edge resources from providers (e.g., organizations, cellular companies or individual) ensuring low latency, optimized resource utilization, and reduced infrastructure costs. Furthermore, PEaaS maintains significantly lower communication costs compared to Cloud, making it a financially viable option for large scale deployments. While Cloud incurs substantial data transmission expenses, PEaaS efficiently balances cost and performance by limiting long distance data transfers.

Going with PEaaS, organizations can achieve near Edge performance without the burden of deploying and maintaining dedicated hardware, making it a cost effective and scalable solution for big data offloading. The findings suggest that PEaaS is the optimal choice for applications requiring a balance between efficiency, cost, and scalability. It is a sustainable alternative to traditional Edge and Cloud computing paradigms.

5 Conclusion and Future Work

This paper introduces PEaaS as a novel framework for managing the increasing demands of IoT big data. By shifting from a provider owned infrastructure to a shared edge service model, PEaaS enables organizations to hire edge resources rather than investing in dedicated infrastructure, significantly reducing ownership and capital costs while improving computational efficiency. The performance evaluation shows that PEaaS effectively reduces latency and operational expenses compared to traditional cloud approaches. Utilizing distributed public edge servers, the framework minimizes network congestion and transmission delays, resulting in faster response times and optimized resource utilization. The proposed model addresses the TCO challenge, allowing service providers to dynamically allocate resources based on demand. With this, provider can run their services without the financial burden of maintaining private edge infrastructure.

Future research will focus on enhancing the dynamic resource allocation mechanisms by integrating AI decisions to further optimize cost efficiency and performance. Additionally, the impact of 5G and 6G networks on PEaaS will be explored to assess their role in improving connectivity and scalability.

Acknowledgement: Not applicable.

Funding Statement: The authors declare that no funds, grants, or other support were received for this study.

Author Contributions: Ateeqa Jalal conceptualized the idea, proposed the methodology, and wrote the initial manuscript draft. Umar Farooq and Ihsan Rabbi provided supervision, technical guidance, and critical revisions. Afzal Badshah contributed to simulation design, result analysis, and manuscript refinement. Aurangzeb Khan assisted in data interpretation and validation. Muhammad Mansoor Alam contributed to the review of related work and overall technical insights. Mazliham Mohd Su'ud provided project oversight, expert consultation, and final approval of the manuscript. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The datasets generated and/or analyzed during the current study will be made available from the corresponding authors on reasonable request.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

Ethics Approval: Not applicable.

References

1. Badshah A, Daud A, Alharbey R, Banjar A, Bukhari A, Alshemaimri B. Big data applications: overview, challenges and future. *Artif Intell Rev.* 2024;57(11):290. doi:10.1007/s10462-024-10938-5.
2. Laney D. 3D data management: controlling data volume, velocity and variety. *META Group Res Note.* 2001;6(70):1.
3. Statista. Volume of data/information created, captured, copied, and consumed worldwide from 2010 to 2020, with forecasts from 2021 to 2025. 2023 [cited 2025 Aug 4]. Available from: <https://www.statista.com/statistics/871513/worldwide-data-created/>.
4. Alshemaimri B, Badshah A, Daud A, Bukhari A, Alsini R, Alghushairy O. Regional computing approach for educational big data. *Sci Rep.* 2025;15(1):7619. doi:10.1038/s41598-025-92120-7.
5. Zou X, Yuan J, Shilane P, Xia W, Zhang H, Wang X. From hyper-dimensional structures to linear structures: maintaining deduplicated data's locality. *ACM Trans Storage (TOS).* 2022;18(3):1–28. doi:10.1145/3507921.
6. Sirbu DI, Negru C, Pop F, Esposito C. Storage service for edge computing. In: *SAC '21: Proceedings of the 36th Annual ACM Symposium on Applied Computing; 2021 Mar 22–26; Virtual Event, Republic of Korea.* New York, NY, USA: Association for Computing Machinery; 2021. p. 1165–71. doi:10.1145/3412841.3441991.
7. Arroba P, Buyya R, Cárdenas R, Risco-Martín JL, Moya JM. Sustainable edge computing: challenges and future directions. *arXiv:2304.04450.* 2023.
8. Huang W, Li T, Cao Y, Lyu Z, Liang Y, Yu L, et al. Safe-NORA: safe reinforcement learning-based mobile network resource allocation for diverse user demands. In: *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management; 2023 Oct 21–25; Birmingham, UK.* p. 885–94.
9. Sonmez C, Ozgovde A, Ersoy C. Edgecloudsim: an environment for performance evaluation of edge computing systems. *Trans Emerg Telecomm Technol.* 2018;29(11):e3493. doi:10.1002/ett.3493.
10. Ma Y, Li T, Zhou Y, Yu L, Jin D. Mitigating energy consumption in heterogeneous mobile networks through data-driven optimization. *IEEE Trans Netw Serv Manag.* 2024;21(4):4369–82. doi:10.1109/tnsm.2024.3416947.
11. Shi W, Cao J, Zhang Q, Li Y, Xu L. Edge computing: vision and challenges. *IEEE Internet Things J.* 2016;3(5):637–46. doi:10.1109/jiot.2016.2579198.
12. Zhou Z, Wu Q, Chen X. Online orchestration of cross-edge service function chaining for cost-efficient edge computing. *IEEE J Sel Areas Commun.* 2019;37(8):1866–80. doi:10.1109/jsac.2019.2927070.

13. Badshah A, Daud A, Khan HU, Alghushairy O, Bukhari A. Optimizing the over and underutilization of network resources during peak and off-peak hours. *IEEE Access*. 2024;12(2):82549–59. doi:10.1109/access.2024.3402396.
14. Sun G, Liao D, Zhao D, Xu Z, Yu H. Live migration for multiple correlated virtual machines in cloud-based data centers. *IEEE Trans Serv Comput*. 2015;11(2):279–91. doi:10.1109/tsc.2015.2477825.
15. Chen P, Luo L, Guo D, Tang G, Zhao B, Li Y, et al. Why and how lasagna works: a new design of air-ground integrated infrastructure. *IEEE Netw*. 2024;38(2):132–40. doi:10.1109/mnet.2024.3350025.
16. Cheng Y, Deng X, Li Y, Yan X. Tight incentive analysis of Sybil attacks against the market equilibrium of resource exchange over general networks. *Games Econ Behav*. 2024;148(2):566–610. doi:10.1016/j.geb.2024.10.009.
17. Research A, Consulting. Big data market size to reach USD 473.6 billion by 2030. 2023 [cited 2025 Aug 4]. Available from: <https://www.acumenresearchandconsulting.com/press-releases/big-data-market>.
18. Akherfi K, Gerndt M, Harroud H. Mobile cloud computing for computation offloading: issues and challenges. *Appl Comput Inform*. 2018;14(1):1–16. doi:10.1016/j.aci.2016.11.002.
19. Wang P, Song W, Qi H, Zhou C, Li F, Wang Y, et al. Server-initiated federated unlearning to eliminate impacts of low-quality data. *IEEE Trans Serv Comput*. 2024;17(3):1196–211. doi:10.1109/tsc.2024.3355188.
20. Tong L, Li Y, Gao W. A hierarchical edge cloud architecture for mobile computing. In: *IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications*; 2016 Apr 10–14; San Francisco, CA, USA. p. 1–9.
21. Talebkhah M, Sali A, Marjani M, Gordan M, Hashim SJ, Rokhani FZ. Edge computing: architecture, applications and future perspectives. In: *2020 IEEE 2nd International Conference on Artificial Intelligence in Engineering and Technology (IICAIET)*; 2020 Sep 26–27; Kota Kinabalu, Malaysia. p. 1–6.
22. Cao X, Tang G, Guo D, Li Y, Zhang W. Edge federation: towards an integrated service provisioning model. *IEEE/ACM Trans Netw*. 2020;28(3):1116–29. doi:10.1109/tnet.2020.2979361.
23. Cao K, Li L, Cui Y, Wei T, Hu S. Exploring placement of heterogeneous edge servers for response time minimization in mobile edge-cloud computing. *IEEE Trans Industr Inform*. 2021;17(1):494–503. doi:10.1109/tii.2020.2975897.
24. Fernández CM, Rodríguez MD, Muñoz BR. An edge computing architecture in the internet of things. In: *2018 IEEE 21st International Symposium on Real-Time Distributed Computing (ISORC)*; 2018 May 29–31; Singapore. p. 99–102.
25. Ren J, Yu G, He Y, Li GY. Collaborative cloud and edge computing for latency minimization. *IEEE Trans Veh Technol*. 2019;68(5):5031–44. doi:10.1109/tvt.2019.2904244.
26. Zaslavsky A, Perera C, Georgakopoulos D. Sensing as a service and big data. *arXiv:1301.0159*. 2013.
27. Alsahfi T, Badshah A, Aboulola OI, Daud A. Optimizing healthcare big data performance through regional computing. *Sci Rep*. 2025;15(1):3129. doi:10.1038/s41598-025-87515-5.
28. Davy S, Famaey J, Serrat J, Gorricho JL, Miron A, Dramitinos M, et al. Challenges to support edge-as-a-service. *IEEE Commun Mag*. 2014;52(1):132–9. doi:10.1109/mcom.2014.6710075.
29. Li C, Xue Y, Wang J, Zhang W, Li T. Edge-oriented computing paradigms: a survey on architecture design and system management. *ACM Comput Surv (CSUR)*. 2018;51(2):1–34. doi:10.1145/3154815.
30. Zhu R, Lin X, Wu S, Fu W, Luo L, Zhang B. Resource sharing among edge service providers: modeling and solution. In: *2023 9th International Conference on Big Data and Information Analytics (BigDIA)*; 2023 Decr 15–17; Haikou, China. p. 322–9.
31. Premsankar G, Di Francesco M, Taleb T. Edge computing for the internet of things: a case study. *IEEE Internet Things J*. 2018;5(2):1275–84. doi:10.1109/jiot.2018.2805263.
32. Singh S. Optimize cloud computations using edge computing. In: *2017 International Conference on Big Data, IoT and Data Science (BIGD)*; 2017 Dec 20–22; Pune, India. p. 49–53.
33. Sun X, Ansari N. PRIMAL: profit maximization avatar placement for mobile edge computing. In: *2016 IEEE International Conference on Communications (ICC)*; 2016 May 22–27; Kuala Lumpur, Malaysia. p. 1–6.
34. Chen W, He Y, Qiao J. Cost minimization for cooperative mobile edge computing systems. In: *2019 28th Wireless and Optical Communications Conference (WOCC)*; 2019 May 9–10; Beijing, China. p. 1–5.
35. Gu S, Guo D, Tang G, Luo L, Sun Y, Luo X. HyEdge: a cooperative edge computing framework for provisioning private and public services. *ACM Trans Internet Things*. 2023;4(2):1–23. doi:10.1145/3585078.

36. Badshah A, Iwendi C, Jalal A, Hasan SSU, Said G, Band SS, et al. Use of regional computing to minimize the social big data effects. *Comput Ind Eng.* 2022;171:108433.
37. Zhang M, Cao J, Yang L, Zhang L, Sahni Y, Jiang S. Ents: an edge-native task scheduling system for collaborative edge computing. In: 2022 IEEE/ACM 7th Symposium on Edge Computing (SEC); 2022 Dec 5–8; Seattle, WA, USA. p. 149–61.
38. Wang Q, Guo S, Liu J, Pan C, Yang L. Profit maximization incentive mechanism for resource providers in mobile edge computing. *IEEE Trans Serv Comput.* 2022;15(1):138–49. doi:10.1109/tsc.2019.2924002.
39. Sun W, Liu J, Yue Y, Zhang H. Double auction-based resource allocation for mobile edge computing in industrial internet of things. *IEEE Trans Industr Inform.* 2018;14(10):4692–701. doi:10.1109/tii.2018.2855746.
40. Ning F, Shi Y, Tong X, Cai M, Xu W. Manufacturing cost estimation based on similarity. *Int J Comput Integr Manuf.* 2023;36(8):1238–53. doi:10.1080/0951192x.2023.2165160.
41. Liu Q, Zeng H, Chen M, Liu L. Cost minimization in multi-path communication under throughput and maximum delay constraints. In: IEEE INFOCOM 2020-IEEE Conference on Computer Communications; 2020 Jul 6–9; Toronto, ON, Canada. p. 2263–72.
42. Badshah A, Daud A, Alhajlah M, Alsahfi T, Alshemaimri B, Ur-Rehman G. Smart cities' big data: performance and cost optimization with regional computing. *IEEE Access.* 2024;12(70):128896–908. doi:10.1109/access.2024.3457269.
43. Xia X, Chen F, He Q, Grundy JC, Abdelrazek M, Jin H. Cost-effective app data distribution in edge computing. *IEEE Trans Parallel Distrib Syst.* 2020;32(1):31–44. doi:10.1109/tpds.2020.3010521.
44. Wang K, Hu Z, Ai Q, Zhong Y, Yu J, Zhou P, et al. Joint offloading and charge cost minimization in mobile edge computing. *IEEE Open J Commun Soc.* 2020;1:205–16. doi:10.1109/ojcoms.2020.2971647.
45. Borusu VCLN, Agarwala M, Karamchandani N, Moharir S. Online partial service hosting at the edge. *ACM Trans Model Perform Eval Comput.* 2023;9(1):1–31. doi:10.1145/3616866.
46. Gustavsson U, Frenger P, Fager C, Eriksson T, Zirath H, Dielacher F, et al. Implementation challenges and opportunities in beyond-5G and 6G communication. *IEEE J Microwaves.* 2021;1(1):86–100. doi:10.1109/jmw.2020.3034648.
47. Lv Z, Qiao L, Verma S. AI-enabled IoT-edge data analytics for connected living. *ACM Trans Internet Technol.* 2021;21(4):1–20. doi:10.1145/3421510.
48. Peng X, Song S, Zhang X, Dong M, Ota K. Task offloading for IoAV under extreme weather conditions using dynamic price driven double broad reinforcement learning. *IEEE Internet Things J.* 2024;11(10):17021–33. doi:10.1109/jiot.2024.3360110.
49. Yamanaka H, Kawai E, Teranishi Y, Harai H. Proximity-aware IaaS in an edge computing environment with user dynamics. *IEEE Trans Netw Serv Manag.* 2019;16(3):1282–96. doi:10.1109/tnsm.2019.2929576.
50. Sun G, Wang Z, Su H, Yu H, Lei B, Guizani M. Profit maximization of independent task offloading in MEC-enabled 5G internet of vehicles. *IEEE Trans Intell Transp Syst.* 2024;25(11):16449–61. doi:10.1109/tits.2024.3416300.
51. Forbes: Cloud computing forecast. 2017 [cited 2025 Aug 4]. Available from: <https://www.forbes.com/sites/louisacolumbus/2017/04/29/roundup-of-cloud-computing-forecasts2017/#5c42322c31e8/>.
52. Annual forecast of data-sphere. 2019 [cited 2025 Aug 4]. Available from: www.forbes.com/sites/tomcoughlin/2018/11/27/175-zettabytes-by-2025/#59c815325459/.
53. EdgeCloudSim. EdgeCloudSim. 2023 [cited 2025 Aug 4]. Available from: <https://github.com/CagataySonmez/EdgeCloudSim>.
54. Cloudslab. CloudSim. 2023 [cited 2025 Aug 4]. Available from: <https://github.com/Cloudslab/cloudsim>.