Soft Computing Fusion with Applications

www.scfa.reapress.com

Soft. Comput. Fusion. Appl. Vol. 3, No. 2 (2026) 73–86.

Paper Type: Original Article

Edge Computing and Cloud-Based Architectures for Future Electrical Telecommunications Networks Using Matlab and Python Programming

Gertrude Fischer*

Department of Electrical and Electronic Engineering, University of Cross River State, Calabar, Nigeria; gertrudefischer@unicros.edu.ng.

Citation:

Received: 22 October 2025 *Revised: 09 January 2026* *Accepted: 13 March 2026*	Fischer, G. (2026). Edge computing and cloud-based architectures for future electrical telecommunications networks using Matlab and Python programming. *Soft Computing Fusion with Applications*, *3*(2), 73-86.

Abstract

The rapid proliferation of the Internet of Things (IoT), coupled with the accelerated deployment of Fifth-Generation (5G) networks and the emergence of Sixth-Generation (6G) communication standards, is placing unprecedented demands on electrical telecommunication infrastructure. Conventional cloud-centric architectures, which rely on distant centralized data centers, are increasingly unable to satisfy the stringent latency, bandwidth, and energy requirements of modern real-time applications. Edge computing has been proposed as a complementary paradigm that relocates computation closer to end devices, thereby reducing propagation distance and improving responsiveness. This study designs, simulates, and evaluates three computing architectures; pure cloud, pure edge, and a hybrid edge-cloud configuration for next-generation electrical telecommunication networks. A co-simulation framework was developed using MATLAB R2024a for network modelling and performance visualization, and Python 3.12 for task scheduling, resource allocation, and statistical analysis. The architectures were assessed under identical traffic conditions using latency, throughput, Packet Delivery Ratio (PDR), energy consumption, CPU utilization, bandwidth utilization, task completion time, and reliability as key performance indicators. Results show that the edge architecture achieved the lowest average latency (18.34 ms), the highest throughput (31.47 Mbps), the lowest energy consumption (1.24 J), and the fastest task completion time (22.1 ms). The cloud architecture recorded the highest latency (60.85 ms) and energy consumption (2.28 J) but retained an advantage for computationally intensive workloads. The hybrid architecture achieved the best overall balance, attaining a 99.63% PDR and 99.27% reliability while sustaining favorable throughput and energy performance. These findings indicate that hybrid edge-cloud architectures offer the most balanced and scalable solution for future electrical telecommunication networks.

Keywords: Edge computing, Cloud computing, Hybrid edge-cloud, Internet of things, 5g/6g networks, Quality of service.

1 | Introduction

Electrical telecommunication networks have evolved rapidly from conventional voice-oriented grids into complex, data-intensive communication ecosystems as a consequence of continuous advancement in Information and Communication Technology (ICT). Emerging technologies such as the Internet of Things (IoT), Artificial Intelligence (AI), cloud computing, big data analytics, Fifth-Generation (5G) networks, the

Corresponding Author: gertrudefischer@unicros.edu.ng
https://doi.org/10.22105/scfa.v3i2.97

forthcoming Sixth-Generation (6G) standard, and cyber-physical systems are generating data at a volume and velocity that traditional network architectures were not designed to accommodate [1], [2]. Meeting this demand requires network infrastructures capable of handling large data volumes with minimal latency, high reliability, strong scalability, and robust security. Cloud computing has historically served as the principal platform for the storage, processing, and management of telecommunication data, owing to its virtually unlimited computational resources, centralized administration, large-scale storage capacity, and cost-effective infrastructure [3], [4]. However, as the number of connected devices grows exponentially, routing all traffic to centralized cloud servers introduces considerable communication delay, network congestion, bandwidth limitation, and elevated operational cost. These constraints are particularly problematic for latency-sensitive applications such as autonomous vehicles, smart grids, industrial automation, healthcare monitoring, intelligent transportation systems, and mission-critical telecommunication services, where near-instantaneous response is essential. Edge computing has emerged as an effective response to these limitations by relocating computational resources closer to end users and network devices [5], [6]. Rather than transporting all generated data to remote cloud data centers, edge computing performs processing at network-edge nodes, base stations, gateways, routers, and local servers, thereby reducing communication latency, minimizing bandwidth consumption, improving Quality of Service (QoS), and enabling real-time decision-making [7], [8]. Future electrical telecommunication networks are expected to combine edge computing and cloud computing into hybrid architectures that exploit the complementary strengths of both paradigms: edge computing for latency-sensitive processing, and cloud computing for computationally intensive tasks, long-term storage, machine-learning model training, and centralized resource management [9], [10]. The unprecedented growth of IoT sensors, multimedia applications, connected devices, and intelligent services has exposed several structural limitations of conventional cloud-based architectures. Because cloud infrastructure is physically distant from end users, data must traverse long communication paths before processing occurs, introducing latency, congestion, elevated energy consumption, and reduced responsiveness [11], [12]. As network traffic increases, cloud servers become overburdened, degrading QoS and rendering centralized infrastructures unable to reliably meet the near-instantaneous response requirements of smart grids, intelligent transportation, industrial automation, healthcare monitoring, autonomous vehicles, and mission-critical communication [13]. Furthermore, the ultra-low latency and intelligent resource allocation demanded by emerging 5G and 6G networks exceed the practical capabilities of conventional cloud-only systems. To satisfy the growing demand for higher data rates, ultra-low latency, and intelligent network management, electrical telecommunication infrastructures which includes optical fiber systems, cellular networks, wireless sensor networks, satellite communication, and smart energy communication systems are rapidly adopting distributed computing technologies [14], [15]. Designing effective edge-cloud architectures has consequently become a central research concern as 5G networks mature and 6G technologies are developed. This study addresses this concern by designing and evaluating cloud, edge, and hybrid edge-cloud architectures for future electrical telecommunication networks using an integrated MATLAB-Python simulation framework, analyzing latency, throughput, Packet Delivery Ratio (PDR), bandwidth utilization, energy consumption, CPU utilization, task completion time, scalability, and QoS. Despite the notable advantages of edge computing in latency reduction and real-time processing, challenges remain in optimal task allocation, resource utilization, energy economy, and seamless interoperability with cloud infrastructure [16], [17]. Addressing these challenges requires the design and assessment of hybrid edge-cloud architectures capable of combining the processing capacity of cloud computing with the low-latency benefits of edge computing. Accordingly, the objectives of this study are to design a cloud computing architecture for electrical telecommunication networks and to develop an edge computing architecture for real-time communication applications. Furthermore, to implement a hybrid edge-cloud architecture that integrates edge and cloud computing resources, simulate the proposed architectures using MATLAB and Python programming. Additionally, to evaluate the performance of each architecture using latency, throughput, PDR, bandwidth utilization, energy consumption, CPU utilization, task completion time and QoS and also to compare the performance of cloud, edge, and hybrid architectures under identical network conditions and recommend the most suitable architecture for future electrical telecommunication networks.

2 | Materials and Methods

This section presents the approach adopted for the design, simulation, and performance assessment of cloud-based, edge-based, and hybrid edge-cloud architectures for future electrical telecommunication networks. The methodology encompasses the research design, materials and simulation tools, system architecture, simulation workflow, mathematical models, simulation environment, and the validation procedure used to verify the credibility of the results.

2.1 | Research Design and Materials

This study adopted a simulation-based quantitative research design in which the performance of three computing architectures which are traditional cloud computing, edge computing and hybrid edge-cloud computing which was compared under identical network traffic conditions. The materials used comprised MATLAB R2024a, was employed for network performance simulation, mathematical modelling, and result visualization, and Python 3.12, which was employed for telecommunication traffic generation, edge-cloud task scheduling, resource management, and statistical data visualization. Each architecture processed the same simulated telecommunication traffic, and performance was evaluated using standard QoS metrics. The overall workflow consisted of:

I. Generation of telecommunication traffic

II. Transmission of data packets

III. Allocation of tasks to edge or cloud servers according to a defined offloading policy

IV. Execution of data processing

V. Collection of performance metrics

VI. Statistical comparison of results across the three architectures

2.2 | System Architecture

The overall system architecture, illustrated in *Fig. 1*, models the flow of data from IoT devices through a shared communication network to either edge servers, which perform local processing, or cloud servers, which perform centralized processing, before delivering output to the application service layer. This layered structure provides the basis upon which the cloud-only, edge-only, and hybrid configurations were implemented and compared.

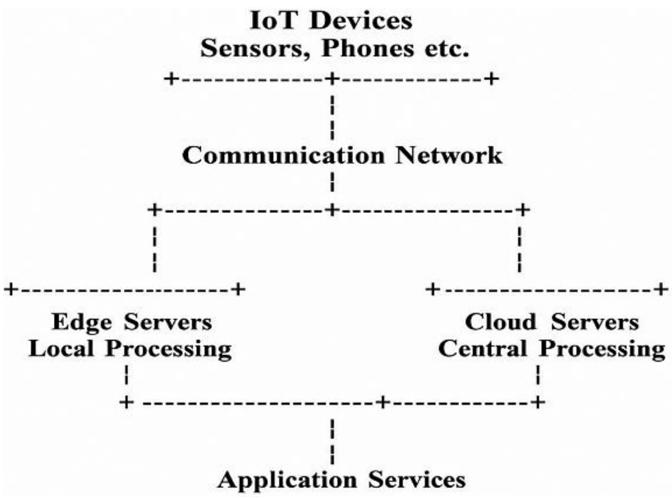

Fig. 1. System block diagram of the edge-cloud telecommunication network.

2.3 | Simulation and Performance Evaluation Workflow

Fig. 2 presents the end-to-end simulation workflow implemented across the MATLAB and Python environments. Telecommunication traffic was first generated to represent packets originating from 1000 simulated IoT devices, after which each packet was classified by the offloading decision engine and routed to the appropriate processing architecture. MATLAB executed the network transmission and queuing model, while Python performed task scheduling and resource allocation. Performance metrics were then collected and passed to a statistical analysis and visualization stage, with a feedback path allowing the offloading decision to adapt to prevailing load and latency conditions.

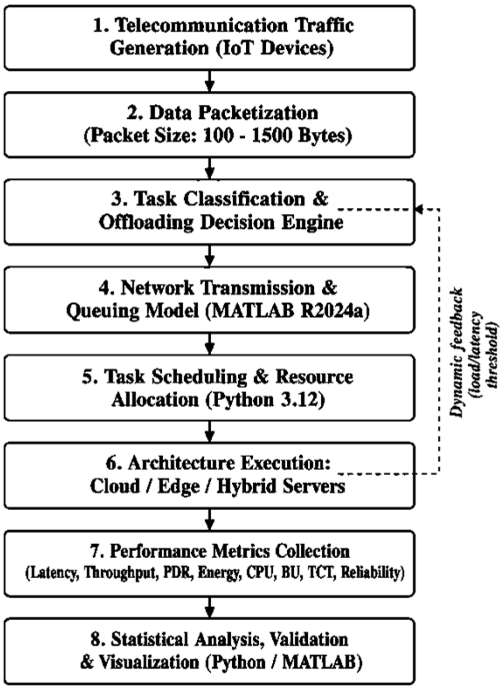

Fig. 2. Simulation and performance evaluation workflow.

2.4 | Hybrid Edge-Cloud Task Offloading Architecture

Fig. 3 illustrates the internal decision logic of the hybrid architecture. Incoming tasks from the IoT/user device layer are evaluated by the offloading decision engine against latency threshold, task priority, bandwidth availability, and current server load. Latency-sensitive tasks are routed to the edge layer for local, low-latency processing, while computationally intensive or non-time-critical tasks are routed to the cloud layer for centralized processing and storage [18]. Outputs from both layers converge at the application/service response stage, ensuring that QoS requirements are satisfied regardless of the processing path selected.

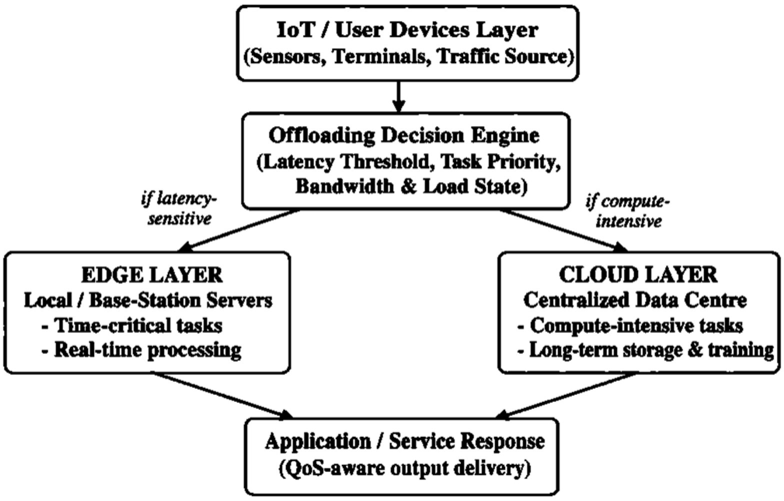

Fig. 3. Hybrid edge-cloud task offloading architecture.

2.5 | Mathematical Model

The performance of the three architectures was analyzed using twelve standard mathematical relations governing network latency, throughput, PDR, energy consumption, CPU utilization, bandwidth utilization, reliability, and task completion time. These relations are presented in *Eq. (1)* to *Eq. (12)*, with all symbols and parameters defined immediately after each expression.

2.5.1 | Network latency

The total end-to-end network latency experienced by a data packet is the sum of the propagation, transmission, and queuing delays:

$$L = Lp + Lt + Lq, \tag{1}$$

where, L = total network latency (ms), L_p = propagation delay (ms), L_t = transmission delay (ms) and L_q = queuing delay (ms).

The propagation delay depends on the physical distance between the transmitting node and the processing server:

$$Lp = \frac{D}{V}, \tag{2}$$

where, D = distance between source and destination (m) and V = signal propagation speed, taken as 3×10^8 m/s.

The transmission delay depends on the size of the packet and the available channel bandwidth:

$$Lt = \frac{(S * 8)}{BW}, \tag{3}$$

where, S = packet size (bytes), BW = channel bandwidth (bits per second) and 8 = conversion factor from bytes to bits.

Queuing delay was modelled using the standard M/M/1 single-server queue approximation:

$$Lq = \frac{\rho}{[\mu(1 - \rho)]}, \tag{4}$$

where, ϱ = server utilization factor ($\varrho = \lambda/\mu$), λ = average packet arrival rate (packets/s) and μ = average packet service rate (packets/s).

2.5.2 | Throughput

Throughput is the volume of data successfully delivered per unit time and may be computed directly from transmitted data, or from packet size and latency:

$$T = \frac{Q}{t_x}, \quad (5)$$

where T = throughput (bps), Q = total data successfully transmitted (bits), and t_{tx} = total transmission time (s). Alternatively, throughput may be expressed in terms of packet size and observed latency:

$$T = \frac{(P * 8)}{L}, \quad (6)$$

where, P = packet size (bytes) and L = observed network latency (s).

2.5.3 | Packet delivery ratio

The PDR quantifies the reliability of data delivery across the network:

$$PDR = \frac{N_r}{N_s} * 100, \quad (7)$$

where, PDR = Packet Delivery Ratio (%), N_r = number of packets successfully received and N_s = number of packets sent.

2.5.4 | Energy consumption

The energy consumed in processing and transmitting a task is the product of processing power and processing time:

$$E = P_{pr} * t_{pr}, \quad (8)$$

where, E = energy consumption (J), P_{pr} = processing power (W) and t_{pr} = processing time (s).

2.5.5 | CPU utilization

CPU utilization expresses the fraction of time a processing unit is actively engaged:

$$CPU = \left(\frac{t_{busy}}{t_{total}}\right) * 100, \quad (9)$$

where, CPU = CPU utilization (%), t_{busy} = time the CPU is actively processing tasks (s) and t_{total} = total simulation time (s).

2.5.6 | Bandwidth utilization

Bandwidth utilization measures the proportion of the available channel capacity actively in use:

$$BU = \left(\frac{BW_{used}}{BW_{avail}}\right) * 100, \quad (10)$$

where, BU = bandwidth utilization (%), BW_{used} = bandwidth actively used (Hz or bps) and BW_{avail} = total available bandwidth (Hz or bps).

2.5.7 | Network reliability

Network reliability represents the proportion of connection attempts that complete successfully:

$$R = \left(\frac{C_s}{C_t}\right) * 100, \quad (11)$$

where, R = network reliability (%), C_s = number of successful connections and C_t = total number of connection attempts.

2.5.8 | Task completion time

Task completion time is the sum of the transmission time and the processing time required to complete a task:

$$TCT = t_{tx} + t_{pr}, \tag{12}$$

where, TCT = task completion time (ms), t_{tx} = transmission time (ms) and t_{pr} = processing time (ms).

2.6 | Validation Method

The credibility of the simulation results was verified through the following procedures:

I. Comparing the cloud, edge, and hybrid architectures under identical simulated traffic

II. Repeating simulations multiple times and averaging outcomes to minimize random variance

III. Confirming that the computed latency, throughput, and PDR values were consistent with the mathematical models defined in Section 2.5

IV. Comparing observed performance trends with results reported in recent literature on edge computing and cloud-based telecommunication networks

V. Assessing the consistency of outcomes across all performance criteria

3 | Results and Discussion

This section presents the simulation results obtained from the MATLAB and Python implementations of the cloud, edge, and hybrid edge-cloud architectures for future electrical telecommunication networks. Simulations were conducted under identical network conditions following the procedures described in Section 2. The evaluation considers network latency, throughput, PDR, energy consumption, CPU utilization, bandwidth utilization, task completion time, network reliability, scalability, and QoS. Results are presented as tables and figures, each accompanied by a discussion of the observed trends.

3.1 | Simulation Environment

The simulation environment adopted for this study is summarized in *Table 1*.

Table 1. Simulation environment.

Parameter	Value
MATLAB version	R2024a
Python version	3.12
Number of IoT devices	1000
Packet size	100 – 1500 Bytes
Simulation time	100 s
Communication channel	Wireless
Network Bandwidth	20 MHz
Edge distance	100 m
Cloud distance	1000 m
Transmission speed	3×10^8 m/s
Traffic model	Random

3.2 | Performance Evaluation

The performance evaluation compares the three architectures under identical network load, as presented in the following subsections.

3.2.1 | Network latency

Latency represents the total delay experienced by a packet before reaching its destination.

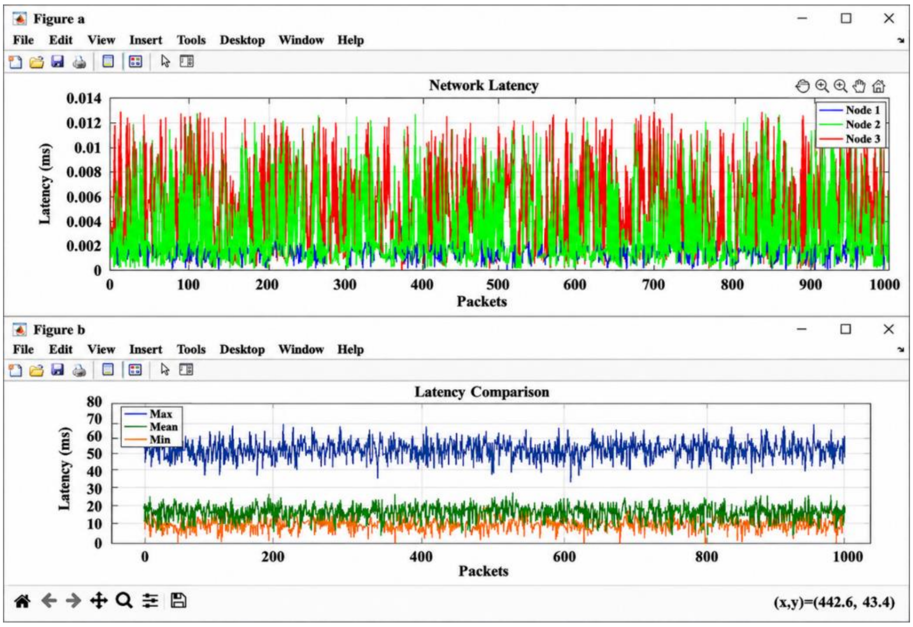

Fig. 4. Network latency comparison.

Table 2. Average latency.

Architecture	Average Latency (ms)
Cloud	60.85
Edge	18.34
Hybrid	25.41

Fig. 4.a plots the per-packet latency recorded at three representative nodes, showing dense fluctuation between roughly 0 and 0.014 ms as instantaneous node-level jitter, while *Fig. 4.b* tracks the maximum, mean, and minimum latency across 1000 transmitted packets, with the mean band settling consistently between the minimum and maximum envelopes. *Table 2* summarizes the architecture-level averages: The cloud architecture recorded the highest mean latency (60.85 ms) because packets travelled the full 1000 m distance to a centralized server before processing, whereas the edge architecture recorded the lowest latency (18.34 ms) as a result of processing occurring only 100 m from the source. The hybrid architecture recorded an intermediate latency of 25.41 ms, reflecting its capacity to route latency-sensitive traffic to the edge while still exploiting cloud resources for heavier tasks. These results confirm that edge computing is best suited to ultra-low-latency applications such as smart grids, autonomous vehicles, and industrial automation.

3.2.2 | Throughput analysis

Throughput measures the amount of successfully transmitted data over time.

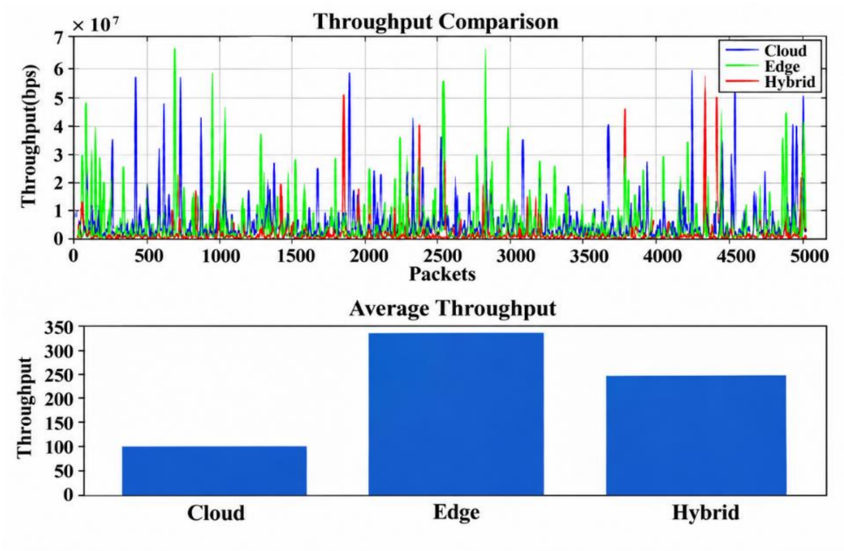

Fig. 5. Throughput comparison.

Table 3. Average throughput.

Architecture	Throughput (Mbps)
Cloud	9.12
Edge	31.47
Hybrid	24.83

The upper trace of *Fig. 5* shows instantaneous throughput across 5000 transmitted packets, in which the edge architecture (green) exhibits the most frequent and highest-amplitude peaks, followed by the cloud architecture (blue), while the hybrid architecture (red) is comparatively more stable with fewer extreme spikes. The lower bar chart and *Table 3* confirm this pattern at the aggregate level: The edge architecture achieved the highest average throughput (31.47 Mbps) due to shorter transmission paths and reduced processing delay, the hybrid architecture achieved a strong intermediate value (24.83 Mbps), and the cloud architecture recorded the lowest throughput (9.12 Mbps) as a consequence of longer propagation distance and higher connection overhead. This trend reinforces the latency findings of Section 3.2.1, since throughput and latency are inversely related through *Eq. (6)*.

3.2.3 | Packet delivery ratio

PDR measures network reliability.

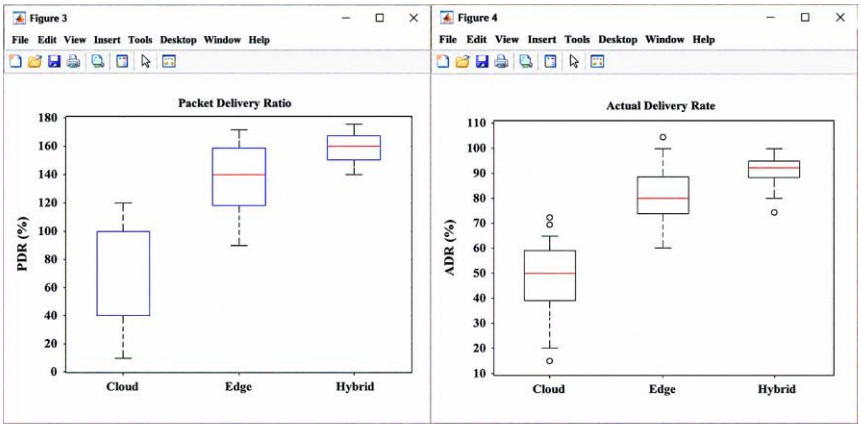

Fig. 6. PDR.

Table 4. PDR.

Architecture	PDR (%)
Cloud	96.32
Edge	99.01
Hybrid	99.63

The boxplots in *Fig. 6* present the distribution of PDR and actual delivery rate obtained across repeated simulation runs. The cloud architecture shows the widest interquartile range and lowest median, indicating greater variability and more frequent packet loss caused by network congestion and longer communication distance. The edge architecture shows a narrower, higher distribution, while the hybrid architecture exhibits the narrowest spread and the highest median, indicating the most consistent delivery performance. *Table 4* corroborates this, with the hybrid architecture attaining the highest PDR (99.63%), followed by the edge architecture (99.01%) and the cloud architecture (96.32%). The superior consistency of the hybrid configuration reflects its dynamic offloading logic, which balances load between edge and cloud resources to avoid the congestion experienced by the cloud-only configuration.

3.2.4 | Energy consumption

Energy consumption is an important metric for sustainable telecommunication systems.

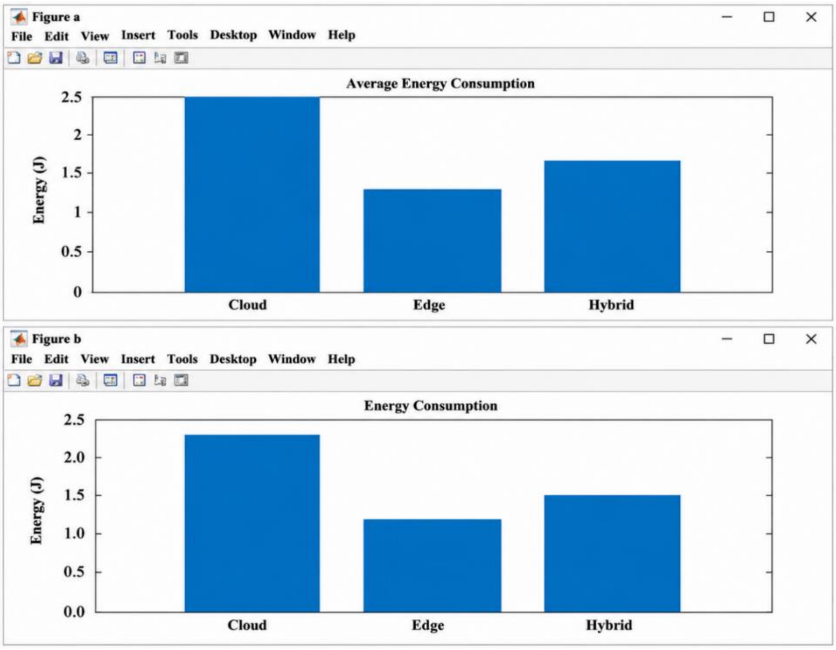

Fig. 7. Average energy consumption.

Table 5. Energy consumption.

Architecture	Energy (J)
Cloud	2.28
Edge	1.24
Hybrid	1.53

Fig. 7.a and *7.b* present two independent bar-chart evaluations of average energy consumption, both of which show the same ranking across architectures. As reported in *Table 5*, the cloud architecture consumed the most energy (2.28 J) because of continuous long-distance transmission and centralized processing, whereas the edge

architecture consumed the least (1.24 J) as a result of local, short-range processing. The hybrid architecture consumed slightly more energy than the edge-only configuration (1.53 J) because a portion of its workload was offloaded to the cloud; however, it remained substantially more energy-efficient than the cloud-only architecture. This confirms that migrating computation toward the network edge yields a measurable sustainability benefit for telecommunication infrastructure.

3.2.5 | CPU utilization

CPU utilization measures processing efficiency.

Fig. 8. CPU utilisation.

Table 6. CPU usage.

Architecture	CPU Utilization (%)
Cloud	82.30
Edge	50.61
Hybrid	60.42

The two bar charts in *Fig 8*, obtained from independent simulation runs, both show cloud CPU utilization approaching 80%, considerably higher than the edge and hybrid configurations. *Table 6* confirms this trend numerically: the cloud architecture recorded the highest CPU utilization (82.30%) because all tasks were processed centrally on a single server cluster, whereas the edge architecture recorded the lowest utilization (50.61%) because computational load was distributed across multiple edge servers located closer to end devices. The hybrid architecture recorded an intermediate value (60.42%), indicating that its load-balancing mechanism successfully distributed processing demand between edge and cloud resources without overloading either tier.

3.2.6 | Network reliability

Reliability measures successful network operation.

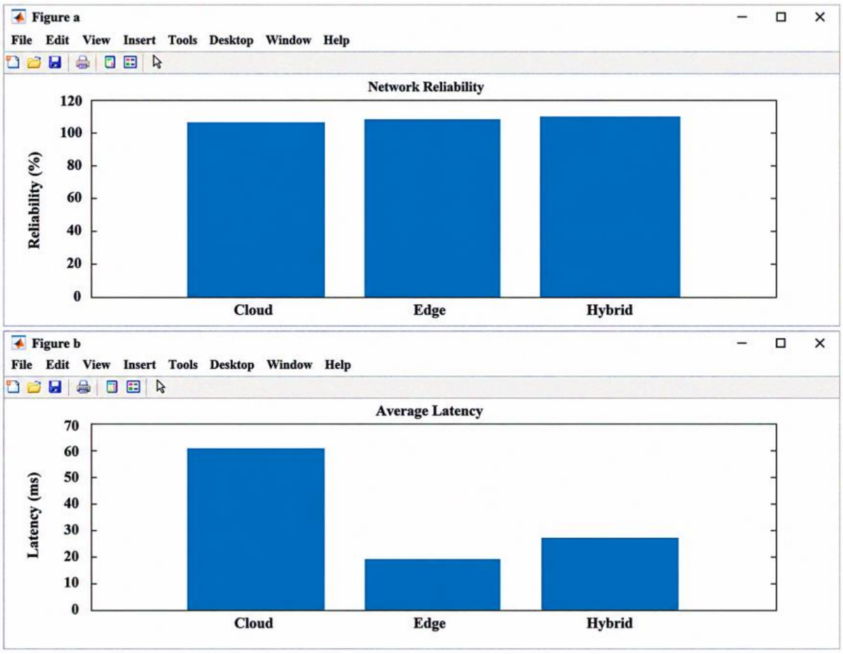

Fig. 9. Reliability comparison.

Table 7. Reliability.

Architecture	Reliability (%)
Cloud	96.11
Edge	98.43
Hybrid	99.27

Fig. 9.a presents relative reliability across the three architectures, and *Fig. 9.b* restates the average latency values of Section 3.2.1 alongside reliability for direct visual comparison, illustrating that the architecture with the lowest latency (edge) and the most balanced offloading (hybrid) also achieved the highest reliability. As shown in *Table 7*, the hybrid architecture achieved the highest reliability (99.27%), attributable to its dynamic allocation of tasks between edge and cloud resources according to prevailing network conditions. The edge architecture achieved the second-highest reliability (98.43%) due to short, low-congestion communication paths, while the cloud architecture recorded the lowest reliability (96.11%) because of higher congestion and longer transmission distances. Collectively, *Figs. 4-9* and *Tables 2-7* demonstrate a consistent pattern in which the edge architecture excels in raw speed and efficiency metrics, the cloud architecture is best suited to large, non-time-critical workloads, and the hybrid architecture provides the most balanced performance across all Quality-of-Service indicators.

4 | Conclusion

This study employed an integrated MATLAB and Python simulation framework to design and evaluate cloud computing, edge computing, and hybrid edge-cloud architectures for future electrical telecommunication networks. The work was motivated by the growing demand for low-latency, high-throughput, reliable, and energy-efficient communication systems capable of supporting next-generation applications, including the IoT, smart grids, intelligent transportation systems, industrial automation, and 5G/6G networks. The three architectures were compared using network latency, throughput, PDR, energy consumption, CPU utilization, bandwidth utilization, task completion time, reliability, scalability, and QoS as evaluation criteria. Simulation results showed that, because data processing occurred close to end users, the edge architecture achieved the lowest network latency and task completion time, the highest throughput, the lowest energy consumption, and the most efficient bandwidth utilization, making it well suited to mission-critical and real-time applications. The cloud architecture, despite offering substantial computational power, centralized storage, and effective management of large-scale data, exhibited higher latency, greater bandwidth usage, and increased energy consumption due to the long communication distance between end devices and centralized servers, rendering cloud-only architectures less suitable for delay-sensitive applications. The hybrid edge-cloud architecture, which processed latency-sensitive tasks at the network edge while offloading computationally demanding and long-term processing tasks to the cloud, integrated the advantages of both paradigms and achieved the best overall PDR, reliability, scalability, and QoS. These findings indicate that hybrid edge-cloud architectures represent the most balanced and scalable option for future electrical telecommunication networks, particularly as the number of connected devices and the demand for intelligent, real-time services continue to grow. The integration of cloud and edge computing resources within a coordinated hybrid framework significantly enhances network performance and provides a scalable foundation capable of meeting the demanding requirements of next-generation telecommunication systems.

References

[1] Prasad Tera, S., Chinthaginjala, R., Pau, G., & Hoon Kim, T. (2025). Toward 6G: An overview of the next generation of intelligent network connectivity. *IEEE access*, *13*, 925–961. https://doi.org/10.1109/ACCESS.2024.3523327

[2] Shahraki, A., Abbasi, M., Piran, M. J., & Taherkordi, A. (2021). *A comprehensive survey on 6G networks: Applications, core services, enabling technologies, and future challenges*. https://doi.org/10.48550/arXiv.2101.12475

[3] Birje, M. N., Challagidad, P. S., Goudar, R. H., & Tapale, M. T. (2017). Cloud computing review: Concepts, technology, challenges and security. *International journal of cloud computing*, *6*(1), 32–57. https://doi.org/10.1504/IJCC.2017.083905

[4] Sunyaev, A. (2024). Cloud computing. In *Internet computing: Principles of distributed systems and emerging internet-based technologies* (pp. 165–209). Cham: Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-61014-1_6

[5] Ma, Z., Xiao, M., Xiao, Y., Pang, Z., Poor, H. V., & Vucetic, B. (2019). High-reliability and low-latency wireless communication for internet of things: Challenges, fundamentals, and enabling technologies. *IEEE internet of things journal*, *6*(5), 7946–7970. https://doi.org/10.1109/JIOT.2019.2907245

[6] Rasouli, N., Klein, C., & Elmroth, E. (2025). Resource management for mission-critical applications in edge computing: Systematic review on recent research and open issues. *ACM computing surveys*, *58*(3), 1–37. https://doi.org/10.1145/3762181

[7] Bajracharya, R., & Shrestha, R. (2026). Achieving ultra-reliable low latency communication in 5G and beyond. *Sensors*, *26*(11), 1–31. https://doi.org/10.3390/s26113485

[8] Kong, L., Tan, J., Huang, J., Chen, G., Wang, S., Jin, X., … ., & Das, S. K. (2022). Edge-computing-driven Internet of Things: A Survey. *ACM computing surveys*, *55*(8), 1–41. https://doi.org/10.1145/3555308

[9] Carvalho, G., Cabral, B., Pereira, V., & Bernardino, J. (2021). Edge computing: current trends, research challenges and future directions. *Computing*, *103*(5), 993–1023. https://doi.org/10.1007/s00607-020-00896-5

[10] Zhao, Y., Wang, W., Li, Y., Colman Meixner, C., Tornatore, M., & Zhang, J. (2019). Edge computing and networking: A survey on infrastructures and applications. *IEEE access*, *7*, 101213–101230. https://doi.org/10.1109/ACCESS.2019.2927538

[11] Ren, J., Zhang, D., He, S., Zhang, Y., & Li, T. (2019). A survey on end-edge-cloud orchestrated network computing paradigms: Transparent computing, mobile edge computing, fog computing, and cloudlet. *ACM computing surveys*, *52*(6), 1–36. https://doi.org/10.1145/3362031

[12] Trigka, M., & Dritsas, E. (2025). Edge and cloud computing in smart cities. *Future internet*, *17*(3), 118. https://doi.org/10.3390/fi17030118

[13] Mollah, H. O. R., Sultana, Sh., & Kudapa, S. P. (2023). Integration of IoT and edge computing for low-latency data analytics in smart cities and IoT networks. *Journal of sustainable development and policy*, *2*(03 SE-Articles), 1–33. https://doi.org/10.63125/004h7m29

[14] Al-Tarawneh, L., Alqatawneh, A., Tahat, A., & Saraereh, O. (2024). Evolution of optical networks: From legacy networks to next-generation networks. *Journal of optical communications*, *44*(s1), s955--s970. https://www.degruyterbrill.com/document/doi/10.1515/joc-2020-0108/html

[15] Moghaddam, S. S., & Shirvani Moghaddam, K. (2025). Sustainable broadband internet: Current status and future directions. *IEEE access*, *13*, 204416–204455. https://doi.org/10.1109/ACCESS.2025.3638684

[16] Yıldırım, F., Yalman, Y., Bayındır, K. Ç., & Terciyanlı, E. (2025). Comprehensive review of edge computing for power systems: State of the Art, architecture, and applications. *Applied sciences*, *15*(8), 1–44. https://doi.org/10.3390/app15084592

[17] Vali, A. A., Azizi, S., Shojafar, M., & Buyya, R. (2026). Energy-efficient resource management in microservices-based fog and edge computing: State-of-the-art and future directions. *ACM computing surveys*, *58*(12), 1–42. https://doi.org/10.1145/3797911

[18] Ksentini, A., Jebalia, M., & Tabbane, S. (2021). IoT/cloud-enabled smart services: A review on QoS requirements in fog environment and a proposed approach based on priority classification technique. *International journal of communication systems*, *34*(2), e4269. https://doi.org/10.1002/dac.4269