Advancements and Challenges in Cloud Computing, Edge Computing and Edge-Cloud Computing for IoT: A Comprehensive Review

Ahmed Er-Rahmani[1](✉), Mohamed El Ghmary[2], Wijdane Lakhouari[1], and Hassan Echoukairi[1]

[1] Faculty of Sciences, Department of Computer Science, Mohammed V University, Rabat, Morocco
ahmed_errahmani@um5.ac.ma, wijdane.lakhouari@um5r.ac.ma, hassan.echoukairi@fsr.um5.ac.ma

[2] Department of Mathematics and Computer Science, Sidi Mohamed Ben Abdellah University, FSDM, Fes, Morocco
mohamed.elghmary@usmba.ac.ma

Abstract. Internet of Things (IoT) advancements drive cloud computing (CC), edge computing (EC), and hybrid edge-cloud computing (ECC) paradigms to address challenges like resource allocation, latency, and scalability. ECC dynamically allocates tasks between cloud and edge, optimizing workload demands. This study compares CC, EC, and ECC across architectural frameworks, latency, scalability, security, and energy efficiency. Results emphasize adaptable edge-cloud systems requiring AI [1,2] and blockchain integration to bolster security, interoperability, and efficiency in evolving IoT ecosystems.

Keywords: Internet of Things · Cloud computing · Edge computing · Hybrid architectures · Scalability · Security · Cost-efficiency · Energy consumption

1 Introduction

The Internet of Things (IoT) has evolved significantly, enabling seamless device integration across various applications. To address growing computational demands and real-time responsiveness, cloud computing (CC), edge computing (EC), and edge-cloud computing (ECC) have emerged as key paradigms for IoT development and resource management.

CC [3] provides scalable, on-demand access to shared computing resources via the internet, enabling IoT applications by offloading tasks from constrained devices to the cloud. Despite benefits like resource pooling, challenges such as latency and bandwidth demands persist, driving research into energy-efficient resource allocation to balance performance and power consumption.

Y. Mejdoub et al. (Eds.): COCIA 2025, LNNS 1584, pp. 165–170, 2026.
https://doi.org/10.1007/978-3-032-01536-5_25

In order to address these challenges, EC [4] processes data near the source, minimizing latency and bandwidth use while boosting real-time performance. Its hierarchical architecture combining edge devices, local servers, and cloud services enhances IoT reliability and privacy through localized data handling. Challenges like scalability constraints and adaptive workload distribution drive efforts to optimize resource utilization.

The ECC merges centralized cloud resources with decentralized edge processing, balancing computation to optimize performance and resource utilization. The hybrid model supports dynamic task offloading and seamless coordination between cloud servers and edge nodes.

This study provides a comparative analysis of CC, EC, and ECC by examining key factors such as architecture, latency, scalability, security, and cost-efficiency. The analysis aims to guide stakeholders in selecting the most suitable model for IoT applications. The paper is structured to review related work, discuss the pros and cons of each paradigm, and provide a comprehensive comparative analysis.

This paper is structured as follows: Sect. 2 reviews related work on CC, EC, and ECC. Section 3 discusses the advantages and disadvantages of these paradigms, and presents a comparative analysis based on key factors such as performance, scalability, and security. Section 4 provides a discussion on challenges and future opportunities. Finally, Sect. 5 concludes the paper with key findings and future directions.

2 Literature Review

IoT paradigms have advanced significantly in recent years to address the growing need for computational power and real-time responsiveness in IoT applications. Notable among these paradigms are CC, EC, and ECC.

Research has demonstrated that CC effectively offloads computation-intensive tasks, reducing energy consumption and prolonging IoT device lifespan. Rajagopalan et al. [5] emphasized CC's ability to handle large-scale data processing, while Aslanpour et al. [6] found that CC provides lower response times for data-intensive applications.

EC has gained attention for reducing latency and improving privacy by processing data closer to IoT devices. Kong et al. [7] highlighted EC's advantages in enabling real-time applications and minimizing network congestion. Additionally, Mansouri et al. [8] showed that EC outperforms CC in latency-sensitive scenarios, offering greater energy efficiency. Nevertheless, EC faces scalability challenges due to the limited capabilities of edge devices.

In comparative studies on ECC, this paradigm integrates the advantages of both CC and EC by combining cloud resources with edge processing capabilities. Although direct comparisons between ECC and other paradigms are relatively limited, existing research has examined its potential benefits. Yuan et al. [9] introduced a resource allocation algorithm for ECC, emphasizing the advantages of dynamic task offloading and workload balancing between cloud and edge resources. Additionally, Andriulo et al. [10] performed a comparative analysis of ECC and CC, demonstrating that ECC achieves lower latency and enhanced resource utilization by effectively leveraging both cloud and edge infrastructures.

Overall, comparative analyses of CC, EC, and ECC provide valuable insights into their performance trade-offs, helping stakeholders choose the most suitable solution for specific IoT applications.

3 Comparative Analysis of Cloud Computing, Edge Computing, and Edge-Cloud Computing

Cloud models (CC [3], EC [4], ECC) address IoT challenges like latency and resource constraints, balancing benefits and limitations. This section assesses their contributions and drawbacks for IoT systems.

Table 1 compares CC, EC, and ECC. CC centralizes processing, offering scalability but higher latency. EC decentralizes computation, enhancing privacy and reducing latency, yet faces scalability issues. ECC integrates both, balancing efficiency through dynamic cloud-edge resource allocation. The comparison highlights trade-offs across IoT applications.

The analysis of advantages and disadvantages in this section provides a foundation for understanding the strengths and limitations of each paradigm. Building on this, the next section presents a detailed comparative analysis, offering insights into their performance across various criteria.

Emerging developments in IoT have facilitated notable improvements, offering diverse applications despite challenges such as constrained resources, latency, and real-time demands. To mitigate these issues, approaches like cloud computing (CC), edge computing (EC), and edge-cloud collaboration (ECC) have been introduced. This section compares their architectures, benefits, and limitations to guide optimal selection in mobile application scenarios.

CC in IoT uses remote servers for processing/storage, offering scalable data management. Centralization causes latency and bandwidth strain from distant transfers. CC struggles with real-time processing in time-sensitive tasks, requiring significant energy for infrastructure upkeep. Third-party data storage raises security risks, yet CC remains cost-efficient for large IoT deployments via scalable resources.

CC is ideal for IoT use cases that require extensive data storage and processing power, such as smart agriculture, industrial IoT (IIoT) for predictive maintenance, and healthcare systems managing large-scale patient data in centralized repositories.

EC decentralizes resources to the edge, enabling local data processing. Proximity to IoT devices reduces latency and bandwidth, improving real-time performance for autonomous systems or automation. While EC lowers energy use via shorter data transfers, scalability suffers due to resource-heavy edge nodes. Security risks persist with decentralized data vs. centralized clouds. Still, EC benefits mobile IoT via low-latency processing.

Table 1. Advantages and disadvantages of CC, EC and ECC [10–13]

Criteria	CC	EC	ECC
Latency	High latency due to data transmission to distant cloud servers, which results in delays in processing and response times	Reduced latency due to local processing of data closer to the IoT devices, supporting time-sensitive applications	Moderate latency by distributing tasks between cloud and edge computing resources, thus balancing low-latency processing with cloud's capabilities
Bandwidth Usage	High bandwidth usage because data needs to be continuously transferred to and from the cloud, consuming significant resources for IoT devices	Reduced bandwidth consumption as processing occurs locally at the edge, minimizing the need for constant data transfers	Optimized bandwidth by processing critical tasks at the edge and transferring less critical data to the cloud, minimizing overall data usage
Scalability	Highly scalable due to the cloud's large resource pool, enabling dynamic resource allocation for varying IoT demands	Scalability is limited by the resources available at the edge, as devices have finite processing power and storage	Scalable by combining the strengths of both cloud and edge computing, dynamically allocating tasks between the two to meet varying IoT needs
Real-time Processing	CC is less suited for real-time processing due to high latency in transmitting data to and from the cloud	Excellent for real-time processing since edge devices handle data locally, providing rapid responses for time-critical applications	Real-time processing is supported by distributing tasks between the cloud and edge, ensuring that time-sensitive tasks are handled at the edge and less critical ones are processed in the cloud
Security & Privacy	Centralized security protocols in the cloud, but increased risks due to data transmission across the network, which may expose data to potential breaches	Enhanced privacy and security as sensitive data is processed locally at the edge, keeping it within the local network and minimizing exposure	Provides a balanced approach to security by processing sensitive data at the edge and leveraging cloud-based security for non-sensitive tasks, ensuring better protection overall
Cost Efficiency	Can become costly due to data transfer and storage fees in the cloud, especially for data-intensive applications	Cost-efficient due to the reduced need for data transfer and reliance on local resources for processing, though the edge devices require upfront investment	Balances cost efficiency by processing data locally at the edge and leveraging cloud resources for heavier computations, reducing the overall cost of operations
Energy Consumption	High energy consumption as cloud data centers require significant power for operations and cooling	Lower energy consumption due to localized processing at the edge, which typically consumes less power compared to cloud data centers	Moderates energy consumption by splitting the load between the cloud and edge nodes, reducing the overall power required by IoT devices
Deployment Complexity	Low complexity as cloud solutions require less infrastructure and are easier to implement, provided there is internet connectivity	Higher complexity due to the distributed nature of edge computing, requiring deployment and management of multiple edge devices and systems	Moderate complexity as it involves managing a hybrid infrastructure, coordinating tasks between cloud and edge resources while ensuring seamless integration

EC is well-suited for IoT applications that demand real-time processing and low latency, such as autonomous vehicles, smart surveillance systems, and industrial automation, where immediate decision-making is critical.

ECC merges CC and EC into a hybrid framework, balancing IoT demands: EC handles low-latency tasks, while CC manages intensive processing. This division optimizes scalability, energy efficiency, and latency via dynamic workload distribution. Sensitive data processed locally at the edge mitigates security risks. However, coordination complexities between cloud and edge hinder deployment. ECC remains versatile for diverse IoT needs despite these challenges.

ECC is particularly beneficial for hybrid applications that require both local responsiveness and centralized processing, such as smart cities, healthcare monitoring systems with AI-based analytics, and connected manufacturing ecosystems.

4 Discussion

CC, EC, ECC balance distinct pros and cons in IoT. CC's scalability suits large-scale analytics, but suffers latency from centralized data centers. EC prioritizes real-time processing and privacy via local computation, yet struggles with scalability.

ECC merges CC and EC, optimizing cost and performance via adaptive workload distribution. However, hybrid systems face coordination and security challenges, demanding robust encryption and access controls.

Future IoT trends may integrate 5G, AI, and energy-efficient devices to enhance cloud-edge ecosystems, prioritizing performance, scalability, and security. The following Fig. 1 shows a graphic representation of the various components involved in the development of the ECC Architecture discussed in this paper.

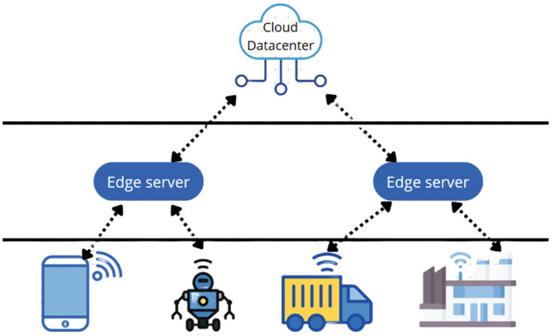

Fig. 1. Cloud-Edge Computing Architecture

5 Conclusion

This paper presented a comparative analysis of CC, EC, and ECC for IoT environments. Each paradigm offers unique advantages: CC excels in scalability but suffers from latency, EC provides low-latency processing but has scalability limitations, and ECC balances both for optimized performance.

The choice of the most suitable paradigm depends on factors such as latency, scalability, and security requirements. While CC is ideal for data-intensive applications, EC is better suited for real-time needs, and ECC offers a hybrid solution for complex IoT scenarios.

References

1. Echoukairi, H., Ghmary, M.E., Ziani, S., Ouacha, A.: Improved methods for automatic facial expression recognition. Int. J. Interact. Mobile Technol. **17**(6), 1–12 (2023)
2. Ouacha, A., El Ghmary, M.: Virtual machine migration in mec based artificial intelligence technique. IAES Int. J. Artif. Intell. **10**(1), 244 (2021)
3. Cloud, H.: The nist definition of cloud computing. Natl. Inst. Sci. Technol. Spec. Publ. **800**(2011), 145 (2011)
4. Shi, W., Cao, J., Zhang, Q., Li, Y., Lanyu, X.: Edge computing: vision and challenges. IEEE Internet Things J. **3**(5), 637–646 (2016)
5. Rajagopalan, A., et al.: Empowering power distribution: unleashing the synergy of IoT and cloud computing for sustainable and efficient energy systems. Res. Eng. **21**, 101949 (2024)
6. Aslanpour, M.S., Gill, S.S., Toosi, A.N.: Performance evaluation metrics for cloud, fog and edge computing: a review, taxonomy, benchmarks and standards for future research. Internet Things **12**, 100273 (2020)
7. Kong, L., et al.: Edge-computing-driven internet of things: a survey. ACM Comput. Surv. **55**(8), 1–41 (2022)
8. Mansouri, Y., Babar, M.A.: A review of edge computing: features and resource virtualization. J. Parallel Distrib. Comput. **150**, 155–183 (2021)
9. Yuan, H., Zhou, M.C.: Profit-maximized collaborative computation offloading and resource allocation in distributed cloud and edge computing systems. IEEE Trans. Autom. Sci. Eng. **18**(3), 1277–1287 (2020)
10. Andriulo, F.C., Fiore, M., Mongiello, M., Traversa, E., Zizzo, V.: Edge computing and cloud computing for internet of things: a review. In: Informatics, vol. 11, p. 71. MDPI (2024)
11. Pal, S., Jhanjhi, N.Z., Abdulbaqi, A.S., Akila, D., Almazroi, A.A., Alsubaei, F.S.: A hybrid edge-cloud system for networking service components optimization using the internet of things. Electronics **12**(3), 649 (2023)
12. Kuchuk, H., Malokhvii, E.: Integration of iot with cloud, fog, and edge computing: a review. Adv. Inf. Syst. **8**(2), 65–78 (2024)
13. Kreković, D., Krivić, P., Žarko, I.P., Kušek, M., Le-Phuoc, D.: Reducing communication overhead in the iot-edge-cloud continuum: a survey on protocols and data reduction strategies. arXiv preprint arXiv:2404.19492 (2024)