Middle East Journal of Scientific Publishing
Vol. (9) Issue (5)
Edition 35th
2026(1 - 17)

From Cloud to Mobile Edge Computing: A Survey of Architectures, Challenges, and the Scalability Gap in Next-Generation Networks

Sherin Hijazi,
Anas Abdelraziq
Palestine Technical University,
Kadoorie, Department of Computer Science, Palestine, Tulkarem.
Published on: 6 may 2026

Abstract

The move from centralized cloud computing to Mobile Edge Computing technology has enabled lower latencies and real-time processing in mobile and IoT networks. This survey outlines the transition from cloud computing to Mobile Edge Computing by providing architectural insights along with exploring scalability issues. A systematic literature review has been done using the IEEE Xplore and Google Scholar databases from 2015 to 2025, which produced highly relevant peer-reviewed journal papers indexed with Scopus. Results have shown that while handling latency and mobility for both offloading schemes and reinforcement learning–based techniques is fairly reasonable, system throughput declines drastically after reaching a certain congestion level. Other trends under observation are around Blockchain from the security domain area to Digital Twins that can enable predictive offloading and AI-driven orchestration. Unfortunately, most of the existing solutions are verified only in controlled scenarios and lack support for large heterogeneous deployments. This survey highlights that scalability needs to be considered as a first-class requisite in next-generation Mobile Edge Computing architecture and outlines future perspectives on distributed orchestration, lightweight AI, and edge-cloud elasticity.

Keywords: Mobile Edge Computing (MEC); Scalability Gap; Task Offloading; Reinforcement Learning; Blockchain; Digital Twin; 5G/6G Networks

* Introduction

Internet of Things (IoT) devices are now prevalent, and expectations from users have moved from connecting devices to getting real-time processing and low-latency services. Cloud computing originally responded to these concerns via centralized computing and storage. Conversely, end-users and cloud data centers are separated physically, where network congestion, high latency, bandwidth limitations, and scalability bottlenecks cannot be avoided (Shi et al., 2016).

To get around these limitations, more intermediate models developed. Cloudlets (Talebkhah et al., 2020) and fog computing (Shi et al., 2016) provide a partial connectivity solution which decentralizes computations to the network edge, reducing latency. Based on such foundations, edge computing enhances computation by shifting computations closer to the source of data, while its mobile variant, Mobile Edge Computing (MEC), leverages computation and storage resources available in base stations and access points to support real-time interactions in highly dynamic environments (Garcia Lopez et al., 2015).

Although considerable improvements have been made towards reducing latency, managing mobility, and offloading tasks, a crucial issue that is yet to be extensively explored pertains to scalability. Most MEC-based studies have been performed under limited scale environments, where there is little insight into how systems behave when subjected to dense user population demands. According to Deng et al. 2021, the throughput offered by MEC is enhanced to a certain extent, it then starts decreasing, an issue we will term as the scalability gap.

The objective of the present survey is to provide an overview of the transition path from cloud computing to MEC, highlight the main problems including latency, mobility, security, resource management, and scalability, integrate architectural approaches and solutions adopted to date, and pinpoint scalability issues and make it a focus of future research. Unlike other surveys that concentrate on performance improvement under certain limitations, the current one deals specifically with the behavior of MEC systems when there is an increasing number of users.

The rest of this research is divided as follows: section 2 presents the background and terminology, section 3 describes the methodology, section 4 presents the literature review, section 5 the discussion, and section 6 ends with the conclusion and future works.

* Background and Terminology

To ensure clarity for all readers coming from various backgrounds, this part of the paper presents the main terms that will be used in the survey. The terms needed for the understanding of MEC along with the related concepts are provided in Table 1.

Table 1: Core Terminology

Term / Abbreviation	Explanation
MEC (Mobile Edge Computing)	A paradigm that moves computation and storage from centralized cloud servers to edge servers closer to end-users, reducing latency and improving efficiency.
VEC (Vehicular Edge Computing)	A specialized MEC environment for connected vehicles, where tasks are offloaded to roadside units (RSUs) equipped with edge servers.
IoT (Internet of Things)	A network of interconnected smart devices that generate and exchange data, requiring efficient processing.
IIoT (Industrial Internet of Things)	Application of IoT in industrial settings (Industry 4.0), involving delay-sensitive and computation-intensive tasks.
RSU (Roadside Unit)	A roadside infrastructure element equipped with communication and computing capabilities to support vehicular networks.
RES (Roadside Edge Server)	An edge server deployed at RSUs to process computation tasks offloaded from vehicles.
UAV (Unmanned Aerial Vehicle)	Drones used as flying edge servers to assist overloaded ground infrastructure.
Latency	The time delay between sending a task and receiving the result.
Energy Consumption	The amount of energy used to execute or offload tasks.
Throughput	The number of tasks successfully processed within a given time frame.
Partial Offloading	A flexible strategy where part of a task is executed locally and the rest is offloaded to MEC servers or other devices.

Table 2: Advanced Techniques and Emerging Technologies

Term / Abbreviation	Explanation
ILP (Integer Linear Programming)	A mathematical optimization technique using integer variables to solve resource allocation and scheduling problems.
MDP (Markov Decision Process)	A stochastic model for decision-making where future states depend only on the current state, used to model MEC dynamics.
RL (Reinforcement Learning)	A machine learning approach where agents learn optimal policies by interacting with the environment.
DDPG (Deep Deterministic Policy Gradient)	A deep reinforcement learning algorithm (Actor-Critic) that outputs continuous offloading ratios for finer optimization.
TD3 (Twin Delayed Deep Deterministic Policy Gradient)	An improved version of DDPG designed to enhance stability and performance in continuous action spaces.
MATD3 (Multi-Agent TD3)	A multi-agent extension of TD3 enabling cooperative decision-making among multiple UAVs or devices.
Blockchain	A distributed ledger technology ensuring secure and tamper-proof data sharing in MEC/VEC environments.
PBFT (Practical Byzantine Fault Tolerance)	A consensus mechanism in blockchain systems that ensures reliability and prevents malicious attacks.
Proof of Service	A blockchain-based consensus mechanism where nodes prove service delivery to gain trust.
Digital Twin (DT)	A virtual replica of physical vehicles, RSUs, and networks that synchronizes with real-world data to predict system behavior and optimize offloading decisions.
K-means	A clustering algorithm that partitions data into k groups based on similarity, used to group devices or vehicles.
Gk-means	An enhanced version of K-means introduced in IGNITE, incorporating computing resources, communication quality, and social trust for more accurate clustering.
Social Trust Model	A model evaluating the reliability of Vehicle-to-Vehicle communication links based on direction and speed similarity.
Dynamic Pricing	A mechanism where service providers set adaptive prices for computing resources, optimized via reinforcement learning.
Pareto Optimality	A concept in multi-objective optimization where no objective can be improved without worsening another.
PSO (Particle Swarm Optimization)	A computational intelligence algorithm inspired by the social behavior of bird flocks or fish schools. Each particle represents a candidate solution, and particles move through the solution space by learning from their own and neighbors' experiences. Widely used in MEC research to solve complex optimization problems such as minimizing delay, energy, or cost.
PSOCO	A PSO-based Computation Offloading algorithm designed to obtain Pareto-optimal solutions for minimizing delay and cost in Vehicular Edge Computing.

* Methodology

The research involved applying an approach that is clear, replicable, and scientific in nature.

This methodology was formulated to make it possible for other scholars to duplicate the study and confirm the results obtained from it. The methods utilized involves five main steps, including databases employed, keywords, inclusive/exclusive conditions, appropriate studies identification, and data extraction.

1- Databases

The databases applied in this survey include IEEE Xplore and Google Scholar. To demonstrate that the sources are reliable, it was necessary to find out whether these journals had been indexed in Scopus.

2- Keywords

To conduct an effective search, some specific keywords and Boolean logic operators were used. The most important search string was: -

("Mobile Edge Computing" OR "MEC") AND ("Survey" OR "Review") AND ("Challenges" OR "Scalability" OR "Architecture").

3- Inclusion and Exclusion Criteria

Inclusion criteria: -

1- Studies were published between 2015 and 2025.

2- Written in English.

3- Published in peer-reviewed journals indexed in Scopus.

4- Directly addressing Mobile Edge Computing (MEC) architecture, challenges, or scalability.

5- High-cited papers (≥50 citations) preferred to cover the foundational aspects, while recent papers on pertinent topics despite being less cited are included.

Exclusion criteria: -

1- Conference papers (to ensure peer-reviewed journal quality, though we acknowledge that high-impact conferences contribute valuable insights).

2- Non-peer-reviewed articles.

3- Studies that repeated similar ideas without adding new insights.

4- Studies that deviated from the survey's objectives.

4- **Study Selection Process**

In the beginning, specified keywords used returned roughly 329,000 results on Google Scholar. Limiting these publications to those from after 2015 decreased the total number to 22,400 papers. Imposing a limit of at least 50 citations resulted in a reduction to 1,650 papers. By excluding papers written in conference proceedings, the total number dropped to 400 papers. After journals not included in the Scopus database were omitted, the number reduced to 200 papers. The removal of papers with repetitive and similar ideas, caused the number of papers to be decreased to 150 papers. Finally, studies that were inconsistent with the scope of the survey were also eliminated, leaving only 33 studies that stayed relevant for detailed analysis.

While the total number of 33 studies might appear small compared to comprehensive surveys, these studies focus on representative works that illustrate the main architectural models and arising. This survey is not concerned with providing an exhaustive listing of all existing papers, but it is intended as a critical review of the fundamental studies on MECs.

5- Data Extraction

For each selected study, the following information was systematically extracted and organized in an Excel sheet: -

1- Title of the study.

2- Research problem addressed.

3- Proposed solutions and methodologies.

4- Techniques and models used (e.g., reinforcement learning, blockchain, optimization)

5- Reported challenges and applications.

6- Identified research gaps.

*** Literature review**

1- **From cloud to MEC**

The survey traces the conceptual development from cloud computing to MEC. Cloud computing started as a centralized system designed for storage and processing of large amounts of data, yet this approach faced problems such as latency, bandwidth usage, and low efficiency. To resolve the problem of latency, new computing architectures have been developed. For instance, according to Talebkhah et al. (2020), cloudlets represent mini-data centers that process time-critical applications and, thus, decrease latency. Although this architecture is efficient at resolving the issue of latency, it is not scalable. Similarly, Shi et al. (2016) defined fog computing as an intermediate layer located between edge and the cloud, consisting of gateways and routers. In this case, the edge computing architecture mentioned by these authors implies shifting the computations to a location near the source of information. Conversely, there exist issues related to programmability and complex service control due to the heterogeneity of platforms. Additionally, Garcia Lopez et al. (2015) note that edge computing was derived from previous peer-to-peer and content delivery systems. In their view, decentralization, scalability, and privacy were key features of the paradigm. At the same time, the ability to manage large numbers of heterogeneous edge devices efficiently and securely remained a challenge.

In contrast to existing computing paradigms, MEC is applied in mobile settings, usually on base stations, enabling real-time applications and operations. Nonetheless, even with the developments outlined above, some challenges remain. One of them relates to privacy and security in light of the increasing number of IoT sensors and limited capacity of edge devices, along with the issue of scalability. Based on the analysis, it can be concluded that MEC is a significant step towards decreasing latency and decentralization, but there is still a need to resolve issues related to interoperability, programmability, and scalability.

2- Challenges and Applications

Table 3 provides a comparative outline of challenges and applications discussed in different MEC research. Various studies are compared based on their approach to resolving recurring challenges (latency, mobility, security, privacy, resource allocation, and scalability) and corresponding domains (smart cities, vehicular systems, healthcare, industrial IoT, and immersive entertainment).

Table 3: Summary of Challenges and Applications in MEC Literature

Reference	Year	Reported Challenges	Reported Applications
Fernández et al.	2018	Device heterogeneity, interoperability, security and privacy, resource management, balancing edge/cloud analytics	Smart gateways, structural health monitoring, smart transportation, smart homes
Xie et al.	2019	Coalition formation, secure task offloading, mobility management, energy efficiency	Collaborative vehicular edge computing, autonomous driving, augmented reality, intelligent traffic management
Qiu et al.	2020	5G integration, load balancing, edge intelligence, secure data sharing	Smart grids, connected vehicles, intelligent manufacturing
Carvalho et al.	2021	Lack of standardization, interoperability, limited edge resources, security and privacy	Smart cities, augmented reality, connected vehicles, smart grids
Meneguette et al.	2021	High mobility, intermittent connectivity, limited resources, security and privacy, fairness	Vehicular edge computing, smart traffic management, CAVs, environmental monitoring, infotainment
Xie et al.	2020	QoE requirements, cooperative offloading, multi-node scheduling, mobility management, fault recovery	Satellite-terrestrial integrated edge computing, content caching, holographic communication, AR/VR, smart cities, disaster recovery
Hartmann et al.	2022	Data management, privacy and legal compliance, AI/5G integration, limited resources, usability	Healthcare monitoring, wearable sensors, emergency detection, personalized healthcare
Liu et al.	2022	Heterogeneous data streams, synchronization with cloud, security and privacy, deployment cost, regulation	Smart grids, ubiquitous power IoT, EV management, customer energy services
Ali et al.	2021	Security fragmentation, interoperability, resource allocation, centralized security needs	Healthcare, transportation (V2V, V2I), entertainment (VR/AR), smart cities, industrial automation
Varma & Indukurib	2023	Service placement, interoperability, security and privacy, energy efficiency, latency optimization	Transportation (CAVs), healthcare, entertainment (AR/VR, gaming), smart cities, industrial IoT
Raeisi-Varzaneh et al.	2023	NP-hard scheduling, latency-energy tradeoff, fairness, load balancing, heterogeneity	Connected autonomous vehicles, healthcare BANs, smart cities, AR/VR, surveillance

From Table 3 above, device heterogeneity and interoperability in smart gateways and homes were considered by Fernández et al. (2018). In turn, coalition formation and task offloading in vehicular MEC was the focus of Xie et al. (2019), while 5G and edge intelligence in the domain of industrial applications was addressed by Qiu et al. (2020). At the same time, the problem of non-standardized devices was identified by Carvalho et al. (2021) as opposed to high mobility and quality requirements in vehicular and satellite–terrestrial domains as addressed by Meneguette et al. (2021) and Xie et al. (2020). Domain-specific investigations include the works by Hartmann et al. (2022) who discussed healthcare and privacy in MEC systems presented in figure 1 and by Liu et al. (2022) who examined problems of energy systems in MEC displayed in figure 2. As for the security and service placement strategies, they were explored by Ali et al. (2021) and Varma & Indukurib (2023) focused on AI and blockchain-based mechanisms. Lastly, Raeisi-Varzaneh et al. (2023) considered scheduling in MEC networks as an NP-hard problem with a rigorous mathematical solution which raises doubts as to applicability.

Thus, one can conclude from the analysis provided above that different research is characterized by a focus on various challenges (such as performance, security, privacy, and service placement) and specific applications (smart cities, vehicular applications, industry, energy systems, etc.). The diversity of approaches in addressing existing issues and challenges in MEC systems suggests the necessity to develop a general and more integrative solution for heterogeneous MEC environments.

3- Architecture

The current study does not consider all possible architectural aspects, however, the analyzed papers show that the majority of edge computing architectures follow a layered approach, which is composed of the device layer (sensors, IoT devices, or mobile terminals), the edge layer (which involves local servers, gateways, or base stations performing calculations and storage operations), and the cloud layer (for analytics and long-term data storage). While a three-layered architecture provides an easy-to-implement structure, its application shows many variations depending on the technologies used. For example, whilst Carvalho et al. (2021) mentioned performance improvements due to the use of 5G and AI, Varma and Indukuri (2023) emphasize security and energy efficiency with the help of blockchain technology. Meanwhile, security and privacy was prioritized by Ali et al. (2021). In other words, although the three-layered model is commonly accepted by researchers, there appears to be various approaches for improvement based on performance, privacy, and security without achieving uniformity. Such innovative methods allow for improving the possibilities offered by MEC but at the same time causes difficulties related to interoperability. Therefore, further research should revolve around developing adaptive architectures for integrating different technologies efficiently.

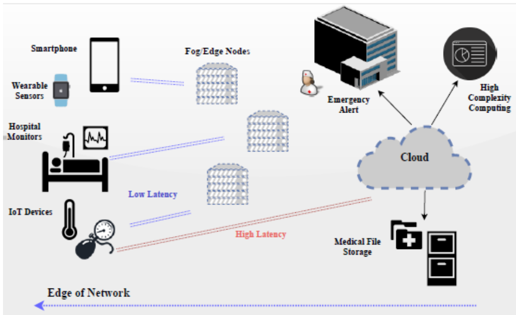

Figure 1: Typical edge computing architecture (Hartmann et al., 2022)

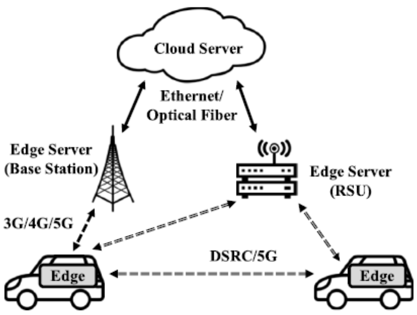

Figure 2: Mobile edge computing scenario in the vehicular domain (Liu et al., 2019)

4- Applied Studies

These challenges have, therefore, inspired several studies which have developed specific mechanisms for dealing with various aspects associated with MEC. Among others, MEC requires constant improvements in many performance aspects such as latency, energy expenditure, throughput, efficiency of task offloading, resource allocation, and security. In order to solve these problems, researchers have recommended a number of methods that are application dependent. For instance, Naouri et al. (2021) proposed the Device-Cloudlet-Cloud framework to minimize latency and communication expenses in the IoT environment presented in figure 3. Meanwhile, Lang et al. (2022) proposed a blockchain-cooperative offloading framework to improve vehicular networks' security. The use of reinforcement learning (DDPG) was recommended by Deng et al. (2021) to minimize delays in the multi-user industrial IoT setting.

Moreover, Zhao et al. (2022) used multi-agent deep reinforcement learning to optimize energy–delay trade-offs for UAV-assisted MEC, and Zhao et al. (2023) suggested a combination of digital twin and clustering together with deep learning in order to increase the precision of offloading and decrease the cost of vehicular edge computing. Moreover, Xu et al. (2021) suggested an ILP-based heuristic method to increase throughput subject to the handover issue in the high-speed railway mobile edge computing networks.

Other relevant works include Deng et al. (2021)'s modeling MEC through a Markov decision process approach in order to optimize user association and resource allocation exhibited in figure 4, Luo et al. (2021) recommending a particle swarm optimization technique to simultaneously minimize delay and cost in vehicular MEC, and Li et al. (2019) focused specifically on a cache-based scheduling algorithm in order to increase data locality and reduce transmission time. Finally, Sadatdiynov et al. (2023) have conducted an extensive review of various optimization techniques used in MEC including Lyapunov, convex, heuristic, game-theoretical, and machine learning approaches.

Comparative analysis of the studies shows that there are some differences in their methodologies. For example, Naouri et al. (2021) and Xu et al., (2021) leverage hierarchical models and ILP heuristics to improve throughput and decrease latency, while Deng et al. (2021) and Zhao et al., (2023) propose to develop reinforcement learning and Digital Twin approaches for offloading optimization in varying conditions. Conversely, Lang et al., (2022) and Ali et al. (2021) prioritize building trust and security based on blockchain technology, whereas Luo et al. (2021) and Li et al. (2019) focus on optimization algorithms, including PSO and cache-aware scheduling. This range illustrates that although MEC is innovative, it is not cohesive enough, each paper targets a specific issue (delay, energy, security, or throughput) but does not consider scalability and interoperability across billions of different devices.

Overall, these functional strategies verify the practical significance of MEC in building trust, improving throughput, and reducing delays. Nonetheless, many studies are only applicable to specific fields and artificial environments. Thus, the scalability problem persists, emphasizing the significance of integrated methods and adaptable architectures to ensure high performance in large IoT systems and future 5G/6G networks.

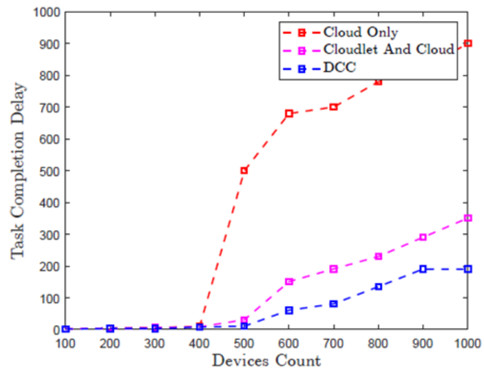

Figure 3: Hierarchical task distribution across Device–Cloudlet–Cloud layers (Naouri et al., 2021)

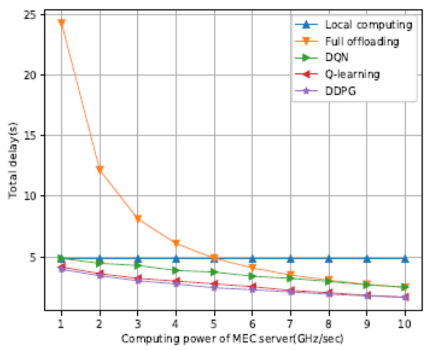

Figure 4: Delay vs. MEC server computing power under different offloading strategies (Deng et al., 2021)

5- Scalability Gap

Whereas considerable efforts have been made to tackle the issues associated with latency, mobility management, and security, the aspect of scalability appears to be overlooked in the reviewed literature. Specifically, the majority of the research examined performance under certain conditions, and there is little evidence about scalability of MEC systems when used for a large number of users. For instance, the analysis by Deng et al. (2021) revealed that an increase in throughput is observed until a certain threshold of congestion is reached, after which the overall system performance is negatively impacted.

Comparatively reviewing the literature, it can be concluded that while the papers authored by Naouri et al. (2021) and Zhao et al. (2022, 2023) focus on reducing latency and achieving trade-offs between energy consumption and time delays, the works published by Lang et al. (2022) and Ali et al. (2021) emphasize the importance of building trustable and secure architectures. Notably, even advanced methods of MEC optimization, including reinforcement learning (Deng et al., 2021) or blockchain-based offloading (Varma & Indukurib, 2023), have been experimentally evaluated mostly in the setting of controlled environments. As a result, a crucial gap becomes apparent. Despite the significant advances in performance and security of MEC, the technology lacks in scalability. This in turn, can be interpreted as an indicator that scalability becomes one of the core aspects to be addressed in future MEC research.

To conclude, the analyzed literature shows that scalability should not be considered a simple technical improvement but rather the basis for MEC to become the basic infrastructure of the next generation.

Since scalable architecture and resource management strategies appear to be required to achieve efficient performance, future studies will have to fill in this gap and prove that the technology can operate effectively in large-scale implementations, especially concerning widespread IoT expansion and within 5G/6G networks.

6- Complementary Contributions and Emerging Trends

Apart from the main issues described above, there are complementary research works on the development of MEC-related technologies. Concerning security and privacy, Zhang et al. (2018) and Shahzadi et al. (2017) presented their initial framework for lightweight encryption and interoperability, whereas Sonmez et al. (2018) developed EdgeCloudSim that allows evaluating MEC performance under both mobility and security setups. In vehicular applications, Tang et al. (2022) discussed caching methods to improve offloading techniques, while Li et al. (2022) proposed using energy harvesting for the purpose of optimizing task scheduling. Finally, Martín et al. (2018) presented CoAP gateway architecture, allowing linking legacy IoT devices with edge foundations.

As far as further research directions are concerned, one can identify at least three areas that should be prioritized in the coming years. First, Xie et al. (2021) focused on the use of serverless-edge systems for dynamic scheduling purposes, though Kong (2023) emphasized the importance of intelligent architecture design within edge-enabled 6G systems. The role of SDN and NFV integration was highlighted by Ahmed and Rehmani (2017). They provided the first taxonomy of MEC approaches. New models like Space Air Ground Integrated Networks (SAGIN) promise universal edge coverage in the future, but Carvalho et al. (2021) called for the fog, cloudlets, and MEC system standardization. Optimization techniques, like Lyapunov, heuristics, and game-theoretical models, have been examined by Sadatdiynov et al. (2023).

In conclusion, one can state that the current literature makes important advances towards addressing the existing MEC research gaps. However, there are some that require more attention, such as the need for unifying frameworks, strategic integration of blockchain and AI to reduce operational overhead, and synchronization of rigor and scalability in practice.

*** Discussion**

From a literature standpoint, it becomes clear that the field of MEC is undergoing continual development and evolution. The concept of MEC has grown and developed gradually from cloud computing infrastructure to its current state. Indeed, all MEC solutions can be traced back to the initial idea of creating highly performant cloud computing environments to the creation of the new computing concept aimed at decreasing latency. Nevertheless, even now, certain aspects remain problematic in relation to MEC.

1- Comparative Analysis of Challenges and Applications

Based on the analysis of relevant articles, there appears to be several recurring challenges when designing MEC systems. These include reducing latency, managing mobility, security and privacy considerations, resource management and allocation, and scalability of applications for heterogeneous devices.

For instance, according to Fernández et al. (2018) and Carvalho et al. (2021), some of the problems to consider include the issue of interoperability and limited resources. On the other hand, Raeisi-Varzaneh et al. (2023) and Xie et al. (2020) emphasize the complexity associated with cooperative offloading of tasks and NP-hard task scheduling, respectively. As for potential applications, a wide variety can be observed from smart cities and healthcare applications to AR/VR solutions and self-driving vehicles.

However, it is important to note that while latency and mobility can be successfully addressed using methods like offloading tasks or reinforcement learning, scalability appears to be relatively ignored by researchers, who tend to focus on optimizing systems within the limits of controlled or simulated environments.

2- Architectural Patterns and Integration Trends

The literature follows a tri-layer architectural approach encompassing the device layer (sensors and mobile devices), edge layer (gateways and base stations), and cloud layer (analytics and centralized data storage). Although this approach provides modularity and clarity, the application efficacy depends on seamless integration with emerging technologies.

Empirical studies conducted by Carvalho et al. (2021) and Varma & Indukurib (2023) emphasize the need to integrate AI and blockchain technology to improve its trustworthiness, intelligence, and energy efficiency. Ali et al. (2021) also explored security frameworks powered by 5G technologies, proclaiming that the future of MEC lies in hybrid systems, leveraging cloud reliability and edge flexibility.

Then again, such integrations raise new concerns: -

1- How do AI models scale across edge nodes with limited computation?

2- Can blockchain maintain low latency in real-time MEC scenarios?

3- What trade-offs emerge between decentralization and centralized control?

These questions point to a need for adaptive, context-aware architectures that dynamically reconfigure based on workload, user density, and network conditions.

3- Evaluation of Applied Studies

Applied studies highlight practical methods to tackle key problems inherent to MEC. The common methods adopted include: -

1- Blockchain-based cooperative offloading (Lang et al., 2022)

2- Reinforcement learning for delay minimization (Deng et al., 2021)

3- Multi-agent DRL for UAV-assisted MEC (Zhao et al., 2022)

4- Digital Twin integration for vehicular MEC (Zhao et al., 2023)

5- ILP-based heuristics for high-speed rail MEC (Xu et al., 2021)

6- Cache-aware scheduling (Li et al., 2019)

These methods are innovative and application-specific optimizations, but their implementation remains limited to certain conditions, rendering the methods nongeneralizable across MEC environments. Moreover, interoperability and coordination across a wider scope, especially within 5G/6G ecosystems, are infrequently considered.

4- Scalability as a Research Gap

Scalability is one of the key issues, but it remains poorly addressed. As explained by Deng et al. (2021), the throughput of Mobile Edge Computing (MEC) systems grows linearly with the number of users up to a certain point, then the system reaches a state of congestion where its throughput declines. Hence, scalability emerges as a critical challenge for MEC that requires attention while still meeting the pertinent needs in terms of energy use, security, and latency.

To address this gap, future research must explore: -

1- Distributed resource orchestration across thousands of edge nodes under dynamic conditions.

2- Load balancing strategies that adapt instantaneously to traffic patterns and mobility.

3- Lightweight and scalable AI models capable of operating on resource-constrained edge devices.
4- Elastic edge–cloud coordination that dynamically shifts workloads based on demand and network state.

6- Conclusion and Future Work

This paper highlights the evolution of cloud computing towards Mobile Edge Computing (MEC) and discusses the relevant issues. The present systematic review covers relevant papers indexed in Scopus to determine how scalability compares to other common challenges, such as reduced latency, efficient handling of mobility, security and privacy concerns, resource allocation, etc.

While latency and mobility are effectively addressed through task offloading and reinforcement learning, scalability remains a critical gap. Most existing solutions are validated under controlled scenarios, and system performance degrades beyond a congestion threshold, a phenomenon observed in Deng et al. (2021). Emerging trends include blockchain for security, Digital Twins for predictive offloading, and AI-driven orchestration, but their scalability under massive deployments remains unverified.

To advance the field, future research should focus on: -

1- Scalable Resource Orchestration: Designing distributed algorithms capable of managing thousands of edge nodes under dynamic traffic and mobility conditions.

2- Interoperable Edge Intelligence: Developing lightweight, federated AI models that operate across heterogeneous devices without centralized coordination.

3- Elastic Edge–Cloud Coordination: Creating adaptive frameworks that balance workloads based on real-time demand, network state, and user density.

Addressing the scalability gap is not merely a technical enhancement but a foundational requirement for MEC to serve as a cornerstone of next-generation 5G/6G networks, smart cities, autonomous vehicles, and industrial automation.

* Acknowledgment

The author would like to thank the Palestine Technical University - Kadoorie for their financial support to conduct this research.

* References

Ahmed, E., & Rehmani, M. H. (2017). Mobile edge computing: Opportunities, solutions, and challenges. Future Generation Computer Systems, 70, 59–63.

Ali, B., Gregory, M. A., & Li, S. (2021). Multi-access edge computing architecture, data security and privacy: A review. IEEE Access, 9, 18706–18721.

Carvalho, G., Cabral, B., Pereira, V., & Bernardino, J. (2021). Edge computing: Current trends, research challenges and future directions. Computing, 103(5), 993–1023.

Deng, X., et al. (2021). Intelligent delay-aware partial computing task offloading for multiuser industrial Internet of Things through edge computing. IEEE Internet of Things Journal, 10(4), 2954–2966.

Deng, Y., et al. (2021). Throughput maximization for multiedge multiuser edge computing systems. IEEE Internet of Things Journal, 9(1), 68–79.

Fernández, C. M., et al. (2018). An edge computing architecture in the internet of things. In Proc. IEEE 21st Int. Symp. Real-Time Distributed Computing (ISORC) (pp. 99–102).

Garcia Lopez, P., et al. (2015). Edge-centric computing: Vision and challenges. ACM SIGCOMM Computer Communication Review, 45(5), 37–42.

Hartmann, M., Hashmi, U. S., & Imran, A. (2022). Edge computing in smart health care systems: Review, challenges, and research directions. Transactions on Emerging Telecommunications Technologies, 33(3), e3710.

Kong, L. (2023). Edge-driven intelligence for 6G networks: Architectures, challenges, and opportunities. IEEE Network, 37(2), 112–119.

Lang, P., et al. (2022). Cooperative computation offloading in blockchain-based vehicular edge computing networks. IEEE Transactions on Intelligent Vehicles, 7(3), 783–798.

Li, C., et al. (2019). Collaborative cache allocation and task scheduling for data-intensive applications in edge computing environment. Future Generation Computer Systems, 95, 249–264.

Li, Y., Wang, S., & Zhang, X. (2022). Energy-aware task offloading with energy harvesting in mobile edge computing. IEEE Transactions on Green Communications and Networking, 6(2), 1123–1136.

Liu, D., et al. (2022). Edge computing application, architecture, and challenges in ubiquitous power internet of things. Frontiers in Energy Research, 10, 850252.

Liu, S., et al. (2019). Edge computing for autonomous driving: Opportunities and challenges. Proceedings of the IEEE, 107(8), 1697–1716.

Luo, Q., et al. (2021). Minimizing the delay and cost of computation offloading for vehicular edge computing. IEEE Transactions on Services Computing, 15(5), 2897–2909.

Martín, D., Lanza, J., Sánchez, L., & Galache, J. A. (2018). A CoAP gateway model for edge computing in IoT environments. Future Generation Computer Systems, 85, 234–248.

Meneguette, R., et al. (2021). Vehicular edge computing: Architecture, resource management, security, and challenges. ACM Computing Surveys, 55(1), 1–46.

Naouri, A., et al. (2021). A novel framework for mobile-edge computing by optimizing task offloading. IEEE Internet of Things Journal, 8(16), 13065–13076.

Qiu, T., Chi, J., Zhou, X., Ning, Z., Atiquzzaman, M., & Wu, D. O. (2020). Edge computing in industrial internet of things: Architecture, advances and challenges. IEEE Communications Surveys & Tutorials, 22(4), 2462–2488.

Raeisi-Varzaneh, M., et al. (2023). Resource scheduling in edge computing: Architecture, taxonomy, open issues and future research directions. IEEE Access, 11, 25329–25350.

Sadatdiynov, K., Cui, L., Zhang, L., Huang, J. Z., Salloum, S., & Mahmud, M. S. (2023). A review of optimization methods for computation offloading in edge computing networks. Digital Communications and Networks, 9(2), 450–461.

Shahzadi, S., et al. (2017). Multi-access edge computing: Open issues, challenges and future perspectives. Journal of Cloud Computing, 6(1), 30.

Shi, W., Cao, J., Zhang, Q., Li, Y., & Xu, L. (2016). Edge computing: Vision and challenges. IEEE Internet of Things Journal, 3(5), 637–646.

Sonmez, C., Ozgovde, A., & Ersoy, C. (2018). Edgecloudsim: An environment for performance evaluation of edge computing systems. Transactions on Emerging

Telecommunications Technologies, 29(11), e3493.

Talebkhah, M., et al. (2020). Edge computing: Architecture, applications and future perspectives. In Proc. IEEE 2nd Int. Conf. Artificial Intelligence in Engineering and Technology (IICAIET) (pp. 1–6).

Tang, C., et al. (2022). When cache meets vehicular edge computing: Architecture, key issues, and challenges. IEEE Wireless Communications, 29(4), 56–62.

Varma, P. S., & Indukurib, V. R. (2023). AI and blockchain enabled edge computing for secure and energy-aware service placement. Journal of Parallel and Distributed Computing, 172, 45–60.

Xie, J., Jia, Y., Chen, Z., & Liang, W. (2021). Serverless edge computing: Architecture, challenges, and open issues. IEEE Wireless Communications, 28(5), 112–118.

Xie, R., et al. (2019). Collaborative vehicular edge computing networks: Architecture design and research challenges. IEEE Access, 7, 178942–178952.

Xie, R., et al. (2020). Satellite-terrestrial integrated edge computing networks: Architecture, challenges, and open issues. IEEE Network, 34(3), 224–231.

Xu, J., et al. (2021). Throughput maximization of offloading tasks in multi-access edge computing networks for high-speed railways. IEEE Transactions on Vehicular Technology, 70(9), 9525–9539.

Zhang, J., et al. (2018). Data security and privacy-preserving in edge computing paradigm: Survey and open issues. IEEE Access, 6, 18209–18237.

Zhao, L., et al. (2023). A digital twin-assisted intelligent partial offloading approach for vehicular edge computing. IEEE Journal on Selected Areas in Communications, 41(11), 3386–3400.

Zhao, N., et al. (2022). Multi-agent deep reinforcement learning for task offloading in UAV-assisted mobile edge computing. IEEE Transactions on Wireless Communications, 21(9), 6949–6960.