Edge Computing Benefits in Low-Latency IoT Applications: A Comprehensive Survey

Prof. Claudio Enrico Palazzi
Department of Mathematics
University of Padua
Padua, Italy
cpalazzi@math.unipd.it

Michael Amista'
Department of Mathematics
University of Padua
Padua, Italy
michael.amista@studenti.unipd.it

***Abstract*—The rapid proliferation of Internet of Things (IoT) devices has generated unprecedented volumes of data, creating significant challenges in computational processing, latency, and resource management. This comprehensive survey explores the transformative role of edge computing in addressing critical limitations of traditional cloud-based IoT architectures. By examining recent advances in edge computing technologies, this work provides insights on how computational capabilities can be extended from centralized cloud infrastructure to the network's edge, enabling more efficient, responsive, and intelligent IoT applications.**

The investigation reveals that edge computing fundamentally reshapes IoT capabilities by bringing computational resources closer to data sources, thereby mitigating network bandwidth constraints and communication latency. The survey systematically analyses edge computing architectures across diverse domains, including mobile applications, industrial manufacturing, and interactive technologies. Emphasis is placed on mobile edge computing (MEC) paradigm that addresses energy efficiency, computation offloading, and low-latency requirements. Key findings demonstrate edge computing's potential to support real-time processing, enhance computational autonomy, and optimize resource utilization for resource-constrained smart devices.

Furthermore, the survey identifies persistent research challenges and opportunities, highlighting the need for continued innovation in edge computing infrastructure, communication protocols, and intelligent resource management strategies. By providing a comprehensive overview of current technological landscapes, this study offers researchers and practitioners a critical reference for understanding and implementing edge computing solutions in IoT ecosystems.

***Keywords*—Edge computing, Internet of Things (IoT), Low-Latency Applications, Mobile game, IoT-Based Manufacturing, Mobile Edge Computing (MEC), Real-Time Processing, Energy Efficiency**

I. INTRODUCTION: CONTEXT AND CASE STUDIES

The Internet of Things (IoT) represents a transformative technological paradigm that has rapidly evolved over the past decade. At its core, IoT encompasses a vast network of interconnected devices equipped with sensors, software, and network connectivity, enabling them to collect, exchange, and process data. From smart home devices and wearable technologies to industrial sensors and urban infrastructure monitoring systems, IoT has emerged as a critical technological ecosystem that bridges the physical and digital worlds.

The proliferation of IoT devices has been exponential, with global estimates suggesting over 75 billion connected devices by 2025. These devices range from simple sensors measuring environmental parameters to complex systems managing critical infrastructure. However, the initial generation of IoT devices was predominantly limited to data collection and transmission, with minimal on-site computational capabilities.

A. *Limitations of traditional cloud computing*

Traditional cloud computing models have been the primary approach for processing IoT-generated data. In this centralized paradigm, devices collect data and transmit it to remote data centres for processing and analysis. While this approach worked effectively for early IoT applications, it has become increasingly inadequate for emerging low-latency use cases.

The primary limitations of traditional cloud computing in IoT contexts include:

- **Network Bandwidth Constraints**: Transmitting large volumes of data from numerous devices to centralized cloud servers creates significant network congestion and bandwidth challenges.
- **Communication Latency**: The physical distance between IoT devices and cloud data centres introduces substantial processing delays, rendering the approach unsuitable for low-latency applications.
- **Resource Inefficiency**: Sending all collected data to remote servers for processing is computationally and energetically inefficient, especially for resource-constrained devices.
- **Privacy and Security Concerns**: Continuous data transmission to external servers raises critical questions about data privacy and potential security vulnerabilities.

B. *Emergence of edge computing*

Edge computing has emerged as a revolutionary solution to address the inherent limitations of traditional cloud-based IoT architectures. By bringing computational capabilities closer to the data sources, directly at the network's edge, this paradigm fundamentally transforms how IoT systems process and analyse information.

Key characteristics of edge computing include:

- Distributed computational resources
- Local data processing and filtering
- Reduced latency
- Enhanced real-time decision-making capabilities
- Improved energy efficiency
- Increased system autonomy

The technological advancements in embedded systems have been fundamental in enabling edge computing. Modern IoT devices now possess sufficient computational resources to perform complex data processing tasks locally, marking a significant departure from earlier generations of limited-capability sensors.

C. *Paper scope and contribution*

This survey provides a structured exploration of edge computing technologies within IoT systems, highlighting critical computational challenges and advancements. The content is organized as follows:

- **Section I** introduces the context of IoT and the emergence of edge computing as a response to the limitations of traditional cloud computing architectures, emphasizing reduced latency, enhanced autonomy, and resource optimization.
- **Section II** delves into the core concepts of edge computing, including its roles, enabling technologies, key requirements, and applications in IoT environments.
- **Section III** presents a case study on mobile gaming, showcasing the advantages of edge computing for achieving low-latency performance in real-time applications.
- **Section IV** examines the application of edge computing in industrial manufacturing, discussing its architectural design, key benefits, and challenges in implementation.
- **Section V** explores Mobile Edge Computing (MEC) as a specialized framework for low-latency IoT applications, detailing strategies for efficient computation offloading and energy optimization.
- **Section VI** concludes with a synthesis of insights, open research challenges, and directions for future innovation in edge computing for IoT ecosystems.

II. EDGE COMPUTING IN INTERNET OF THINGS

This section provides a complete overview on edge computing, highlighting the main features, its role in IoT networks, key requirements for successful deployment, enabling technologies and the different computing paradigms. This section also provides some edge computing applications that witness the strength of this paradigm in low-latency IoT applications.

A. *Roles of edge computing*

Edge computing emerges as a strategic technological paradigm that fundamentally transforms how Internet of Things (IoT) systems process and interact with data. As highlighted by Hassan et al. [1], edge computing plays multiple critical roles in IoT ecosystems:

Data Acquisition and Processing: Edge devices, including sensors and intelligent machines, now possess the capability to capture streaming data and perform immediate analysis. This approach aligns with the evolving computational philosophy of "moving the algorithm to the data" rather than transporting data to centralized algorithms. For instance, in smart transportation systems, traffic light cameras can simultaneously capture and analyse data, enabling instantaneous decision-making to optimize traffic flow.

Inferential Controls: Edge devices are increasingly equipped with sophisticated inferential capabilities, allowing them to interpret environmental contexts accurately. These devices can communicate with broader infrastructural systems while making intelligent, contextually-aware decisions. In smart transportation scenarios, this translates to providing drivers with highly intelligent navigation instructions by integrating data from GPS and multiple camera inputs.

Real-Time Data Analysis: By enabling localized data analysis at the point of generation, edge computing significantly reduces information latency. This approach offers multiple advantages:

- Faster generation of actionable insights
- Reduced network bandwidth consumption
- Decreased operational costs
- Immediate decision-making capabilities

Across industries such as manufacturing, healthcare, telecommunications, and finance, edge computing facilitates more efficient and responsive IoT implementations.

Enhanced Data Security: By localizing data collection and analysis, edge computing inherently improves data security. Reduced extensive routing minimizes potential vulnerability points, making it easier to identify and mitigate suspicious activities before they escalate into significant security breaches. In fact, since edge computing provides computational resources, data are generated and processed within nodes in the edge network. This allows to remove private information before sending data to the cloud, avoiding potential privacy issues.

B. *Key requirements for successful deployment*

As discussed by Hassan et al. [1], successful deployment of edge computing in IoT environment should meet specific requirements to achieve the different features provided by edge computing paradigm. Since several of these requirements are conflicting, application designers must find a good balance among all of them. These key requirements are:

Latency Minimization: Edge computing addresses the delay of traditional cloud models by processing data closer to its source, enabling real-time responsiveness. This is critical for applications like healthcare, autonomous vehicles, and industrial automation, where even minimal delays can have severe consequences.

Reliability: Reliable edge systems ensure consistent performance with minimal downtime, even in diverse and challenging conditions. This includes maintaining computational integrity and service quality while integrating with varied IoT ecosystems.

Mobility Support: As IoT devices increasingly operate on the move, edge computing must provide seamless connectivity and uninterrupted service. Robust mobility support ensures smooth handoffs, session continuity, and reliable performance in dynamic network environments.

Real-time interactions: Many IoT applications require immediate data processing and response. Edge computing supports real-time interactions essential for systems like collision avoidance in autonomous vehicles and industrial monitoring.

Security: Distributed edge systems introduce multiple vulnerability points. Comprehensive security measures, including data protection and continuous threat monitoring, are vital to safeguard against attacks while leveraging localized data processing.

Interoperability: Edge computing must integrate seamlessly across diverse hardware, protocols, and applications. Standardized interfaces and adaptive frameworks enable effective communication and collaboration in complex IoT ecosystems.

C. *Edge computing paradigms*

Edge computing represents a decentralized computing platform that brings cloud computing capabilities closer to IoT devices at the network edge. Unlike traditional cloud computing, which relies on centralized remote servers, following Hassan et al. [1], edge computing offers several paradigmatic approaches:

- **Fog Computing**: Utilizes local network devices like routers or switches within a limited geographic region to provide computational services, emerging as a premier technology following IoT success.
- **Cloudlet Computing**: Involves performing delay-sensitive and computation-intensive tasks on local area network servers, reducing processing time for IoT devices.
- **Mobile Edge Computing (MEC)**: Brings cloud computing capabilities to the edge of cellular networks, with computational and storage services provided at base stations.
- **Mobile Ad Hoc Cloud (MAC)**: Leverages shared resources of available mobile devices in local proximity to process computation-intensive tasks.
- **Hybrid Computing**: Combines cloud and edge computing infrastructures to overcome cloud latency issues while maintaining access to large computing resources.

D. *Enabling technologies*

Edge computing platforms are made possible by several key enabling technologies that are also crucial in the evolution of current mobile networks towards 5G. Based on [2], three fundamental technologies emerge as critical enablers for edge computing: virtualization, NFV+SDN and computational offloading.

Virtualization enables multiple independent software instances to run on a single physical server while maintaining isolation between instances. While Virtual Machines (VMs) have been the dominant means of virtualization in cloud environments, container-based virtualization has emerged as a lightweight alternative. Containers share the host operating system resources without requiring a separate OS for each instance, resulting in reduced start times and improved performance. Live migration capabilities in both VMs and containers are particularly important for edge computing, allowing the movement of computing resources between servers to adapt to user mobility or optimize resource utilization.

Network Function Virtualization (NFV) enables the implementation of network functions as software modules running on general-purpose hardware, decoupling software from the underlying hardware. This approach allows network functions to be executed on general-purpose nodes rather than dedicated hardware. **Software-Defined Networking (SDN)** complements NFV by separating the control plane from the data plane, enabling more flexible network management through abstractions and a centralized controller. Together, these technologies enable automated deployment of virtual resources and flexible network configuration, crucial for managing edge computing environments.

Computation offloading allows resource-constrained mobile devices to transfer processing-heavy tasks to edge computing platforms or the cloud. This capability extends battery life of end devices by avoiding complex local processing and enables resource-intensive applications to run on devices with limited capabilities. Offloading to edge nodes instead of the cloud results in lower energy consumption at the end device while supporting applications such as mobile gaming, natural language processing, and mobile healthcare.

These enabling technologies form the foundation for implementing efficient and flexible edge computing solutions, particularly in scenarios requiring low latency and efficient resource management.

E. *Data types*

Edge computing addresses various data types categorized primarily by their delay sensitivity:

- **Hard Real-Time Data**: Requires absolute zero tolerance for delay. These are critical data streams where even minimal latency can compromise system performance or safety.
- **Soft Real-Time Data**: Allows for bounded delays within specific time constraints. These data types can tolerate some processing time without significant performance degradation.
- **Non-Real-Time Data**: Represents delay-tolerant applications where immediate processing is not crucial, offering more flexibility in data handling and analysis.

The primary motivation for edge computing is its ability to handle these diverse data types more effectively than traditional cloud computing, especially for delay-sensitive applications.

F. *Applications*

According to what discussed by Hassan et al. [1], edge computing has emerged as a transformative technology across multiple domains:

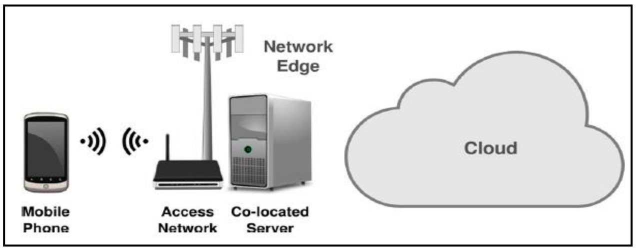

Fig. 1. Mobile game: testbed setup used for the network edge scenario.

- Smart Homes: Enables automated monitoring and metering of utilities like water, electricity, and gas, allowing real-time data analytics at the network edge.
- Healthcare: Supports remote health monitoring, enables immediate reactions to health-related data, and resolves latency issues inherent in cloud-based medical applications.
- Video Surveillance: Facilitates efficient storage, management, and processing of video content from multiple cameras and sensors, often collaborating with cloud computing.
- Smart Grid: Supports real-time energy management by sensing consumption and distribution patterns, with edge computing providing agility and load distribution.
- Smart Cities: Assists in various urban management tasks such as street lighting control, environmental monitoring, emergency route exploration, and automated garden watering.
- Smart Logistics: Automates transaction flows between manufacturers and consumers, enhancing cost and time efficiency.
- Environment Monitoring: Enables comprehensive tracking of critical environmental parameters like gas concentration, water levels, soil humidity, and land position changes.

These applications demonstrate edge computing's potential to transform IoT by bringing computation closer to data sources, reducing latency, and enabling more responsive and intelligent systems.

III. Use case: mobile game

Premsankar et al. [2] carried out an experimental evaluation of mobile gaming using a prototype edge computing platform. This experiment witnesses the power of edge computing deployment to achieve low-latency performances required by resource-demanding mobile games.

A. *Testbed setup*

The researchers used the open-source GamingAnywhere cloud gaming platform to conduct their experiments. They focused on measuring the response delay, which is defined as the time elapsed between an action performed by the user and the occurrence of the corresponding outcome at the client device. The response delay includes three main components:

1. **Processing Delay (PD)**: The time taken by the server to process the user input and render the corresponding frame.
2. **Playout Delay (OD)**: The time taken by the client to decode and display the frame on its own screen.
3. **Network Delay (ND)**: The round-trip time (RTT) between the client and the server.

The experiments were carried out using a Google Nexus 5 mobile phone as the client device and a workstation with a 4-core Intel Xeon E3-1230 CPU, 16 GB of RAM, and two NVIDIA Quadro 2000 GPUs as the gaming server. In the experiments the authors considered two access technologies, i.e., Wi-Fi and LTE. The game of interest was Neverball, a game that is representative of a larger class of applications that rely on rendering complex 3-D environments, including virtual and augmented reality, and that need so a fast response to provide a smooth user interaction.

The researchers considered three different server deployment scenarios: 1) a local deployment at the network edge (as illustrated in Fig. 1), 2) a special-purpose cloud computing infrastructure, and 3) a commercial public cloud provider.

B. *Experimental results*

The researchers first studied the impact of server deployment on the network delay (ND). They found that the edge network scenario, with the server co-located with the LTE base station, achieved an ND of less than 20 ms, which is considered the state of the art of currently available wireless communication technologies. In contrast, the public cloud scenarios incurred significantly higher delays, at least twice as much as the edge network scenario, around 50 ms.

Next, the researchers examined the overhead of different virtualization technologies (bare metal, container, and virtual machine) and how the screen resolution affected the response delay. They found that the performance of containers was almost the same as the bare metal configuration, while hypervisor-based virtualization incurred about a 30% higher processing delay (PD).

Finally, the researchers evaluated whether the use of more powerful computing resources in the cloud could compensate for the higher network delay. They found that the additional computational resources offered by the cloud were not effective for the full HD resolution, as most of the PD was due to the encoding of the source video content, rather than rendering.

C. *Implications and discussion*

The researchers concluded that hosting computing resources very close to the end-users, at the access network edge, is the only viable option to achieve a satisfactory quality of experience for mobile gaming applications. While a response delay below 150 ms is generally considered acceptable for interactive applications, fast-paced interactions cannot tolerate delays beyond 70 ms. The edge network configuration allowed them to play the game at an HD resolution with processing times below 70 ms, which was not possible with the cloud-based deployments.

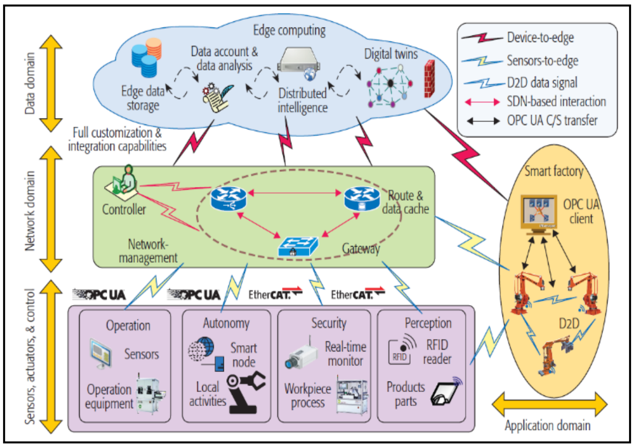

Fig. 2. Architecture of an edge computing platform in IoT-based manufacturing.

The researchers also discussed the implications of their findings for future IoT applications, highlighting the need for advanced virtualization, network function virtualization (NFV), and software-defined networking (SDN) techniques to handle the complexity of resource allocation and optimization in edge computing environments.

IV. USE CASE: INDUSTRIAL MANUFACTURING

Industrial IoT environments present unique challenges that make them particularly suitable for edge computing implementations. According to Chen et al. [3], edge computing in manufacturing environments enables critical capabilities including agile connections, real-time data analytics via edge nodes, highly responsive cloud services, and enhanced privacy policies. Following what discussed by Chen et al. [3], this section provides an overview on edge computing deployments in IoT-based manufacturing.

A. *Edge computing architecture in manufacturing*

The implementation architecture spans four key domains that form a comprehensive framework for industrial applications:

Device Domain: The device domain represents the foundation of the architecture, integrating seamlessly with field devices including sensors, meters, robots, and machine tools. This domain implements a flexible communication infrastructure that supports standardized communication models based on protocols like OPC UA and DDS. Through these protocols, the system enables unified semantics for information interaction while ensuring data security and privacy at the device level. Additionally, the device domain provides essential computing and storage capabilities that allow for dynamic strategy adjustment based on real-time conditions.

Network Domain: Building upon the device layer, the network domain establishes flat connectivity between field equipment and data platforms. This domain leverages Software Defined Networking (SDN) to achieve effective separation of network transmission and control functions. A critical component of this domain is the implementation of Time-Sensitive Network (TSN) protocols, which enable precise management of time-critical data and maintain consistent handling of sensitive time nodes. This careful attention to timing ensures that critical manufacturing processes receive the necessary network resources when needed.

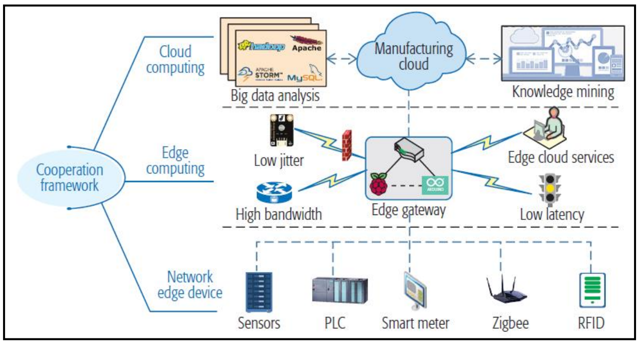

Fig. 3. Cooperation mechanism in IoT-based manufacturing.

Data Domain: The data domain serves as the intelligent processing centre at the edge, performing crucial functions such as data cleaning and feature extraction at the source. This domain enables real-time response based on IoT data while processing abstract data for manufacturing resource virtualization. By improving the availability and quality of heterogeneous industrial data, the data domain supports pre-defined responses to real-time events, ensuring that the system can react appropriately to changing conditions without requiring constant communication with the cloud.

Application Domain: At the highest level, the application domain integrates technologies across network, data, computing, and control functions. This domain enables dynamic management and optimal scheduling of field equipment while providing service composition based on specific manufacturing process requirements. The result is a flexible and interoperable system capable of supporting a wide range of intelligent applications.

B. *Key benefits and implementations*

This section explores the key advantages and practical implementations of edge computing, highlighting its role in revolutionizing industrial processes through advanced maintenance strategies, data analytics, and seamless integration with cloud computing.

Active Maintenance: The integration of edge computing in manufacturing has demonstrated significant benefits, particularly in active maintenance scenarios. For instance, a case study involving a candy packaging production line revealed remarkable improvements in both production efficiency and network optimization. The self-organized task mechanism showed superior agility, particularly when handling large order volumes exceeding 2000 orders. This implementation also led to a dramatic reduction in backbone network traffic, cutting data transmission rates from 16-17 Mb/s to an optimized 5-6 Mb/s, representing a 60% decrease.

Information Fusion and Advanced Analytics: The system's information fusion and analytics capabilities represent another crucial advantage. Edge nodes perform sophisticated data processing, including feature extraction from sensor signals, time series analysis, frequency analysis, and wavelet analysis. This local processing enables the implementation of AI/ML models at the edge, supporting real-time prediction capabilities and dynamic reasoning based on local data. The system continuously updates its knowledge base, ensuring that decision-making improves over time.

Cloud-Edge Cooperation: Fig. 3 highlights how the cooperation between cloud and edge computing should work in a way to create a powerful synergy in industrial environments. Edge layers handle real-time processing, short-term data analysis, local business logic execution, and data security, while cloud layers focus on big data analysis, knowledge mining, periodic maintenance planning, and long-term decision support. This division of responsibilities ensures optimal performance at both levels while maintaining system efficiency.

C. *Industrial implementation challenges*

The implementation of edge computing in manufacturing faces several key challenges. Protocol compatibility remains a significant concern, requiring support for multiple access modes and diverse communication protocols. Real-time requirements demand careful attention to time-sensitive data processing and low-latency communication needs. System integration also poses challenges, particularly regarding

compatibility with legacy equipment and scalability considerations.

D. *Future implications*

Looking to the future, this industrial use case demonstrates several promising directions for development. The evolution of autonomous systems, continued network optimization, and digital transformation through digital twins and virtual manufacturing environments represent exciting possibilities. As these technologies mature, edge computing will play an increasingly crucial role in enabling the smart factories of tomorrow, supporting advanced process optimization and improved decision-making at the edge.

Through this comprehensive analysis of industrial manufacturing applications, it becomes clear that edge computing provides essential capabilities for modern smart factories while creating new opportunities for optimization and automation. The successful integration of edge computing in manufacturing environments demonstrates its potential to revolutionize industrial processes and pave the way for Industry 4.0 implementations.

V. MOBILE EDGE COMPUTING TO ENHANCE LOW-LATENCY IOT APPLICATIONS

Mobile Edge Computing (MEC) has emerged as a transformative solution in the IoT landscape, addressing critical challenges in latency, bandwidth, and computation capacity. As IoT networks grow to include billions of devices, the demand for real-time data processing and low-latency responses becomes paramount, particularly in applications such as autonomous vehicles, smart grids, and healthcare systems. As discussed before, traditional cloud computing, with its centralized architecture, struggles to meet these requirements due to latency and bandwidth constraints. MEC bridges this gap by bringing computation closer to the data source, enabling swift decision-making and efficient resource utilization. Following what discussed by Zhang et al. [4], this section explores MEC's role in enhancing low-latency IoT applications.

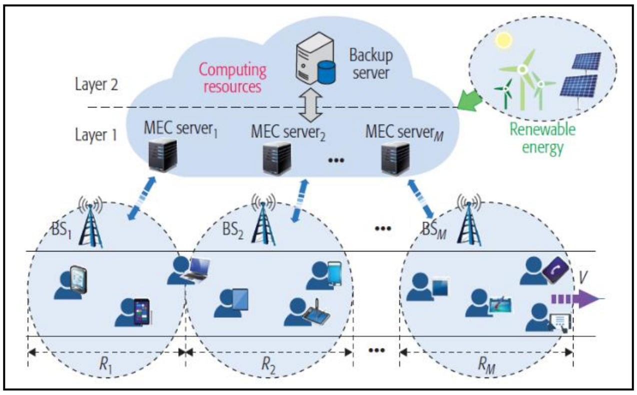

Fig. 4. Mobility-aware hierarchical MEC framework.

A. *MEC in IoT: core characteristics*

Proximity: MEC servers are strategically positioned near IoT devices, significantly reducing the data transmission distance. This proximity ensures faster data processing and supports real-time IoT applications**.**

Low Latency: By processing data at the edge of the network, MEC eliminates delays caused by round-trip data transfers to the cloud. This improvement is crucial for applications like autonomous driving and industrial automation.

High Bandwidth Utilization: MEC leverages the local access network, optimizing data transmission and ensuring high-speed communication between devices and servers.

Location and Network Context Awareness: MEC servers analyze local data, including the location and context of connected IoT devices. This contextual understanding allows servers to make decisions based on real-time information, improving service delivery and resource allocation.

Flexible Deployment: The adaptable nature of MEC enables its integration into diverse scenarios, from mission-critical applications to everyday IoT operations, tailored to specific performance demands.

Heterogeneous Resource Synergy: MEC synergizes resources across cloud, edge, and communication domains, maximizing computational efficiency and meeting diverse application needs.

Mobility Awareness: As ubiquitous IoT devices move, MEC supports continuous task offloading and service delivery. Mobility-aware mechanisms ensure seamless transitions between MEC servers, enabling uninterrupted operations in dynamic environments like vehicular networks.

Security and Privacy Protection: MEC addresses the risks of attacks, such as Distributed Denial of Service (DDoS), by implementing robust security measures. It also ensures that sensitive data, especially in privacy-critical applications, is protected during the offloading process.

Real-Time Response: MEC is designed to meet the demands of emergency IoT applications requiring real-time responses. By addressing latency issues and optimizing task offloading, MEC ensures timely decision-making and action in critical situations.

B. Enhancing IoT applications with MEC

MEC has revolutionized the capabilities of IoT applications by providing real-time data processing and reducing latency across diverse domains. In intelligent transportation, MEC facilitates rapid sensory data analysis for autonomous driving and stable task offloading in dynamic vehicular networks. Smart grids benefit from MEC's distributed computing for efficient energy management and integration with renewable resources and electric vehicles. In agriculture, MEC supports precision farming by processing environmental data locally and enabling collaborative analysis across regions. Healthcare applications leverage MEC for near-patient data processing, emergency response, and privacy-preserving health management. Similarly, smart buildings utilize MEC for efficient automation and energy optimization, while smart retail adopts it for real-time customer behavior analysis and inventory management. By addressing the computational and latency challenges of these applications, MEC serves as a cornerstone technology for next-generation IoT systems.

C. Green and low-latency offloading MEC schemes

Zhang et al. [4] presents an innovative two-layer framework, as depicted in Fig. 4, for managing computation offloading in MEC networks, with a focus on both energy efficiency and low latency. The framework employs a hierarchical structure where the first layer consists of multiple MEC servers powered by renewable energy, each equipped on base stations positioned along a road. These servers have specific computation capacity limits that determine their ability to process offloaded tasks. The second layer introduces a backup computing server that provides additional resources when the first-layer MEC servers become overwhelmed, charging a set price per unit of computing resource utilized.

The system model considers a scenario where multiple mobile smart devices arrive at the road's starting point and travel at a constant speed. Each device carries a computing task characterized by three parameters: the size of task input data, required computing resources, and a delay constraint. These tasks can either be executed remotely by offloading to a MEC server or processed locally on the device itself. When tasks are offloaded, the total time cost includes the device's running time to access the MEC server, task input data transmission time, and the actual computation execution time.

The authors formulate two interconnected optimization problems to address this complex scenario. The first focuses on device utility optimization, where devices aim to maximize their utility by considering factors such as delay reduction, energy efficiency improvement, and offloading service cost. This optimization is subject to various constraints, including task completion time limits, server computation capacity, and task allocation decisions. The second optimization problem addresses MEC server utility, where servers seek to maximize their utility through strategic price adjustment for computing resources and efficient allocation of backup server resources, while considering operation costs and various resource constraints.

To solve these interlinked optimization problems, the authors employ a Stackelberg game approach, where MEC servers act as leaders by setting prices and allocating resources, while smart devices serve as followers by making offloading decisions based on these parameters. The solution utilizes a heuristic iterative approach where devices respond to server prices, and servers continuously optimize their strategies based on device responses until reaching an equilibrium state.

The performance results demonstrate several significant advantages of this approach. The implementation of optimal price schemes shows superior performance compared to fixed price approaches, particularly when combined with the two-layer structure incorporating a backup server. This becomes especially evident under heavy load conditions. The framework achieves substantial reductions in energy consumption, with the mobility-aware optimal offloading showing better performance than traditional graph-matching approaches. The backup server's value becomes increasingly apparent as the number of devices grows.

Regarding latency reduction, the results show that higher device speeds lead to improved performance, with the impact becoming more pronounced as the number of devices increases. The framework effectively utilizes remote resources to optimize performance across various operating conditions. This comprehensive approach represents a significant advancement in the field, successfully addressing both energy efficiency and latency concerns in MEC networks while accounting for device mobility and resource constraints. The hierarchical structure with backup servers provides the flexibility and robustness necessary for practical implementation in real-world scenarios.

This innovative framework demonstrates how careful system design and optimization can significantly improve the performance of mobile edge computing networks, particularly in scenarios involving mobile devices and varying computational demands. The approach's success in balancing energy efficiency with latency requirements while maintaining system stability makes it a valuable contribution to the field of mobile edge computing.

D. Final considerations

MEC has proven to be a cornerstone technology for enabling low-latency IoT applications. By decentralizing computation and bringing it closer to devices, MEC significantly improves efficiency, responsiveness, and resource utilization. As IoT ecosystems continue to expand, ongoing research and innovation will be essential to address emerging challenges and fully realize the potential of MEC in creating smarter, more connected systems.

VI. CONCLUSIONS

The integration of edge computing within IoT ecosystems has emerged as a transformative solution for addressing the inherent limitations of traditional cloud-centric architectures. This survey has comprehensively explored the benefits, applications, and challenges associated with leveraging edge computing for low-latency IoT scenarios, highlighting its potential to drive innovation and efficiency across diverse domains.

A. Key insights and contributions

Edge computing fundamentally enhances IoT systems by decentralizing computational processes and bringing them closer to data sources. This architectural shift mitigates network bandwidth constraints, reduces communication latency, and improves energy efficiency, making it a cornerstone for real-time applications in healthcare, smart cities, manufacturing, and beyond. By examining various paradigms such as Mobile Edge Computing (MEC), fog computing, and cloudlets, this study has underscored the paradigm's versatility and adaptability.

Edge computing addresses critical IoT requirements, including latency minimization, mobility support, and robust data security. It enables real-time interactions, seamless resource allocation, and localized data processing, fostering autonomy and reducing reliance on centralized cloud infrastructures. These capabilities empower IoT applications to achieve unprecedented levels of responsiveness and intelligence.

B. *Open research challenges and future directions*

Despite its numerous benefits, the deployment of edge computing introduces several unresolved challenges. As detailed in the reference work [1], these include:

1. **Heterogeneity**: The diversity of hardware and communication protocols across edge devices necessitates standardized programming models and interfaces.
2. **Resource Management**: Efficient allocation of computational and network resources in dynamic and resource-constrained environments remains a complex issue.
3. **Security and Privacy**: Protecting sensitive data in highly distributed environments is critical, especially as edge nodes are vulnerable to new and evolving threats.
4. **Data Abstraction and Preprocessing**: Developing efficient mechanisms for preprocessing and filtering large volumes of raw IoT data without compromising usability is a pressing need.
5. **System Availability**: Ensuring reliable and uninterrupted service delivery in edge computing ecosystems is vital for its scalability and broader adoption.

C. *Final considerations*

The advancements and challenges discussed in this study affirm the transformative potential of edge computing for IoT applications. As the IoT landscape continues to expand, further research is needed to address these challenges and optimize edge computing frameworks for broader adoption. This includes innovation in intelligent resource management, robust security protocols, and efficient integration with emerging technologies such as AI and 5G networks.

By tackling these open research challenges, edge computing can fully realize its potential as the strategic backbone of IoT ecosystems, fostering a future where intelligent, low-latency systems seamlessly connect the physical and digital worlds.

REFERENCES

[1] N. Hassan, S. Gillani, E. Ahmed, I. Yaqoob and M. Imran, "The Role of Edge Computing in Internet of Things," in IEEE Communications Magazine, vol. 56, no. 11, pp. 110-115, November 2018, doi: 10.1109/MCOM.2018.1700906.

[2] G. Premsankar, M. Di Francesco and T. Taleb, "Edge Computing for the Internet of Things: A Case Study," in IEEE Internet of Things Journal, vol. 5, no. 2, pp. 1275-1284, April 2018, doi: 10.1109/JIOT.2018.2805263.

[3] B. Chen, J. Wan, A. Celesti, D. Li, H. Abbas and Q. Zhang, "Edge Computing in IoT-Based Manufacturing," in IEEE Communications Magazine, vol. 56, no. 9, pp. 103-109, Sept. 2018, doi: 10.1109/MCOM.2018.1701231.

[4] K. Zhang, S. Leng, Y. He, S. Maharjan and Y. Zhang, "Mobile Edge Computing and Networking for Green and Low-Latency Internet of Things," in IEEE Communications Magazine, vol. 56, no. 5, pp. 39-45, May 2018, doi: 10.1109/MCOM.2018.170