Cloud-Edge computing (Cognitive Cloud)

Asror Akbarkhodjaev
University of Amsterdam, The Netherlands
a.akbarkhodjaev@student.uva.nl

ABSTRACT

The cloud-edge landscape is a hybrid computing paradigm that combines the vast resources of cloud computing with the low-latency and localized processing capabilities of edge computing. This framework is particularly effective in enhancing the performance and efficiency of Internet of Things (IoT) applications, where real-time data processing and reduced network congestion are critical. The study examines architectural designs, operational functionalities, real-world applications, and the challenges and opportunities of cloud-edge frameworks in modern computing environments. The research delves into the application domains of these frameworks and highlights open challenges and research directions, emphasizing the need for advancements in standardization, security, and resource optimization.

KEYWORDS

Systematic Literature Review, Cloud-Edge computing, Edge computing, Internet of Things (IoT)

1 INTRODUCTION

The Internet of Things (IoT) has revolutionized the way we interact with the physical world, connecting billions of devices and generating vast amounts of data. However, traditional cloud computing architectures face challenges in handling the real-time and latency-sensitive requirements of IoT applications. To address these limitations, cloud-edge computing emerged as a promising solution that brings processing and data storage closer to the edge of the network, where IoT devices are located.

To lay a solid foundation for this exploration, it is essential to first define two key concepts:

- **Cloud Computing**: A model that enables on-demand network access to a shared pool of configurable computing resources, such as networks, servers, storage, applications, and services, which can be rapidly provisioned and released with minimal management effort.
- **Edge Computing**: A distributed computing paradigm that brings computation and data closer to the edge of the network, closer to where the data originates and is consumed. This can significantly reduce latency and network congestion, and improve the performance of applications that require real-time processing or low-latency data access [29].

Cloud-edge computing is the architecture that integrates cloud computing and edge computing capabilities, enabling seamless data exchange and resource management between the two layers. This hybrid approach offers several advantages over traditional cloud computing, including reduced latency, improved scalability, enhanced security, and localized data processing.

By dissecting current research, case studies, and expert opinions, this literature study aims to provide a comprehensive overview of the cloud-edge framework, offering valuable insights for researchers, practitioners, and policymakers in the field of distributed computing. The study also delves into the application of cloud-edge frameworks in various domains. Finally, it identifies open challenges and research directions in cloud-edge computing, highlighting the need for further advancements in standardization, security, and resource optimization.

2 STUDY DESIGN

2.1 Research Goal

The research was designed to characterize comprehensively the current state of the art of A Cloud-Edge continuum. More specifically, by following the Goal-Question-Metric approach, our goal can be formalized as follows:

To comprehensively analyze and synthesize existing literature on the Cloud-Edge continuum to understand their architectural design, operational functionalities, real-world applications, and the challenges and opportunities they present in modern computing environments.

2.2 Research Questions

RQ1: What are the key architectural components and technological advancements that define a Cloud-Edge continuum?

The objective is to identify and understand the essential architectural elements and recent technological developments that constitute a Cloud-Edge computing architecture. In this paper, we investigate the fundamental components such as edge devices, network connectivity, data processing units, and cloud services. We also study how these components work in tandem, focusing on interoperability between edge devices and cloud infrastructure.

RQ2: What are the proposed solutions for the Cloud-Edge framework (Cognitive Cloud) by academia and industry?

With this question, we aim to explore and identify practical implementations of Cloud-Edge frameworks, particularly those enhanced with cognitive capabilities, proposed by both the academic world and industry leaders. This inquiry seeks to understand the solutions to integrate cloud computing with edge processing.

RQ3 How does implementing a Cloud-Edge architecture impact the security and privacy aspects of data processing and storage? What are the emerging challenges and solutions in this domain?

Edge devices often have less robust security compared to centralized cloud data centers. We investigate how this affects the overall security posture. With data traversing multiple networks, we also explore the challenges in securing data across potentially less secure networks. Lastly, we explore how the dynamic scaling of edge resources poses security and privacy challenges.

2.3 Methodology

The search for relevant articles can be split into three categories:

(1) Initial papers provided by the supervisor: These papers are used as introductory to the topic. By analyzing them, the directions of the study were set and research questions were drawn.
(2) Use online databases: IEEE Xplore, Google Scholar, ResearchGate, and ScienceDirect are used for searching and selecting publications by using specific keywords related to our research topic to get more relevant results.
(3) Snowballing approach: We are starting with a small set of initial articles or papers, and then expanding the search to include additional articles that have been cited in the initial set. Afterward, we can continue to search for additional articles that have been cited in the articles we have already identified.

The selection criteria of the articles for our research have several aspects that we take into consideration both for inclusion and exclusion of the papers.

Inclusion criteria:

(1) Relevance: Studies that focus on Cloud-Edge computing, its applications, benefits, and challenges.
(2) Peer-reviewed articles: Articles published in peer-reviewed journals and conferences.
(3) Published in the last 5 years: Studies published within the last 5 years to ensure relevance to current Cloud-Edge computing trends.

Exclusion Criteria:

(1) Non-peer-reviewed articles: Articles that have not undergone peer-review, such as blog posts, opinion pieces, and news articles.
(2) Older studies: Studies published more than 5 years ago, as they may not reflect current trends and advancements in Cloud-Edge computing.
(3) Non-English language: Studies published in languages other than English, may not be accessible to a wide audience.
(4) Duplicate studies: Studies that have been published multiple times, or studies that have been included in other literature reviews or meta-analyses.
(5) Studies with a low level of detail: Studies that lack detail or provide only a high-level overview of Cloud-Edge computing may not provide sufficient information for a comprehensive literature review.
(6) Studies that are not publicly available: Studies that are not publicly available, such as those published in proprietary publications or behind paywalls, may not be accessible to all researchers.

The total list of papers that were considered as passing our selection criteria can be retrieved in the provided Google Spreadsheet [8]. As can be seen, some articles and industry papers were initially considered relevant for this research but eventually were excluded due to the lack of eventual usefulness of the content.

3 RESULTS

In this section, the results of the literature review are reported. The comprehensive review of existing literature on cloud-edge frameworks revealed several critical areas of focus, reflecting the current state, challenges, and advancements in this field. The findings are categorized according to the research questions.

3.1 RQ1: Key enabling technologies

3.1.1 Network Functions Virtualization and Software Defined Networking.

NFV and SDN are argued to be two key enabling technologies in pushing computing, storage, and networking resources from cloud to edge [21]. In this section, we will discuss these two technologies and their role within the cloud-edge framework.

Software-defined networking (SDN): SDN is an approach to networking that uses software-based controllers or application programming interfaces (APIs) to direct traffic on the network and communicate with the underlying hardware infrastructure [31]. This separation of the control plane (which decides where traffic should go) from the data plane (which forwards traffic to the selected destination) allows for more centralized, flexible, and efficient management of network resources. SDN enables network administrators to manage network services through the abstraction of lower-level functionality, which is particularly useful in large-scale and complex networks.

Network Functions Virtualization (NFV): NFV is a concept that refers to the virtualization of network services that have traditionally been carried out by hardware appliances [13]. These functions, like firewalls, load balancers, and intrusion detection systems, are instead run as software on virtual machines. The primary goal of NFV is to transform the way network services are deployed, operated, and maintained, leading to reduced costs and increased agility in-network service deployment and operation. NFV often leverages cloud computing technologies and is closely associated with SDN, though the two can be implemented independently.

Together, SDN and NFV provide a more flexible, scalable, and efficient approach to managing network resources, leading to reduced operational costs, improved time to market for new services, and enhanced capabilities to meet changing network demands. SDN enables more efficient management of network resources by providing a centralized control plane. This is particularly important in the cloud-edge framework, where data traffic patterns can be highly dynamic due to the distributed nature of resources and services [21]. NFV plays a significant role in virtualizing network functions that are traditionally hardware-based, such as firewalls, load balancers, and routers. In a cloud-edge context, this means that these functions can be deployed quickly and flexibly at various points in the network, including at edge locations [34]. By virtualizing network functions and centralizing control with SDN, the cloud-edge framework can achieve significant cost savings and reduced complexity.

3.1.2 Automated Orchestration.

Automated orchestration in the context of information technology and cloud computing refers to the automated arrangement, coordination, and management of complex computer systems, middleware, and services. This concept is pivotal in modern IT environments, where the deployment and operation of large-scale, distributed systems are increasingly complex and dynamic. There are four key components of automated orchestration: automation, workflow management, resource management, and service coordination. Although this technology is well integrated within cloud computing, we can observe a lack of research with the cloud-edge framework [21]. In this framework, orchestration can automate the deployment and management of virtualized network functions(NFV), aligning closely with SDN (Software-Defined Networking) principles. In this context, orchestration not only manages resources within a centralized cloud infrastructure but also extends its capabilities to the edge of the network, where data processing and services are located closer to the data sources or end-users.

One of the solutions that was proposed by Muhammad Alam et al. is a lightweight, virtualization-based framework called Docker Edge Orchestration (DEO) for orchestrating microservices in IoT environments using edge computing [9]. This framework offers modularity, scalability, and fault tolerance, making it suitable for complex IoT applications. The authors demonstrate how DEO handles the packaging and deployment of microservices using Docker and evaluate its performance through experiments. They also compare it to existing solutions and conclude that DEO outperforms them in certain aspects. However, this framework for orchestration lacks automation which involves using software to manage and automate tasks that would require manual intervention. It is still arguable if the framework will be beneficial in managing large-scale, complex environments efficiently,

3.1.3 Dynamic offloading.

Dynamic offloading refers to the process of dynamically transferring tasks or computational workloads from one system or device to another, based on various factors like resource availability, network conditions, and performance requirements. This concept is particularly relevant in scenarios where devices have limited computing resources or energy constraints. Such scenarios include mobile computing, edge computing, and IoT systems. There are several challenges this technology faces:

(1) The trade-off between latency and bandwidth: While offloading can reduce latency, it might increase bandwidth usage.
(2) Network reliability: The effectiveness of dynamic offloading depends on the reliability and stability of the network connection.
(3) Data security and privacy: Offloading data to external servers can raise security and privacy concerns, necessitating robust encryption and data protection measures.
(4) Energy consumption: While offloading can save energy on one device, it may increase energy consumption overall, especially if the offloaded tasks are sent to distant cloud servers.

To address these challenges Wang et al. introduce a dynamic offloading scheduling scheme for MEC-enabled Vehicular Networks [30]. The authors proposed to use a combination of machine learning and software-defined networking techniques to dynamically determine the optimal offloading strategy for each vehicle in the network. The results show that the dynamic offloading scheduling scheme significantly outperforms traditional offloading schemes in terms of task processing efficiency, network latency, and system scalability.

3.2 RQ2: Frameworks proposed by academia

Cloudlet is one of the key concepts in edge computing architectures [26]. They provide a platform for running applications and performing data processing tasks at the edge of the network, closer to IoT devices or users. More specifically, it is a small-scale cloud computing environment that is designed to provide cloud computing resources and services to a specific group of users or organizations. The concept of cloudlets is designed to bring cloud computing capabilities closer to the user, thereby reducing latency, improving response time, and conserving bandwidth. Cloudlet is designed to augment the computational and storage capabilities of mobile devices [12]. Offloading heavy computing tasks from mobile devices to cloudlets helps in achieving better performance and conserving the battery life of the devices. Essentially there are three ways for the offloading and coordination between mobile devices and the "Cloudlets": a Virtual Machine (VM) based approach, process migration, or software virtualization [21]. Popular examples of cloudlets include OpenStack++ [15], AWS Outposts [2], and Azure Stack [3].

Fog computing is a concept introduced and promoted by Cisco as an extension of cloud computing [19]. It involves bringing the computing, storage, and networking services closer to the end devices, or the "edge" of the network. In the paper by Mandeep Kaur et al., it was stated that "fog computing" is a resourceful layer placed between the cloud and edge device [25]. The idea is to minimize latency, reduce bandwidth use, and improve the efficiency and effectiveness of data processing. In Cisco's implementation of fog computing, data is processed within a fog node or IoT gateway located close to where it is generated, rather than being sent to a distant cloud data center. This approach is particularly beneficial for applications requiring real-time or near-real-time processing and for managing the vast amount of data generated by IoT devices. Cisco's fog computing model provides a more distributed framework for managing and processing data, which is crucial for the success of IoT and other advanced technologies that require rapid processing of large volumes of data at the network's edge.

Mobile-Edge Computing (MEC) Initiative The Mobile-Edge Computing (MEC) Initiative, launched by the European Telecommunications Standards Institute (ETSI), focuses on integrating cloud computing capabilities into the mobile network at the edge, close to mobile subscribers [14]. The initiative aims to reduce latency, ensure efficient network operation and service delivery, and improve user experience by bringing computational resources nearer to the end-users. MEC supports a wide range of applications and services, including content caching, IoT, location-based services, and video analytics, all processed at the network's edge. Arif Ahmed et al. provided a comprehensive overview of mobile edge computing (MEC) [7]. They argue that this approach can significantly improve the performance, responsiveness, and security of mobile applications, as well as enable new classes of applications that were not previously feasible. The authors also highlight the most recent research efforts in MEC. Some of these research include:

- FemtoClouds are a decentralized type of edge computing architecture and can dynamically adapt to changing network conditions and user demands.
- REPLISOM is a framework for efficient resource allocation and management in FemtoClouds. It leverages machine learning techniques to predict mobile device availability and resource requirements, enabling optimized task offloading decisions.
- ME-VOLTE, or Multi-Access Edge Virtualization over Long-Term Evolution, is an architecture that extends the capabilities of FemtoClouds to support seamless connectivity and collaboration among multiple FemtoClouds. It plays a crucial role in enabling large-scale edge computing deployments.
- Multi-user computation Offloading is a technique that enables the offloading of computationally intensive tasks from one or more mobile devices to a shared FemtoCloud infrastructure. This approach optimizes resource utilization and improves the overall performance of mobile applications.

Jinke Ren et al. investigate the collaboration between cloud computing and edge computing, where the tasks of mobile devices can be partially processed at the edge node and at the cloud server [24]. The authors first formulate a joint communication and computation resource allocation problem to minimize the weighted-sum latency of all mobile devices within a multi-user MEC system. They then derive the closed-form optimal task-splitting strategy as a function of the normalized backhaul communication capacity and the normalized cloud computation capacity. Based on the optimal task-splitting strategy, the researchers further transform the original joint communication and computation resource allocation problem into an equivalent convex optimization problem and obtain the closed-form computation resource allocation strategy by leveraging the convex optimization theory. Moreover, they develop a necessary condition to judge whether a task should only be processed at the corresponding edge node, without offloading to the cloud server. According to the results, the implemented framework provides a new and effective way to reduce latency in cloud computing systems. The proposed collaborative cloud and edge computing scheme is shown to be significantly more efficient than conventional schemes.

The paper by Koustabh Dolui et al. provides a comprehensive overview of the three main types of edge computing implementations. It discusses the architecture, deployment, and use cases of each implementation [12]. It also compares the performance, scalability, and reliability of each implementation. The authors argue the choice of edge computing implementation depends on the specific requirements of the application. Fog computing is a good choice for applications that require high scalability and can tolerate some latency. Cloudlet computing is a good choice for applications that require low latency and can tolerate some data loss. Mobile edge computing is a good choice for applications that require very low latency and high reliability. Table 1 compares these three types of edge computing implementations.

Central Office Re-architected as a Datacenter (CORD) is a project that combines Network Functions Virtualization (NFV) and Software Defined Networking (SDN) technologies with the elasticity of commodity clouds to bring data center economics and cloud agility to the Telco Central Office [21]. Larry Petterson et al. proposed this idea in the collaborative project between AT&T and ON.Lab [22]. According to them, CORD transforms Central Offices into agile service delivery platforms, capable of sustaining the increasing demands of data and network services. This transformation enables the deployment of more efficient and flexible network services, leveraging open hardware and software frameworks to build and manage virtual network functions. Researchers proposed architecture consists of three layers:

(1) The edge layer: This layer is responsible for connecting users and devices to the network. It includes radio access networks (RANs), fiber-to-the-home (FTTH) networks, and other access technologies.
(2) The core layer: This layer is responsible for transporting data between the edge and the data center. It includes high-capacity optical transport networks, packet-optical transport networks, and software-defined wide-area networks (SD-WANs).
(3) The data center layer: This layer is responsible for hosting and processing data. It includes cloud computing infrastructure, such as servers, storage, and networking equipment.

The authors argue that this new architecture offers several benefits, including improved scalability, reduced latency, increased resource utilization, and better support for data-intensive applications and services.

Nebula is a framework designed to support data-intensive computing applications at the edge of the network [18]. Nebula aims to distribute computing resources across multiple edge locations to process large volumes of data closer to where it is generated. The concept aligns with the broader trends in edge computing and distributed cloud architectures. Nebula makes use of voluntary resources for computation and data storage in its architecture by allowing nodes to join the system as volunteers. These volunteer nodes can donate their computation and storage resources to carry out tasks on behalf of applications. For computation, Nebula utilizes volunteer compute nodes that execute tasks within a secure sandbox provided by the Google Chrome browser's Native Client (NaCl). This sandboxing technology ensures the safety of the volunteer nodes by restricting privileges and isolating faults from the rest of the system. Users can easily join Nebula by enabling the NaCl plugin in their Chrome browser, without the need to download additional software. In terms of data storage, Nebula employs a DataStore service that supports efficient and location-aware storage. Each application has its own DataStore, consisting of data nodes that store the actual data and a DataStore Master that manages the storage system metadata and data placement decisions. Data in a DataStore is organized in files and can be accessed and processed by the applications.

Edge-centric IoT architecture was proposed by Kewel Sha et al.[28]. It consists of four main components: the cloud, IoT end devices, the edge, and users. This architecture considers the resources and specific features of each component. Users interact with IoT end devices through interfaces provided by the cloud or the edge, using intelligent IoT applications for convenience. IoT end devices are embedded in the physical world for sensing and controlling but lack computational power. The cloud offers vast resources but is often

Table 1: Comparison of edge computing implementations

Feature	**Fog Computing**	**Cloudlet Computing**	**Mobile Edge Computing**
Layer	Fog layer	Cloud	Network
Deployment	Distributed	Clusters	Integrated with mobile network
Target devices	IoT devices, mobile devices	Mobile devices	Mobile devices
Use cases	Real-time data processing, location-based services, virtual reality	Mobile gaming, augmented reality, video streaming	Mobile gaming, augmented reality, video streaming

physically distant from end devices, leading to inefficiencies in real-time systems. The edge is central to this architecture, coordinating all parties and enhancing performance by complementing both the cloud and end devices. In this system, user queries and commands are processed at the edge layer, either directly or forwarded to IoT end devices. The edge serves as a bridge between users, devices, and the cloud, managing data storage and computational tasks. Services previously cloud-based can now be shifted to the edge, which can operate independently or in conjunction with the cloud. The edge can handle all IoT application needs or collaborate with the cloud for more complex tasks like deep learning. The architecture optimizes real-time responses and computational offloading for end devices. The edge is ideal for deploying IoT security solutions due to its resource availability, proximity to end devices, and data collection capabilities, making it effective for security decisions and intrusion detection. It resolves potential conflicts in network policies and is more feasible for firewall deployment than individual end devices. The edge tracks device mobility, maintains secure connections and trust, and can leverage the cloud for additional security support. The architecture thus presents an effective solution for IoT security challenges.

3.3 RQ2: Solutions from industry

AWS offers a comprehensive suite of edge services that focus on delivering data processing, analysis, and storage close to where data is generated [1]. AWS allows you to build an application once and deploy it both on the cloud and at the edge. It extends infrastructure, services, APIs, and tools available in the cloud as a fully managed service to virtually any on-premises data center or co-location space. It also provides managed hardware deployed in locations outside its data centers, bringing secure edge computing capabilities to metro areas, 5G networks, on-premises locations, and ruggedized devices. This gives AWS the largest global infrastructure footprint of any provider.

Additionally, the company offers a variety of cloud-to-edge solutions designed to extend cloud computing capabilities to the edge of the network. One of these solutions is AWS Greengrass. It allows you to run local computing, messaging, data caching, sync, and ML inference capabilities for connected devices in a secure way. The research conducted by Jiedong Bi et al. outlines the basic architecture of AWS IoT Greengrass and its benefits, such as real-time response to local events, offline operation capabilities, and secure communication [10]. Researchers also discuss how AWS IoT Greengrass can reduce the cost of running IoT applications by leveraging its unique architectural features.

Microsoft Azure provides a unified cloud-to-edge platform designed to extend the vast capabilities of Azure's cloud services to the edge of the network [4]. This integration allows seamless deployment and management of applications and services across different environments, from global data centers to local edge sites.

Azure IoT Edge is a service that allows you to run cloud intelligence directly on IoT devices and is scalable across devices. David Jansen in his book argues that Azure IoT Edge is a powerful platform for extending the cloud to the intelligent edge [17]. The book covers the fundamentals of IoT Edge, including its architecture, components, and deployment options. It also delves into the development of custom IoT Edge modules, edge-to-cloud communication, and security considerations. Moreover, David Jensen extensively discusses the security of Azure IoT Edge. He highlights the fundamental security principles and best practices that should be implemented to safeguard IoT Edge solutions. His guidelines can help developers and architects build secure and resilient IoT Edge solutions.

Google Cloud is at the forefront of innovation in edge computing, providing a range of solutions that offer agility, scalability, and security [6]. Their portfolio, from Anthos for Edge to Confidential Computing, addresses the needs of various industries, enabling them to leverage edge computing for improved operational efficiency and enhanced customer experiences. With the Google Edge Network, businesses can deliver content and services with low latency and high bandwidth, showcasing Google Cloud's commitment to providing top-tier solutions for edge computing.

One of the cloud-to-edge solutions from Google Cloud is Cloud IoT Core. It allows you to securely connect and manage IoT devices. It provides a centralized platform for IoT devices to communicate with each other and with the cloud, enabling you to collect, process, and analyze data from your devices. Cloud IoT Core offers edge computing capabilities, allowing you to process data closer to the source, reducing latency, and improving real-time processing [5]. Crisgar et al. in their paper showed the usage of this technology by creating a system that tracks and detects theft of vehicles using GPS technology and Google Cloud IoT Core and Firebase services [11]. The system uses GPS technology to track the location of the vehicle and Google Cloud IoT Core to process and analyze the GPS data.

3.4 RQ3: Security

The Cloud-Edge architecture, which combines cloud computing's powerful centralized resources with edge computing's localized data processing, offers numerous benefits in terms of efficiency, speed, and scalability. However, this hybrid model also introduces unique security challenges that need to be addressed. Here are the main security challenges in the Cloud-Edge architecture:

- *Data Security and Privacy.* Data transit between edge devices, edge servers, and the cloud is vulnerable to interception and tampering. Additionally, data stored on edge devices or edge servers are less secure than data stored in a cloud data center [29].
- *Network Security.* The multitude of connected edge devices significantly enlarges the attack surface. The network is also required to be reliable since interruptions in network connectivity can impact the effectiveness of security measures like real-time monitoring and updates.
- *Authentication and Access Control.* Edge devices often process sensitive personal data, necessitating stringent privacy controls. Ensuring secure authentication for a large number of devices and users is complex.

Addressing these security challenges requires a comprehensive approach that includes robust encryption, secure network protocols, regular security updates and patch management, advanced threat detection and response systems, and stringent access controls. In the following parts of this section, we look into the proposed solutions to tackle these challenges within the cloud-edge framework.

Kewei Sha et al. discuss comprehensive security architectures at the edge layer for IoT applications [28]. These architectures aim to address the security challenges of IoT devices by offloading security protection to the edge layer. Three major categories of edge-based security architectures are identified: user-centric, device-centric, and end-to-end security. *User-centric security architectures* focus on establishing a trusted domain at the edge layer to manage security for users accessing IoT resources from various devices. *Device-centric security architectures* prioritize securing individual IoT devices by deploying security mechanisms at the edge layer. *End-to-end security architectures* aim to provide holistic security solutions across the entire IoT system. These edge-based security architectures leverage the resources and proximity of the edge layer to enhance security and optimize system performance.

3.4.1 User-centric security architectures.

The user-centric edge-based IoT security architecture focuses on ensuring user satisfaction and managing security at the edge layer. This approach recognizes that users may not always access IoT applications from trusted and secure devices and may lack knowledge in effectively managing security. To address these concerns, the edge layer is responsible for managing security for each specific user. This architecture establishes a trusted domain at the edge layer, where users connect to access IoT resources from various devices. By offloading personal security to the network edge and implementing virtualized security at the network edge, user-centric edge-based IoT security designs aim to provide a secure and convenient user experience. Rahman et al. have proposed a new solution that adopts user-centric security architecture [23]. FogTrust is a framework for enforcing fine-grained data access control in IoT networks. It uses a distributed trust management system to manage access control policies, ensuring that only authorized devices can access sensitive data. FogTrust also provides secure data aggregation, allowing data to be aggregated at the edge without compromising user privacy.

3.4.2 Device-centric security architectures.

The focus of the device-centric edge-based IoT security architecture is to customize security solutions for each individual end device based on its available resources, the sensitiveness of the sensing data, and the impact of actuating tasks. It aims to offload security functions from IoT devices to the edge layer. One such representative design is EdgeSec.

EdgeSec is a new edge computing security solution introduced by Ranadheer Errabelly et al. [27]. It is a security service designed to enhance the security of IoT systems at the Edge layer. It consists of seven major components that work together to address specific security challenges in IoT systems.

1. Security Profile Manager (SPM): SPM registers IoT devices to EdgeSec and creates a security profile for each device. It also collects the security requirements of each device.
2. Security Analysis Module (SAM): SAM analyzes the capabilities and security requirements of each device. Based on this analysis, SAM decides where to deploy the security functions, whether at the Things layer, the Edge layer, or the Cloud layer.
3. Protocol Mapping (PM): PM chooses appropriate protocols to satisfy the security requirements based on the decisions made by SAM.
4. Interface Manager (IM): IM masks the communication heterogeneity in IoT devices, allowing for seamless communication between devices at different layers.
5. Security Simulation Module (SSM): SSM evaluates the potential impact of actuation requests by assessing the possible consequences. This helps prevent requests that may cause physical damage.
6. Request Handler (RH): RH handles the requests from IoT devices, ensuring that they are processed securely and efficiently.
7. User Interface (UI): UI provides a user-friendly interface for users to interact with EdgeSec and monitor the security status of their IoT devices.

By deploying EdgeSec at the Edge layer, IoT systems can benefit from increased resources, direct communication with devices, and better real-time performance.

3.4.3 End-to-end security architectures.

The focus of the end-to-end edge-based IoT security architecture is to ensure secure communication between IoT devices and between IoT devices and the cloud. This architecture aims to address the challenge of achieving end-to-end security in the IoT, considering the heterogeneity of devices. The edge layer acts as a bridge, connecting the diverse IoT devices and the cloud, and a secure middleware is deployed at the edge layer to manage security functions and enable secure end-to-end communications.

Blockchain technology integration into a Cloud-Edge computing architecture became one of the novel paradigms [20]. From the theoretical point of view, it offers several advantages for network security, especially when it comes to data transfer between devices on the edge and also between the edge and the cloud. However, we still observing the early stages of the practical implementation of such a system.

Hou et al. proposed a blockchain-based architecture for secure and efficient data sharing in IoT networks [16]. This architecture addresses the security and privacy challenges of traditional IoT data-sharing systems by leveraging the immutability and transparency of blockchain technology. The architecture utilizes multiple state chains to support both real-time and historical data sharing. The real-time chain facilitates fast data distribution, while the historical chain provides a tamper-proof record of data provenance. The authors employ identity-based aggregate signature schemes to reduce the size and communication overhead of signatures. This is particularly beneficial for IoT networks with limited resources and bandwidth. Moreover, the architecture uses blockchain to store IoT data in a tamper-proof and distributed manner. This ensures that data can only be modified with the consent of all network participants, preventing unauthorized access and tampering.

As a result, several benefits are achieved from the proposed architecture. Blockchain's immutability and distributed nature safeguard IoT data from unauthorized modification or deletion. We also can observe improved data privacy because identity-based aggregate signatures protect user privacy by obscuring data ownership and reducing signature sizes. Multiple state chains and efficient consensus algorithms ensure data consistency and facilitate real-time data sharing. Lastly, the decentralized nature of blockchain eliminates the need for a single trusted authority, enhancing resilience and fault tolerance.

4 DISCUSSION

The future trends in Cloud-Edge computing frameworks are expected to be driven by ongoing technological advancements and evolving industry needs. We believe the distinction between cloud and edge computing will continue to blur as hybrid solutions become more prevalent. These solutions will leverage the strengths of both cloud and edge computing, offering more versatile and robust computing frameworks. Due to strong interest from the industry, the concept of Edge-as-a-Service will likely gain traction, offering businesses and organizations flexible and scalable edge computing capabilities without the need for significant upfront investment in infrastructure.

We also can see that as Cloud-Edge computing cuts across various industries, there will be an increased need for cross-industry collaborations and standardization of protocols and interfaces. This will ensure compatibility and interoperability among different systems and devices. However, at the current stage, the lack of standardization in Cloud-Edge computing is a significant challenge that impacts various aspects of this technology. Without standardization, interoperability between different cloud and edge computing systems can be problematic. Devices, platforms, and services from different vendors may not work seamlessly together, leading to compatibility issues. Moreover, scaling Cloud-Edge solutions in a non-standardized environment can be challenging. To address these challenges, there is a growing call for industry-wide standardization efforts. This includes developing common protocols, security standards, and interoperability frameworks.

Another challenge that needs to be addressed in the field of Cloud-Edge computing is enhanced security and privacy measures. As Cloud-Edge frameworks become more widespread, the need for advanced security and privacy measures will increase. Future trends will likely include the development of more sophisticated encryption methods, secure data transmission protocols, and privacy-preserving computation techniques to protect sensitive data at the edge.

5 SUMMARY OF RESULTS

RQ1: *What are the key architectural components and technological advancements that define a Cloud-Edge continuum?*

In this article we explored key technologies that enable a Cloud-Edge continuum, where computing and storage happen closer to users and devices, making things faster and more responsive. Here's a quick rundown of the main ideas:

- *SDN and NFV* make networks more flexible and efficient by separating control and data, and virtualizing network functions. This means faster setup, easier management, and lower costs.
- *Automated orchestration* manages complex systems like cloud-edge deployments, automatically scaling and arranging resources efficiently.
- *Dynamic offloading* moves tasks from devices with limited resources to powerful systems for faster processing, improving performance for things like mobile apps and internet-of-things devices.

These technologies are powerful, but they also come with challenges like balancing speed with data transfer, ensuring security and privacy, and managing energy use.

RQ2: *What are the proposed solutions for the Cloud-Edge framework (Cognitive Cloud) by academia and industry?*

Cloud-Edge computing frameworks proposed by academia:

- *Cloudlets*: Small-scale cloud environments near users or devices, reducing latency and conserving bandwidth. Examples include OpenStack++, AWS Outposts, and Azure Stack.
- *Fog computing*: Brings computing closer to end devices, minimizing latency and managing the vast amount of data from IoT devices. Cisco is a key player in this area.
- *Mobile-Edge Computing (MEC)*: Integrates cloud capabilities into the mobile network, improving performance and enabling new applications. Examples include FemtoClouds, REPLISOM, and ME-VOLTE.
- *Central Office Re-architected as a Datacenter (CORD)*: Combines NFV, SDN, and cloud elasticity to bring agility and efficiency to network services.
- *Nebula*: Framework for data-intensive computing at the edge, utilizing volunteer resources for computation and data storage.
- *Edge-centric IoT architecture*: Optimizes real-time responses and computational offloading for end devices by placing user queries and services closer to the edge.

Cloud-Edge computing frameworks proposed by the industry:

- *AWS*: Offers a large suite of edge services, including AWS Greengrass for running local computing and ML capabilities on devices.

- *Microsoft Azure*: Provides a unified cloud-to-edge platform with Azure IoT Edge for running cloud intelligence directly on devices.
- *Google Cloud*: Offers various solutions like Anthos for Edge and Cloud IoT Core, enabling edge computing for improved operation and customer experiences.

RQ3: *How does implementing a Cloud-Edge architecture impact the security and privacy aspects of data processing and storage? What are the emerging challenges and solutions in this domain?*

Cloud-edge computing combines the power of centralized cloud resources with the localized processing of edge devices. While this offers speed and efficiency, it also brings unique security and privacy concerns:

- Data vulnerability: Data traveling between devices, servers, and the cloud can be intercepted or tampered with. Edge devices might also store data less securely than cloud data centers.
- Network security: Many connected devices increase the attack surface. Reliable network connections are crucial for real-time security measures like monitoring and updates.
- Authentication and access control: Securing sensitive data on edge devices requires strong authentication and access controls, challenging for numerous devices and users.

Some solutions that are designed to tackle these challenges:

- Comprehensive security: Robust encryption, secure network protocols, regular updates, advanced threat detection, and strict access control are essential.
- Edge-based security architectures: These architectures address security challenges directly on the edge, including:
  - User-centric: Manages security for users accessing resources from various devices, offloading personal security to the network edge.
  - Device-centric: Customizes security for each device based on resources, data sensitivity, and tasks.
  - End-to-end: Ensures secure communication between devices and the cloud through a secure edge middleware.
- Blockchain: This technology offers enhanced security and privacy for data transfer between devices and the cloud.

6 CONCLUSION

Cloud-edge computing is an emerging distributed computing paradigm that extends cloud computing capabilities closer to the edge of the network, where IoT devices and data sources are located. This architecture aims to address the limitations of traditional centralized cloud computing, such as latency, scalability, and security. However, integrating edge computing with existing cloud infrastructure and legacy systems can be challenging and require careful planning.

To conduct a thorough literature review on the Cloud-Edge computing architecture, comprehensive literature searches were conducted using academic databases and search engines. Relevant studies were identified based on keywords such as "cloud-edge framework", "IoT", and "edge computing".The research included a variety of sources, including peer-reviewed journal articles, conference papers, industry reports, and technical white papers. The studies were selected based on their relevance to the research topic and their methodological rigor. The literature was analyzed primarily using qualitative methods to identify key trends, patterns, and insights. Finally, the findings were presented in a structured and comprehensive manner, summarizing the state-of-the-art research on Cloud-Edge computing.

Managing multiple edge nodes and orchestrating data flows between edge and cloud can introduce complexity and operational overhead. Automated orchestration seems to be one of the promising enabling technologies in the Cloud-Edge architecture. This technology is well integrated within centralized cloud computing, but we were not able to find a viable solution for its implementation for edge computing. We believe this area of Cloud-Edge computing should be further examined by both academia and the industry.

The are several solutions for the implementation of the Cloud-Edge continuum proposed by academia and the industry. Here we should point out three powerful and potentially highly successful architectures: cloudlet computing, for computing, and mobile-edge computing (MEC). According to our research, these three frameworks dominate the research in the area of cloud-edge computing.

It can be argued that the Cloud-Edge architecture brings enhanced security, but also several challenges. By processing and storing data at the edge, sensitive information is less susceptible to cyberattacks, and data privacy is improved by minimizing the amount of data transferred to the cloud. At the same time, edge nodes may be more vulnerable to cyberattacks due to their distributed nature and potential lack of centralized security protocols. In this paper, we have shown several solutions to address the challenges of the security of this hybrid framework.

Almost all industry leaders that provide cloud computing platforms provide comprehensive cloud-to-edge solutions designed to extend cloud computing capabilities to the edge. There are several studies done on the implementation of these services in different IoT application domains, such as healthcare, smart cities, and industrial automation. However, we can observe the lack of research on extensive comparisons between AWS Greengrass, Azure IoT Edge, Cloud IoT Core, and others. This study will provide valuable insights and information to help organizations make informed decisions about their IoT solutions, such as understanding the similarities and differences between the three platforms, identifying the strengths and weaknesses of each platform, and determining which platform best fits their specific needs and requirements.

REFERENCES

[1] [n. d.]. AWS for the Edge. https://aws.amazon.com/edge/. Accessed: 2023-11-28.

[2] [n. d.]. AWS Outposts. https://aws.amazon.com/outposts/. Accessed: 2023-12-03.

[3] [n. d.]. Azure Stack. https://azure.microsoft.com/en-us/products/azure-stack. Accessed: 2023-12-03.

[4] [n. d.]. Azure: What is edge computing? https://azure.microsoft.com/en-gb/resources/cloud-computing-dictionary/what-is-edge-computing/. Accessed: 2023-11-28.

[5] [n. d.]. Google Cloud Internet of Things. https://cloud.google.com/iot-core. Accessed: 2023-11-29.

[6] [n. d.]. Google Distributed Cloud Edge. https://cloud.google.com/distributed-cloud-edge?hl=en. Accessed: 2023-11-28.

[7] Arif Ahmed and Ejaz Ahmed. 2016. A survey on mobile edge computing. In *2016 10th International Conference on Intelligent Systems and Control (ISCO)*. 1–8. https://doi.org/10.1109/ISCO.2016.7727082

[8] Asror Akbarkhodjaev. [n. d.]. Reviewed Articles. https://docs.google.com/spreadsheets/d/1m1FtLyRoFzhhv3BndS9S0yyD1R04XKrFq1yjv4saTHg/edit?usp=sharing. Accessed: 2023-12-22.

[9] Muhammad Alam, Joao Rufino, Joaquim Ferreira, Syed Hassan Ahmed, Nadir Shah, and Yuanfang Chen. 2018. Orchestration of Microservices for IoT Using

Docker and Edge Computing. *IEEE Communications Magazine* 56, 9 (2018), 118–123. https://doi.org/10.1109/MCOM.2018.1701233

[10] Jiedong Bi, Xinchang Zhang, and Wenpeng Cao. 2021. Comparative Research on Edge Computing System. In *Proceedings of the 2021 International Conference on Modern Management and Education Research (MMER 2021)*. Atlantis Press, 158–163. https://doi.org/10.2991/assehr.k.210915.036

[11] Puji Valen Crisgar, Patrick Ryan Wijaya, Marcell D. F. Pakpahan, Eniman Yunus Syamsuddin, and Muhammad Ogin Hasanuddin. 2021. GPS-Based Vehicle Tracking and Theft Detection Systems using Google Cloud IoT Core Firebase. In *2021 International Symposium on Electronics and Smart Devices (ISESD)*. 1–6. https://doi.org/10.1109/ISESD53023.2021.9501928

[12] Koustabh Dolui and Soumya Kanti Datta. 2017. Comparison of edge computing implementations: Fog computing, cloudlet and mobile edge computing. In *2017 Global Internet of Things Summit (GIoTS)*. 1–6. https://doi.org/10.1109/GIOTS.2017.8016213

[13] European Telecommunications Standards Institute. 2012. *Network Functions Virtualisation (NFV); An Introduction, Benefits, Enablers, Challenges Call for Action.* Technical Report. ETSI. https://docbox.etsi.org/isg/nfv/open/Publications_pdf/White%20Papers/NFV_White_Paper1_2012.pdf

[14] European Telecommunications Standards Institute. 2023. Multi-access Edge Computing - Standards for MEC. https://www.etsi.org/technologies/multi-access-edge-computing Accessed: 2023-12-04.

[15] Kiryong Ha and Mahadev Satyanarayanan. 2015. *Openstack++ for cloudlet deployment*. Technical Report 2014.

[16] Mingyu Hou, Tianyu Kang, and Li Guo. 2020. A Blockchain Based Architecture for IoT Data Sharing Systems. In *2020 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*. 1–6. https://doi.org/10.1109/PerComWorkshops48775.2020.9156107

[17] Daniel Jensen. 2022. *Beginning Azure IoT Edge Computing: Extending the Cloud to the Intelligent Edge*. Apress, Berkeley, CA.

[18] Albert Jonathan, Mathew Ryden, Kwangsung Oh, Abhishek Chandra, and Jon Weissman. 2017. Nebula: Distributed Edge Cloud for Data Intensive Computing. *IEEE Transactions on Parallel and Distributed Systems* 28, 11 (2017), 3229–3242. https://doi.org/10.1109/TPDS.2017.2717883

[19] Author(s) Names. 2012. Title of the Paper. In *Proceedings of the ACM SIGCOMM 2012 Conference*. ACM, Location of the Conference, Page Numbers. https://doi.org/DOINumberifavailable

[20] Dinh C. Nguyen, Pubudu N. Pathirana, Ming Ding, and Aruna Seneviratne. 2020. Integration of Blockchain and Cloud of Things: Architecture, Applications and Challenges. *IEEE Communications Surveys Tutorials* 22, 4 (2020), 2521–2549. https://doi.org/10.1109/COMST.2020.3020092

[21] Jianli Pan and James McElhannon. 2018. Future Edge Cloud and Edge Computing for Internet of Things Applications. *IEEE Internet of Things Journal* 5, 1 (2018), 439–449. https://doi.org/10.1109/JIOT.2017.2767608

[22] Larry Peterson, Ali Al-Shabibi, Tom Anshutz, Scott Baker, Andy Bavier, Saurav Das, Jonathan Hart, Guru Palukar, and William Snow. 2016. Central office re-architected as a data center. *IEEE Communications Magazine* 54, 10 (2016), 96–101. https://doi.org/10.1109/MCOM.2016.7588276

[23] Abdul Rehman, Kamran Ahmad Awan, Ikram Ud Din, Ahmad Almogren, and Mohammed Alabdulkareem. 2023. FogTrust: Fog-Integrated Multi-Leveled Trust Management Mechanism for Internet of Things. *Technologies* 11, 1 (2023), 27.

[24] Jinke Ren, Guanding Yu, Yinghui He, and Geoffrey Ye Li. 2019. Collaborative Cloud and Edge Computing for Latency Minimization. *IEEE Transactions on Vehicular Technology* 68, 5 (2019), 5031–5044. https://doi.org/10.1109/TVT.2019.2904244

[25] Mandeep Kaur Saroa and Rajni Aron. 2018. Fog Computing and Its Role in Development of Smart Applications. In *2018 IEEE Intl Conf on Parallel Distributed Processing with Applications, Ubiquitous Computing Communications, Big Data Cloud Computing, Social Computing Networking, Sustainable Computing Communications (ISPA/IUCC/BDCloud/SocialCom/SustainCom)*. 1120–1127. https://doi.org/10.1109/BDCloud.2018.00166

[26] Mahadev Satyanarayanan, Paramvir Bahl, Ramon Caceres, and Nigel Davies. 2009. The Case for VM-Based Cloudlets in Mobile Computing. *IEEE Pervasive Computing* 8, 4 (2009), 14–23. https://doi.org/10.1109/MPRV.2009.82

[27] Kewei Sha, Ranadheer Errabelly, Wei Wei, T. Andrew Yang, and Zhiwei Wang. 2017. EdgeSec: Design of an Edge Layer Security Service to Enhance IoT Security. In *2017 IEEE 1st International Conference on Fog and Edge Computing (ICFEC)*. 81–88. https://doi.org/10.1109/ICFEC.2017.7

[28] Kewei Sha, T. Andrew Yang, Wei Wei, and Sadegh Davari. 2020. A survey of edge computing-based designs for IoT security. *Digital Communications and Networks* 6, 2 (2020), 195–202. https://doi.org/10.1016/j.dcan.2019.08.006

[29] Weisong Shi, Jie Cao, Quan Zhang, Youhuizi Li, and Lanyu Xu. 2016. Edge Computing: Vision and Challenges. *IEEE Internet of Things Journal* 3, 5 (2016), 637–646. https://doi.org/10.1109/JIOT.2016.2579198

[30] Hansong Wang, Xi Li, Hong Ji, and Heli Zhang. 2018. Dynamic Offloading Scheduling Scheme for MEC-enabled Vehicular Networks. In *2018 IEEE/CIC International Conference on Communications in China (ICCC Workshops)*. 206–210. https://doi.org/10.1109/ICCChinaW.2018.8674508

[31] V. B. Wijekoon, T. M. Dananjaya, P. H. Kariyawasam, S. Iddamalgoda, and Ajith Pasqual. 2016. High performance flow matching architecture for OpenFlow data plane. In *2016 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*. 186–191. https://doi.org/10.1109/NFV-SDN.2016.7919496