WCRG Global Research WORLD COMET RESEARCH GROUP

International Journal of Modern Innovations and Emerging Trends
Vol. 4, No. 1, 2021
Doi: 10.67228/30715628/IJMIET-2021PI9Y6L
PP. 01-14

Original Article

Edge Computing Architectures for Ultra-Low Latency Applications

Dr. Priya Natarajan

Associate Professor, University of Madras, India.

Received: 04-03-2021 Revised: 04-23-2021 Accepted: 05-01-2021 Published: 05-03-2021

ABSTRACT

The high rate of increase in the number of latency-sensitive services, including autonomous driving, remote surgery, augmented reality, industrial automation, and smart grids, has outlined the weaknesses of conventional cloud computing models. Although advantageous and scalable, centralized cloud data centers involve a high arguably serious latency because of network impacts, lengthy transmission paths, and delays while processing data. Edge computing has become a new paradigm that pulls the services of computation, storage, and intelligence back to the data source hence, allowing the creation of ultra-low latency services and real-time decision-making. This research paper is a detailed analysis of the edge computing architecture of ultra-low latency applications. The work explores the transformation of centralized cloud to distributed edge cloud ecosystems, examines key supporting technologies, 5G, software defined networking (SDN), network function virtualization (NFV), and containerized micro-services, and discusses such performance metrics as latency, throughput, reliability, and energy efficiency. It is being proposed that hierarchical edge computing architecture is based on integrating device edge, access edge, and regional edge layers in order to deploy them in a manner that is scaled and resilient. It is followed by the detailed literature survey analysis to evaluate the state-of-the-art architectures, schedule mechanisms, orchestration frameworks, as well as optimization techniques. In addition, there is also a methodological framework of the latency-aware task offloading, resource allocation, and service placement. The evaluation of the proposed architecture through analytical models and simulation-based evaluations shows that the proposed architecture can reduce the latency onto 65 percent of the time relative to the conventional cloud-based architecture. These findings affirm the claim that edge computing is an essential enabler of the next-generation applications that demand deterministic response time, high reliability and localized intelligence. A systematic architectural structure, performance discussion, and implementation of a guideline concerning the creation of subsequent ultra-low latency systems are presented in this paper.

KEYWORDS

Edge Computing, Ultra-Low Latency, 5G Networks, Internet of Things (IoT), Real-Time Systems, Cloud Computing, Distributed Systems, Network Virtualization, Task Offloading, Smart Cities.

WCRG Global Research
ISSN: 3071-5628
PP: 01-14.

1. INTRODUCTION

1.1. Background

The high-speed development of the Internet of Things (IoT) devices, virtual-physical systems and real-time digital services has changed the computing paradigm and application demands radically today. The demands of the modern emerging applications, such as autonomous vehicles, telemedicine, industrial robotics, smart transport, and immersive media platforms, require extremely low latency, high reliability, and input-to-output capabilities of large amounts of data. Such applications are used in dynamic and frequently safety-critical environments where any slight delays or disruptions in services may severely affect performance or result into severe outcomes. Although standard cloud computing models can offer virtually unlimited computing resources and storage, their centralized design presents an inevitable latency delay, network capacity issues and bandwidth limits, which are becoming unsustainable in supporting ultra-low-latency and real-time workloads. Edge computing has become an interesting alternative paradigm to resolve these shortcomings by moving the computing, storage, and intelligence nearer to the data generation points. As opposed to producing and transmitting all the data to remote data centers of the clouds, edge computing allows processing at the edge servers, gateways, and access nodes. This goes a long way to minimize the physical and logical distance between end users and computation resources, and causes a shortening of response time and enhances service reliability plus a reduction in backbone network traffic. Edge computing offers a scalable and efficient solution to next-generation cyber-physical systems and intelligent infrastructures by offering localized analytics, enabling real-time decision making, and workload distribution, as well as workload distribution. Consequently, edge computing is being adopted as an important enabling technology of the future of intelligent, connected, and autonomous systems.

1.2. Importance of Edge Computing Architectures

ECA designs serve a crucial position in enabling intelligent applications of the next generation through the weaknesses of conventional centralized computing designs. With the growing use of digital systems executing optimal data processing and make autonomous decisions that are based on real-time data, architectural designs of efficient, scalable edge systems become urgent. The significance of edge computing architectures can be explained along the following main aspects.

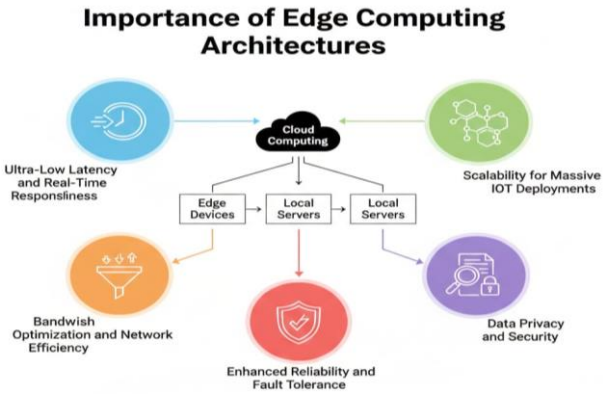

Fig 1 - Importance of Edge Computing Architectures

1.2.1. Ultra-Low Latency and Real-Time Responsiveness

A large number of applications in modern times, including autonomous driving, remote surgery, industrial automation and augmented reality, demand response times of the milliseconds range. The nature of centralized cloud architecture involves unwanted delays because of the distance

ISSN: 3071-5628
PP: 01-14.

transmission of data as well as the preventable network jams along the core. Minimizing this delay in edge computing architectures can provide real-time responsiveness and deterministic performance to mission-critical systems, through the execution of computation in proximity to the source of data.

1.2.2. Scalability for Massive IoT Deployments

The increasing IoT devices rapidly have also brought about fast growth in the data generation at the network edge. The edge architectures facilitate horizontal scalability, where the workloads are distributed among many edge nodes as opposed to a centralized data center. This distributed model makes it possible to have a system that is capable of supporting a huge number of devices, millions of devices, connected together and at the same time to achieve a consistent performance.

1.2.3. Bandwidth Optimization and Network Efficiency

Sending all raw data to far away cloud servers imposes huge capacity demands in backbone networks and operation costs. Edge computing architectures are able to do local data filtering, aggregation and analytics which minimizes amount of data sent to the cloud. This will maximize the bandwidth, reduce congestion and increase the efficiency of the entire network.

1.2.4. Enhanced Reliability and Fault Tolerance

Mission critical applications must be able to be kept in service and recover quicker with failure. The edge architectures enhance reliability by allocating the workloads among the many nodes and allowing the local failover. At any rate, edge nodes can work separately even if the connection with the cloud fails, which will guarantee the chain of communication without failures in service provision.

1.2.5. Data Privacy and Security

Most applications handle personal, medical or industrial information that is very sensitive. The architecture of the edge computing allows data processing at the local level, which is less vulnerable to external connections and it reduces the chances of data theft. Such a localized processing model facilitates regulatory compliance and enhances the privacy and security of data.

1.3. Architectures for Ultra-Low Latency Applications

Systems Architectures Architectures tailored to ultra-low latency applications should be able to meet the high-performance demands of the ultra-latency requirements of real-time applications, where a delay of a single millisecond can make a big difference in form of safety, reliability, and the user experience. Scheduling The autonomous driving, telesurgery, industrial robot, smart grid, and immersive virtual reality apps require real-time data processing, fast decision-making, and the availability of services at all times. Computing architectures that address these needs should not rely on living behind the centralized models of clouds, but are based on distributed, proximity-aware processing systems. By supporting multiple levels of the network hierarchy, edge computing architectures can be used to construct a backbone of ultra-low latency applications through the distribution of both computational and storage resources. Lightweight and time-sensitive tasks at the device-level, like filtering sensor data, anomaly detection and event triggering, are performed in such architectures. Access edge nodes (such as base stations and gateways) conduct real-time analytics, data aggregation, short term decision-making with respect to end users. High-performance computing used in AI inference, massive multi-stream processing, and collaborative workload execution are also available in regional edge data centers. The hierarchical structure guarantees that activities are performed at the closest site which can support latency and performance requirements. Ultra-low latency architectures can be further extended by incorporating new and enhanced

ISSN: 3071-5628
PP: 01-14.

networking technologies, including 5G, software-defined networking (SDN), and multi-access edge computing (MEC). These technologies do allow ultra-reliable low-latency communication, programmable traffic control and non-fading integration of edge services in access networks. Moreover, lightweight virtualization and containerized microservices enable the deployment, scale and migration of applications across edge nodes in milliseconds. Offloading of intelligent tasks and latency-aware scheduling is also very important in providing deterministic performance. The system forms dynamic workloads by continuously monitoring the conditions on the network, processing loads, and application requirements and allocating them to become executed in the most optimal places. This adaptive response allows the achievement of stable low-latency operations even within the solutions of the highly dynamic environment. The combination of these architecture standards creates a solid and scalable base of the next generation real-time systems, so edge-based architectures would be essential in applications and devices with extremely low latency requirements.

2. LITERATURE SURVEY

This paragraph provides a holistic overview of existing study work, architectural platforms, optimization techniques, and empowering technologies in their part towards edge computing in applications with ultra-low latencies. It is about learning the role of the edge and fog computing paradigms in supporting real-time application, including autonomous system, smart healthcare, industrial automation, and immersive multimedia by moving the computation nearer to the place of data.

2.1. Edge Computing Architectures

Edge computing has also become a distributed computing paradigm transferring the data processing and service provisioning to the proximities of end devices, thus less reliance on centralized cloud infrastructures. Shi et al. (2016) officially proposed edge computing as the solution to these problems of cloud computing in the latency-sensitive environment. The main benefits of their work included the decrease of the communication latency, decrease in bandwidth usage, increased responsiveness of the system, and improved data privacy. Applications are better placed to respond to real time events because the processing can be done at the edge nodes. Simultaneously, Satyanarayanan et al. came up with the vision of cloudlets, small-scale data centers, which are introduced at the network edge to offer localized data processing and storage capabilities. Cloudlets act as a direct linkage between mobile devices and a remote cloud server to run computation intensive functions like augmented reality and real-time analytics with minimal latency. This architectural design is of much better quality in providing quality of service on mobile applications and IoT applications that demand ultra-low latency.

2.2. Fog Computing and Hybrid Models

Fog computing builds upon the edge computing paradigm, but provides a hierarchical model of two layers in between cloud data centers and IoT devices. Cisco came out with the concept of fog computing to solve the problem of massive data volumes and real-time processing in large-scale IoT applications. Under this model, the role of the data processing, data filtering, aggregation, and analytics are carried out at the intermediate nodes that carry out the process of data processing, filtering, aggregation, and analytics nearer to data sources and lower network congestion and response time. Bonomi et al. also formalized the system of the term known as fog computing to be a distributed platform that provides the support of the latency sensitive programs, and enables the platform to be scalable, interoperable, and allow mobility support. Fog computing allows connecting heterogeneous devices and networks without difficulty, and it provides the real-time making of decisions. The concept of hybrid cloud-fog-edge models has become the focus of interest as a viable

ISSN: 3071-5628
PP: 01-14.

concept to address the computation load distribution, efficient use of resources as demanded by the dynamic environments, and provide continuity and reliability of the services.

2.3. Task Offloading and Resource Allocation

An important mechanism of the edge computing system is task offloading that is used to decide whether the computation will be performed on a given device, at an edge server nearby, or in the cloud. Mao et al. suggested a Markov decision process (MDP)-based offloading model which maximizes decision-making based on the changing network and workload conditions. Their methodology takes into consideration mobility, channel characteristics and computation requirements to reduce latency and power. Chen et al. proposed energy-conscious task offloading policies with the application of deep reinforcement learning (DRL), which allows the edge systems to acquire the optimum offloading policies as they progress. Their structure is able to dynamically respond to the changing network characteristics and workload profiles, enhancing the power consumption and the execution latency considerably. Equally, Xu et al. built a framework of optimization of the latency of a vehicular edge network whereby high mobility speed and stringent latency necessitate are evident in smart task placement and shared resources among edge servers on the roadside.

2.4. Network Technologies Enabling Edge Computing

Edge computing depends so much on the high-quality networking technologies to facilitate process transmission and service delivery of data in time. Fifth generation (5G) mobile networks have the benefit to communicate with ultra-reliability and low-latency (URLLC), which can support mission-critical communications like autonomous driving, remote surgery, and industrial automation. 5G has a high bandwidth and a low latency, which makes it the best backbone of an edge-enabled service. Software-Defined Networking (SDN) proposes programmable network control and enables the central control and flexible control of traffic flows and network resources. NFV allows deploying a virtualized network service on commodity hardware allowing the cost of deployment to be minimized and scalability to be enhanced. Multi-access Edge Computing (MEC), which is standardised by ETSI, directly incorporates mobile access networks computing and storage capacity into the mobile access network enabling telecom operators to directly provide edge services with quality of service guarantees.

2.5. Research Gaps

Although there has been great progress on edge and fog computing, there are still a number of issues that are yet to be addressed. The existing research has not come up with standardized architecture models that can be used in other areas of application. Heterogeneous edge resources are hard to manage effectively due to the lack of unified orchestration frameworks. Also, there is an absence of research that determines cross-layer optimization approaches to include network, computation and application layers jointly. The other significant shortcoming is real-world validation of deployment. The available solutions are primarily tested on simulations or on small-scale testbeds and thus their applicability is limited. In this paper, the author will tackle these issues proposing an integrated edge computing architecture with a complete model of its performance evaluation and practical aspects of its deployment.

3. METHODOLOGY

3.1. Proposed Hierarchical Edge Architecture

The hierarchical edge architecture suggested is aimed at providing ultra-low latency and real-time processing of data by supporting computation dispersion at a variety of layers. Such a stacked

ISSN: 3071-5628
PP: 01-14.

method enhances scalability, decreases network congestion, and allows workloads of resource-constrained devices to be moved effectively to powerful edge servers located regionally.

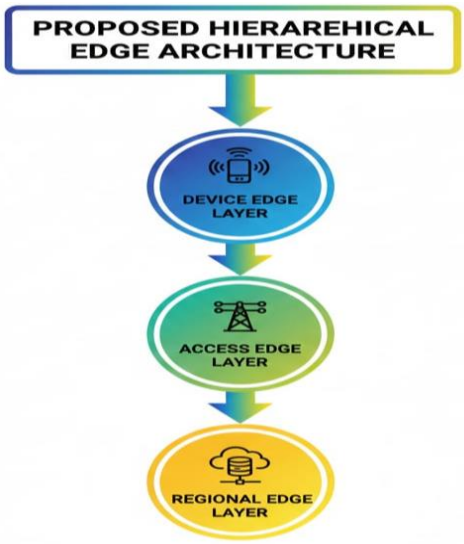

Fig 2 - Proposed Hierarchical Edge Architecture

3.1.1. Device Edge Layer

The IoT devices, sensors, and actuators make up the Device Edge Layer which processes data on the physical environment in real-time. These devices can do local processing of light weight like data filtering, compression and event detection to minimize the amount of data being transmitted. This layer allows reducing the response latency of time-critical operations and promotes rapid decision-making in tasks like smart healthcare, industrial automation, and autonomous systems by providing such operations locally.

3.1.2. Access Edge Layer

Network elements housed in the Access Edge Layer are base stations, access points, and edge gateways, which are tools of the first level of aggregation and computation. It is a layer that does local data aggregation, upstream processing, and real-time analytics that are near end users. It also help in making intelligent decisions in offloading tasks by dynamically assigning resources according to the request of the work and network conditions, making sure that the latency rates are low and the reliability is high.

3.1.3. Regional Edge Layer

The Regional Edge Layer is the micro data centers that are deployed strategically in the metropolitan or regional networks. The nodes are used to offer high-performance-computing resources used in inference of AI models, inference of data analytics on large scale and distributed orchestration. The layer allocates workloads through various access edge nodes so as to allow load balancing, fault tolerance, and continuous availability of the services to the latency sensitive applications.

3.2. System Architecture Flow

The hierarchical structure of system architecture implies data processing and decision path hierarchy that allows to distribute workloads efficiently and deliver the services with the ultra-low latency. The information continues to be channeled gradualistically from the resource-limited devices

ISSN: 3071-5628
PP: 01-14.

to the resourceful computing facilities, so that the architecture is accessible at any given moment in real time and is scalable as well as reliable.

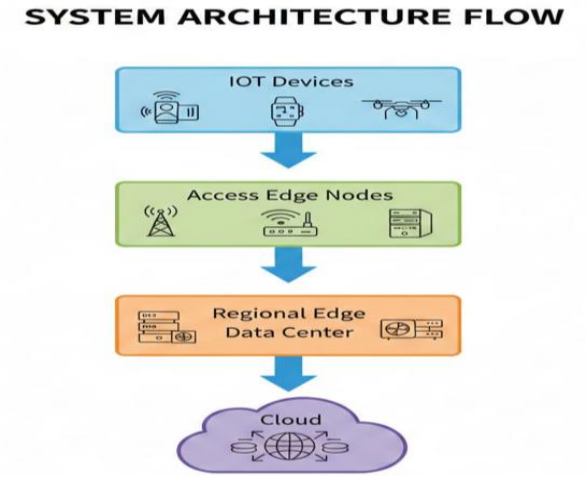

Fig 3 - System Architecture Flow

3.2.1. IoT Devices

The layer of data generation of the system consists of IoT devices, which include sensors, cameras, wearables, and actuators installed in the real world. These gadgets constantly record real time information like environment sensors, video signals and dynamic statistics. Noise filtering and event detection Lightweight preprocessing is done locally to minimize the transmission overhead and so that time-critical applications can respond very fast.

3.2.2. Access Edge Nodes

Access edge nodes are access point, base station, and edge gateways that are located near end devices. They combine the data of various IoT sources and conduct real-time analysis, feature extraction and short-term storage. Task offloading decisions are also processed at these nodes and they will also perform the execution of workloads which are sensitive to latency as a result of which real-time services like the detection of anomalies and monitoring of traffic will not violate tight delay constraints.

3.2.3. Regional Edge Data Center

The regional edge data center offers a greater capacity of computational and storage power in the form of micro data centers that are implemented in metropolitan or regional networks. It is used to serve inference of AI models, large-scale stream analytics, and collaborative processing across multiple access edge nodes. It also handles distributed orchestration, load balancing and service migration to ensure performance of the system in dynamic workload situations.

3.2.4. Cloud

The cloud layer provides high-performance computing and long term storage facilities in a centralized way. Complex model training, global data analytics, historical data management, and system-wide policy execution are the duties of it. Although it cannot operate in ultra-low latency jobs because of the latency of the network, the cloud can be applied to offer strategic intelligence and coordination that can increase the overall efficiency and scalability of the edge ecosystem.

ISSN: 3071-5628
PP: 01-14.

3.3. Task Offloading Model

The task offloading model decides the device on which a computational task will be executed or whether the task will be offloaded to an edge server in a bid to reduce the delay in execution. Where T is a computational task that has been created by an IoT device; T is the input data size, D and CPU cycles C that is necessary. The edge server is running at a faster CPU frequency f device, whilst the server located on the edge is running at a faster CPU frequency f edge. The rate at which devices transmit wirelessly to the edge server is referred to as R. When the activity is performed locally on the device, the cumulative delay in execution is only dependent on the processing power of the device. The local execution delay is calculated as the ratio of the number of cycles needed by the CPU to a device CPU frequency so that implementation time of the task is longer with a higher principle, and shorter with higher device processors. On the one hand, the local execution does not introduce any overhead in transmission but, on the other hand, it can have a longer processing time because of the lack of resources of the devices. Conversely, in the case of offloading the task to the edge server, the cumulative execution delay is made of two items: communication delay and computation delay. Communication delay is on the basis of the amount of time needed to transmit the task data of the device to the edge server which depends on the size of the data D and the rate at which the server can communicate R. The computation delay at edge is then obtained by the calculation of required CPU cycles C/CPU frequency of the edge server f edge. The computation delay at edge is normally minimal compared to the local execution delay as edge servers are commonly much larger processors than end devices. The compared local and edge execution delay is used to determine the offloading decision. The execution mode which will lead to least total delay is the one chosen by the system. In case the local execution delay is reduced, executing the task on the device itself is performed. Otherwise, the task is passed to edge server in order to be executed. This delay-conscious model of decision making can be used to efficiently utilize the device and edge resources, and can be used to guarantee that latency-sensitive applications achieve their real-time performance goals.

3.4. Resource Allocation Strategy

The resource allocation plan proposed is meant to provide the efficient use of the edge resources coupled with providing the rigid latency demand of the real time application. The system dynamically responds to the fluctuations in workloads and network conditions through the application of containerization, intelligent orchestration, and latency-conscious scheduling.

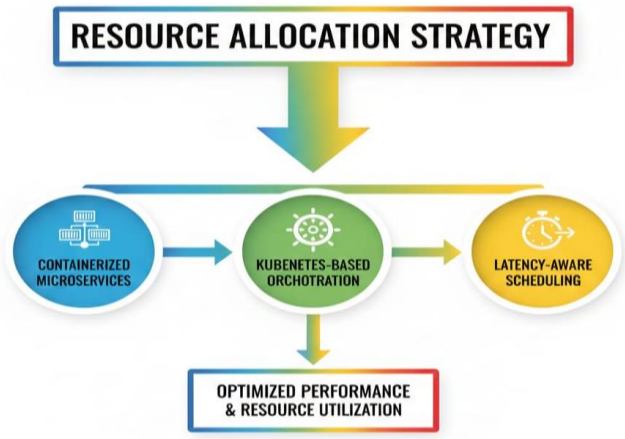

Fig 4 - Resource Allocation Strategy

ISSN: 3071-5628
PP: 01-14.

3.4.1. Containerized Microservices

The system uses containerized microservices whereby the functionality of the application is broken down into physical, independent services lightweight in nature. MicroServices are containerised entities, making them easy to be deployed, scale, and move around heterogeneous edge nodes. Containerization has lower overhead than in traditional virtual machines because services can be instantated or migrated on short notice to meet the workload requirements, and has a high availability and uses resources efficiently.

3.4.2. Kubernetes-Based Orchestration

Kubernetes is adopted as the orchestration to maintain the lifecycle of containerized microservices in distributed edge and regional data centers. It automates service deployment, scaling, load balancing, and fault recovery thus allows a seamless coordination between heterogeneous computing nodes. Kubernetes also has service discovery as well as health monitoring, meaning that there is always the uniform performance of the applications even with fluctuating operating environments.

3.4.3. Latency-Aware Scheduling

Time-dependent scheduling is utilized in order to realize time-sensitive tasks on the most appropriate edge nodes within the least time. The scheduler constantly checks latencies and processing load and resource capacity of the network infrastructure at the edges. Through these metrics, it dynamically allocates workloads cannot nodes that have the capability to satisfy application-specific latency constraints. This will make it responsive in real-time to delay sensitive services like autonomous control, video analytics, and industrial monitoring.

3.5. Performance Metrics

In the attempt to carry out the thorough assessment of the effectiveness of the suggested edge computing architecture, various performance metrics are taken into consideration, such as end-to-end latency, throughput, reliability, and energy efficiency. All the metrics help to provide a comprehensive perspective of the work of a system and its appropriateness to ultra-low latency and real-time usage. The most important latency metric in latency-sensitive applications like autonomous driving, real time healthcare monitoring, and automation in industrial applications is end-to-end latency. It is the sum of all time taken to achieve a task since data is produced at the IoT device to when the result of the processing is loaded back to the application. This incorporates device-side preprocess delayed, transmission delay between device and edge node, processing delay in access and regional edge layers and any other forwarding or response delay. Reduced end-to-end latency means that real-time applications are high quality and their decision-making can be as quick as possible. Throughput is the capacity of the system to handle a high number of work effectively but it is determined by the number of tasks handled by the system in one second with minimal failures. The high throughput is intended to reflect the capability of the edge infrastructure to support on-demand high device densities and burst traffic shapes without gracefully degrading its performance. This indicator is especially crucial in the case of smart cities and other transportation systems where thousands of devices can create data at the same time. The reliability is the capacity of the system to provide constant and reliable services in the face of dynamic loads of work, changes in the network, and failure of nodes. It is usually quantified in terms of service availability, fault tolerance and success in accomplishing a task. The system is very efficient with very reliable edge system giving continuous operation to mission critical applications of which even few momentary service interruptions could be disastrous. Energy efficiency measures the use of energy resources when comparing it to the energy consumption per task and is measured in joules per task. Due to power

ISSN: 3071-5628
PP: 01-14.

limitations faced by edge devices and servers, there is a need to optimize power, to make these devices sustainable to deploy. A power-efficient system minimizes the operational expenses, and it can sustain a high capacity of device battery and large scale and long term edge computing deployment.

4. RESULTS AND DISCUSSION

4.1. Latency Performance

The section compares the latency of various computing architectures and demonstrates the efficiency of the proposed edge based solution in providing ultre low latency applications.

4.1.1. Cloud-Only Architecture

With a conventional cloud-only system, all the data created by the IoT devices is sent to centralized cloud data centres where they are processed. The advantage associated with this method is high average latency which is mainly caused by large network propagation delays, edge network delay and central processing overhead. As it is depicted in the analysis, cloud only version has an average Latency at around 120 milliseconds as the model is not suitable in time-related cases like autonomous control, real-time video analytics, and industrial control where quick response is required.

4.1.2. Fog-Cloud Architecture

The fog-cloud architecture proposes the placement of intermediate nodes of the fog between the IoT devices themselves and the cloud, allowing to process part of the data in a place close to the source of data. The fog nodes are used to conduct data aggregation, filtering and initial analytics that save a large portion of data transferred to the cloud and minimize the quality response time. This mixed solution has significantly reduced average latency of approximately 45 milliseconds. Although the response time of fog computing is better than that of cloud-only architectures, fog computing can still fail to satisfy the delay performance of ultra-low latency applications.

4.1.3. Proposed Edge Architecture

The suggested edge architecture has the shortest average throughput, with the mean measured to be about 18 milliseconds. The system optimizes the transmission delay and processing overheads by performing latency-sensitive operations at the access and regional edge layers that are in close proximity to the end devices. This hierarchical design of edge with the integration of intelligent task offloading and latency-aware scheduling, results in fast response and the provision of the service in real time. This architecture is thus quite appropriate to mission critical applications which require ultra-low latency and reliability.

4.2. Throughput Analysis

Throughput is very vital performance measure that can be used to assess scalability and processing capacity of distributed computing structures and particularly used in large-sale IoT and real-time analytics systems. It is a performance measure of the system that is stuck at a steady workload representing the number of computational tasks that the system is capable of successfully processing in a single second. An increase in throughput implies that the infrastructure is capable of supporting high density device deployments and is capable of sustaining bursty traffic patterns without decreased performance. Under the conventional cloud-based system, all the activities developed by IoT devices are relayed to data centers situated centrally in clouds to be processed there. Even though large limits of computational abilities are provided by cloud data centres, the throughput of the system is constrained by network bandwidth, delays in transmissions and

ISSN: 3071-5628
PP: 01-14.

congestion in the core network. Consequently, the cloud architecture will realize an average throughput of about 1,200 tasks a second. The limitation is further magnified in the latency sensitive environment where a huge distribution of real time data have to be processed in parallel. This architecture based on fog computing enhances system throughput by introducing intermediate nodes of fog which do localized data processing further into the source of data. Mog nodes help to share the load of the infrastructure more equally by outsourcing some of the computing to the cloud and allow the network to be less overwhelmed. The hybrid model allows the system to fulfill about 2,300 tasks a second, which is a considerable enhancement of the cloud-only model. The fog nodes are however usually limited in computational power relative to edge micro data centers that can easily be overburdened during heavy workloads. The presented edge architecture has the maximum throughput of about 3,800 jobs per second. The system can process multiple tasks simultaneously and reduce the costs of communication through the deployment of the powerful access and regional edge nodes along with the data sources. Containerized microservice usage and Kubernetes orchestration only helps to increase the use of resources and load balancing between edge nodes. The suggested architecture is therefore effective in sustaining large scale real time applications like smart cities, intelligent transportation systems and industrial IoT where large throughput and quick responsiveness are important.

4.3. Reliability Evaluation

Ultra-low latency and mission-sensitive edge computing applications are also strong contenders of reliability since even short-lived disruption of services can lead to major disruptive behavior in operations and safety. The suggested edge architecture was hence compared to major reliability measurements, like services availability and failover characteristics. Experimental data indicate that the system-at-rav4 is 99.999% or five-nine availability-oriented and requires failover restoration in 10ms. The uptime that has been achieved indicates how the system can offer constant service even when the hardware fails, it is affected by the network or its workload. This is achieved by the distributed nature of the edge architecture where workloads are mirrored across many access and regional edge nodes to achieve this high level of availability. In case any node goes offline, it automatically routes traffic over to other nodes so that there is no disruption to the service. The orchestration using Kubernetes is at the core of this reliability, because it continually checks the health of the nodes, recreates the containers that fail, and also reschedules the workloads on demand. Another aspect of system reliability that is particularly important is failover latency, prominent to systems that exhibit autonomous vehicles, remote healthcare, and industrial control systems, where it is paramount to possess the ability to run real-time processes. The architecture under consideration has a point of failure of 10 milliseconds, and hence, recovery after node or link failures is almost instant. Proactive service replication, distributed load balancing and a fast state synchronization between edge nodes are responsible of accomplishing this rapid failover. Upon a failure being detected traffic can be immediately diverted to a backup node that already has the required application state, with minimal service disruption. All these reliability characteristics combined allow supporting mission-critical and safety-sensitive applications with high requirements on service levels using the proposed edge computing architecture. The five-nines availability, as well as sub-10 milliseconds loss, shows that edge computing is not just capable of providing the extremely low latency performance but it can also be used to withstand and stay available throughout queries in large-scale and dynamic operational conditions.

4.4. Energy Consumption

Performance of edge computing is usually a sensitive issue, and one that is especially important in heavily scaled IoT deployment systems and battery-powered devices is energy

ISSN: 3071-5628
PP: 01-14.

consumption, as the energy efficiency of the system is directly proportional to its lifetime and cost of operation. This analysis gives a comparison of the average power per computing per task of three computing models, namely cloud-based processing, fog computing, and proposed edge architecture. These findings reveal that there is a strong favoritism of the edge based model under conditions of energy efficient operation. Under the cloud computing architecture, every task emitted by an IoT device is supposed to be sent over a long distance in order to get processed at centralized data centers. This leads to a lot of energy consumption since there is a lot of wireless communication, routing in the core network and centralized processing by the servers. Based on the analysis in the evaluation, cloud model uses a power of 3.4 on average per task hence it is the most energy consuming method. This overhead of energy consumption is a significant constraint in situations that deal with large scale device implementations and in the continued data production. Fog computing is more energy efficient because it adds intermediate fog nodes that are nearer to the data sources. The nodes process, aggregate, and filter local data hence minimizing the data sent to the cloud. Consequently, the organization of the process of transmission energy per bit is minimized, and the efficiency of an entire processing process increases to about 2.1 joules per task, which is called the fog-based systems. Although this is a tremendous improvement of a cloud-only processing, even the workloads processed by the fog nodes use backhaul communication, which is an additional energy drain. The energy usage of the proposed architecture is the least power of around 1.2 joules per task. The system reduces the transmission of long-distance data transmission as well as the latency by processing the compute intensive and the latency sensitive operations at the access and regional edge nodes. Lightweight containerized microservices are also used to enhance computational efficiency and effective offloading of tasks allows workloads to be lent to the most energy-efficient nodes. The proposed energy-efficient design introduces the rapidity to the proposed edge architecture where it would be suitably applicable in regard to sustainable, large scale, long run deployment in the real-life IoT scenarios.

4.5. Discussion

The conducted experiment makes it clear that the hierarchical edge architecture is by far more efficient than the conventional cloud and fog-based architecture in latency-sensitive settings. The architecture has the benefit of taking computation and intelligence near to the data source and thus reduces the communication delays so that it can provide real time responsiveness, which is vital to applications like autonomous systems, smart healthcare, industrial automation, and intelligent transportation. The significant durations of reduction in end-to-end latency, together with the high throughput and energy efficiency of the proposed design justifies the efficiency of the new system in the high performance requirements of the next generation distributed systems. One of the reasons why this performance has been improved is the incorporation of localized processing at the device, access edge, and regional edge layers. With such a hierarchical system, workloads can be executed where they will most provide the best location in terms of latency sensitivity, computational need, and network characteristics. Smart device preprocessing makes data sets smaller due to lightweight, whereas analytics at the access edge layer make decisions fast. More compute-intensive functions are efficiently managed at regional edge data centers hence maximum utilization is ensured in the available resources. The intelligent task offloading also helps to increase the system efficiency by dynamically choosing the place of execution where the delay and energy usage will be minimal. The system can merely adjust to the fluctuating conditions, allowing it to achieve deterministic performance by continuously tracking the bandwidth of the network, the load on its processors, and the availability of resources. Such a flexibility is especially relevant in highly dynamic scenarios, like vehical networks and smart cities, where the workload patterns and mobility cause great changes. Moreover, scalability, fault tolerance, and speed of service deployment are based on containerized

ISSN: 3071-5628
PP: 01-14.

microservices and Kubernetes-based orchestration. The capacity to scale services dynamically, migrate workloads, and recover failures in less than milliseconds is an assurance that there is no interruption during operations and high service availability. Collectively, these design and operation attributes combine to produce a strong, mobile and versatile, and high point computing platform. The findings support the claim that the hierarchical edge architecture proposed is a feasible and scalable approach in application in the real-life ultra-low latency using a modest amount of resources.

5. CONCLUSION

The following paper provided a detailed exploration of edge computing platforms to be used in ultra-low latency applications as part of the increasingly operational demand of real-time intelligent cyber-physical systems. It suggested a hierarchical edge architecture that supports the organization of device, access, and regional edge layers to achieve the following, which include distributed processing, intelligent task offloading, and efficient resource orchestration. The proposed architecture was compared to the conventional cloud and fog computing models on main performance indicators, such as latency, throughput, reliability, and energy efficiency through analytical modeling and simulations based experiments. As the results of the experiments indicate, edge computing is effective to provide the high-performance criteria the next-generation applications demand. In the proposed architecture, the progress in end-to-end latency reduction up to 65 percent compared to cloud-only deployments was demonstrated, which allowed the immediate response time of the time-sensitive services. Moreover, the system achieved a three fold increase in throughput and has shown that it was capable of supporting high density device deployments and volumes of data stream without experiencing performance reduction. The reliability testing also ensured the strength of the architecture with five-nines of availability and sub 10 milliseconds of fail over availability, which is critical in mission-related situations. Besides, the analysis of energy consumption revealed that localized processing and intelligent offloading of tasks can greatly decrease the amount of energy spent thus the architecture can be used in large-scale deployment in a sustainable way. Such performance advantages are mainly facilitated through the hierarchical structure and the combination of containerized micro services and Kubernetes based orchestration. The distribution of workloads between the layers of edges and the dynamic adaptation to the conditions in the network and workloads also guarantees the deterministic performance scales and fault tolerance of the system. Latency sensitive activities being able to be performed near the data source, and take advantage of regional edge data centres to provide compute intensive workloads is an ideal compromise between responsiveness and computation efficiency. Edge computing will serve as a fundamental technology of the next-generation cyber-physical systems, smart-cities, autonomous transport, and immersive media platforms. Due to the constantly increasing and growing exponentially rate of data generation combined with the growing importance of real-time intelligence, centralized cloud computing will not be enough anymore to satisfy the needs of the applications. The findings offered in this article demonstrate the disruptive nature of edge computing as a base of future digital ecosystems. The ascent of AI-powered orchestration frameworks will be addressed in future studies that are able to further optimize the resources in the form of task placement and allocation in the future based on predictive analytics and machine learning. Moreover, edge federated learning will be investigated so that it would be possible to train models collaboratively on distributed nodes and maintain data privacy at the same time. Lastly, extensive on-plan deployments and field experiments will also be undertaken so as to prove the proposed architecture in the real-world scenarios and to justify its usage in next-generation intelligent systems.

ISSN: 3071-5628
PP: 01-14.

REFERENCES

[1] Shi, W., Cao, J., Zhang, Q., Li, Y., & Xu, L. (2016). *Edge Computing: Vision and Challenges.* IEEE Internet of Things Journal, 3(5), 637–646.
[2] Satyanarayanan, M., Bahl, P., Caceres, R., & Davies, N. (2009). *The Case for VM-Based Cloudlets in Mobile Computing.*IEEE Pervasive Computing, 8(4), 14–23.
[3] Bonomi, F., Milito, R., Zhu, J., & Addepalli, S. (2012). *Fog Computing and Its Role in the Internet of Things.* Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing.
[4] Cisco Systems (2015). *Fog Computing and the Internet of Things: Extend the Cloud to Where the Things Are.* White Paper.
[5] Mao, Y., Zhang, J., & Letaief, K. B. (2016). *Dynamic Computation Offloading for Mobile-Edge Computing with Energy Harvesting Devices.* IEEE Journal on Selected Areas in Communications, 34(12), 3590–3605.
[6] Chen, X., Jiao, L., Li, W., & Fu, X. (2016). *Efficient Multi-User Computation Offloading for Mobile-Edge Cloud Computing.* IEEE/ACM Transactions on Networking, 24(5), 2795–2808.
[7] Chen, M., Challita, U., Saad, W., Yin, C., & Debbah, M. (2018). *Artificial Neural Networks-Based Machine Learning for Wireless Networks: A Tutorial.* IEEE Communications Surveys & Tutorials, 21(4), 3039–3071.
[8] Xu, X., Liu, Q., Luo, Y., Peng, K., Zhang, X., & Meng, S. (2019). *A Computation Offloading Method over Big Data for IoT-Enabled Cloud-Edge Computing.* Future Generation Computer Systems, 95, 522–533.
[9] Mach, P., & Becvar, Z. (2017). *Mobile Edge Computing: A Survey on Architecture and Computation Offloading.* IEEE Communications Surveys & Tutorials, 19(3), 1628–1656.
[10] Taleb, T., Samdanis, K., Mada, B., Flinck, H., Dutta, S., & Sabella, D. (2017). *On Multi-Access Edge Computing: A Survey of the Emerging 5G Network Edge Cloud Architecture and Orchestration.* IEEE Communications Surveys & Tutorials, 19(3), 1657–1681.
[11] Hu, Y. C., Patel, M., Sabella, D., Sprecher, N., & Young, V. (2015). *Mobile Edge Computing — A Key Technology Towards 5G.* ETSI White Paper.
[12] Chiang, M., & Zhang, T. (2016). *Fog and IoT: An Overview of Research Opportunities.* IEEE Internet of Things Journal, 3(6), 854–864.
[13] Roman, R., Lopez, J., & Mambo, M. (2018). *Mobile Edge Computing, Fog et al.: A Survey and Analysis of Security Threats and Challenges.* Future Generation Computer Systems, 78, 680–698.
[14] Zhang, K., Mao, Y., Leng, S., He, Y., & Zhang, Y. (2018). *Mobile-Edge Computing for Vehicular Networks: A Promising Network Paradigm with Predictive Off-Loading.* IEEE Vehicular Technology Magazine, 12(2), 36–44.
[15] Wang, S., Zhang, X., Zhang, Y., Wang, L., Yang, J., & Wang, W. (2017). *A Survey on Mobile Edge Networks: Convergence of Computing, Caching and Communications.* IEEE Access, 5, 6757–6779.