Optimizing Real-Time Data Processing: Edge and Cloud Computing Integration for Low-Latency Applications in Smart Cities

[1]Rajesh Kumar Malviya, [2]Ravi Kumar Vankayalapati, [3]Lakshminarayana Reddy Kothapalli Sondinti, [4]Sathiri Machi

[1]Enterprise Architect

[2]Infrastructure Lead,ORCID : 0009-0002-7090-9028

[3]senior software engineer,ORCID: 0009-0005-2775-1730

[4]Quality systems Engineer

Abstract

The rapid expansion of smart city infrastructure necessitates efficient real-time data processing to enhance urban services and improve citizen experiences. This paper explores the integration of edge and cloud computing as a strategic framework for optimizing low-latency applications in smart cities. By distributing data processing tasks between edge devices and cloud servers, we minimize latency and bandwidth usage while maximizing computational efficiency. We analyze various use cases, including traffic management, environmental monitoring, and public safety systems, demonstrating how this hybrid approach addresses the unique challenges of real-time data handling in urban environments. Through experimental evaluations and simulations, we quantify performance improvements and propose best practices for implementing this integrated architecture. Our findings suggest that leveraging edge-cloud synergy can significantly enhance responsiveness and scalability in smart city applications, paving the way for more adaptive and intelligent urban ecosystems.

Keywords: Real-Time Data Processing, Edge Computing, Cloud Computing, Low-Latency Applications, Smart Cities, Data Integration, IoT (Internet of Things), Distributed Computing, Latency Optimization, Data Analytics, Scalability, Network Architecture, Fog Computing, Stream Processing, Data Sovereignty, Machine Learning at the Edge, Real-Time Analytics, Smart Infrastructure, Urban Computing, Sensor Networks, Resource Management, Traffic Management Systems, Predictive Maintenance, Interoperability, Data Security.

1. Introduction

Over the recent decade, the emergence of the Internet of Things (IoT) along with smart cities has attracted a considerable amount of interest as a means for the efficient design and management of urban environments. In the coming decade, the number of IoT devices to be installed is expected to be over 75 billion, which will generate large volumes of data that need to be processed in real-time. Thus, there is a growing need for low-latency applications that can handle vast data flows. In smart cities, an application capable of analyzing real-time data will enable efficient management of smart traffic light systems, smart waste management, and many other smart features. Consequently, in addition to handling low-latency applications, data generated within urban environments have unique characteristics that can be exploited for improved data analysis.

In this paper, we demonstrate the potential benefits of integrating edge and cloud computing. Intuitively, local data processing reduces the amount of data that needs to be processed and transferred to the cloud. This results in a decrease in the cloud's processing load and a reduction in the volume of data needing to cross networks. This is particularly beneficial in large-scale contexts such as smart cities, capable of generating and transferring large volumes of data in near real-time. The ever-increasing availability of electronic devices, associated with the rapid growth of both urbanization and technologies, requires increasingly efficient methods of managing the vast volumes of data generated by such technologies. Generally, data processing architectures include edge computing, cloud computing, or edge and cloud computing. A comprehensive understanding of how edge and cloud computing integrate is essential to handling those massive amounts of data. Thus, we first investigate edge and cloud computing to form the basis for the proposed data processing architecture.

We then propose use case demonstrations of the proposed architecture to represent real-world applications and evaluate the applicability of integrating edge and cloud computing to address latency in data processing. Overall, the proposed studies in the rest of the paper focus on the following two interrelated objectives: (1) Unraveling the interplays of edge and cloud computing; and (2) Real-world application of the potential to add value as a result of integrating edge and cloud computing into a data processing architecture.In this paper, we explore the transformative potential of integrating edge and cloud computing within the context of smart cities, where the proliferation of IoT devices is set to exceed 75 billion in the coming decade. This rapid expansion generates vast volumes of data that necessitate real-time processing for effective urban management, including applications such as smart traffic systems and waste management. By leveraging local data processing at the edge, we can significantly reduce the burden on cloud infrastructures, leading to enhanced efficiency in data transmission and processing. Our investigation delves into the interplay between edge and cloud computing, laying a foundation for a robust data processing architecture capable of handling the unique characteristics of urban-generated data. Through use case demonstrations, we illustrate the practical implications of this integration, highlighting its ability to minimize latency and improve responsiveness in real-world applications. Ultimately, this study underscores the importance of a cohesive data management strategy that harnesses the strengths of both edge and cloud computing to effectively address the challenges posed by rapid urbanization and technological advancement.

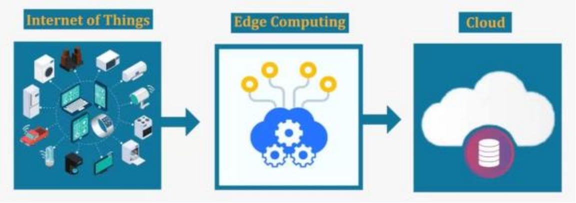

Fig 1 : Real-Time Data Processing in Edge and Cloud Computing

1.1. Background and Motivation

The evolution of information and communication technologies has significantly altered data processing techniques. The increased deployment of devices has led to a massive surge in data collection. In urban contexts, these data streams may correspond to vital applications such as health monitoring, traffic analysis, and environmental sensing. In such applications, data latency and dependability are of significant concern. Smart cities already consist of different resources and computation infrastructures that may be used to process data. Edge computing and cloud computing work together to allow for real-time data processing. With real-time data processing, immediate responses and informed decisions can be made. However, data may be generated at a very high rate, and managing and processing large quantities of data on these two computational platforms can be a tricky task.

Currently, there has been no research on efficiently distributing computation over edge and cloud resources for smart cities. Our objective is to provide a more real-time system model for data management in smart city services that are used for future experimentation. Real-time data management and analysis are utilized in several smart city applications. It aids in the identification of potential abnormalities, mitigation approaches, and visualization of the environment. Generally, data is produced from a variety of sources and in large quantities. The first requirement is to ensure that any irregularities in the environment are detected to advise decision-makers to take proper steps. Real-time service is the solution by which it is feasible. That is, there is a minimal amount of lag between the time the event occurs and the time the impact is displayed on the eco-dashboard. In the road network, a road accident occurs every X minutes. The event is perceived, and a suitable graph of the current disaster is shown on the dashboard. These are gadgets or devices that have been integrated, networked, and have the ability to communicate between themselves and with other gadgets and networks.

1.2. Research Objectives and Scope

In response to the recent advances in technology and information, as well as to the increasingly dynamic and competitive markets in the Internet of Things (IoT) and smart cities contexts, the research explored in this paper is carried out with the following objectives: to identify the main challenges in data processing frameworks currently used; verify how the combination of edge and cloud computing environments can address these challenges; define scenarios and use cases in which data processing combines both environments at the best edge and cloud ratio as each case demands; focus on low-latency applications since it is considered a top priority in urban management and in the delivery of urban services; demonstrate the appeal of the low-latency data processing middleware in action; propose how to combine the benefits of sampling and aggregation, which is part of our mechanism, with the data currently and accurately measured, in this case the conducted analysis of the context-dependent validity of the calculated pollutant concentration.

The new hypes related to the development of edge, fog, or cloudlet data processing are turning into reality. New technologies need to be allied with what is real in these areas. As neither edge nor cloud, and neither edge-cloud nor cloud-edge architecture is determined to be better once and for all, this further leads to a complexity not known in any other data processing area. We investigate further what happens when considering both worlds: edge and cloud. For instance, we simulate a smart utility in a smart city setting the edge-cloud ratio to 20/80, where the data processing power is provided by the edge: 20%, and by the cloud: 80%, respectively. Then, we consider the same smart city use case, offering 95% of the data processing from the edge and only the remaining 5% from the cloud; the edge-cloud ratio changes as per use case demands. As future research, we plan to explore how edge-cloud or cloud-edge can form a hybrid resource pool, persistent graph, and what a reasonable time to live (TTL) is for persisting an object as it gets less and less actively visited.This research paper addresses the evolving challenges in data processing frameworks within the dynamic landscapes of IoT and smart cities. Our objectives include identifying key challenges in existing frameworks, examining how the integration of edge and cloud computing can mitigate these issues, and defining use cases that optimize the edge-cloud ratio based on specific demands. Emphasizing low-latency applications as a priority for effective urban management, we demonstrate the practical implementation of low-latency data processing middleware. Through simulations of a smart utility, we explore varying edge-cloud ratios—such as 20/80 and 95/5—to illustrate how data processing power can be flexibly allocated between edge and cloud environments. As a future direction, we aim to investigate the potential for hybrid resource pools and the optimal time-to-live (TTL) for data persistence in this complex integration of edge and cloud technologies.

Equ 1: Average Network Delay and Queuing Theory basics

$$q = 1P(S_1) + 2P(S_2) + 3P(S_3) + \cdots + infinity\, P(S_{infinity}) => \sum_{n=0}^{n=infinity} nP(S_n)$$

Applying equation 4 and then taking $\rho\, P(S_0)$ out

$$q = \sum_{n=0}^{n=infinity} nP(S_n) => \sum_{n=0}^{n=infinity} n\rho^n P(S_0) => \rho\, P(S_0) \sum_{n=0}^{n=infinity} n\rho^{n-1}$$

we know that from calculus that $\frac{d}{d\rho}\rho^n = n\rho^{n-1}$ and applying sum rule

$$q = \rho\, P(S_0) \sum_{n=0}^{n=infinity} \frac{d}{d\rho}\rho^n => \rho\, P(S_0)\frac{d}{d\rho} \sum_{n=0}^{n=infinity} \rho^n$$

.

$$q = \rho\, P(S_0)\frac{d}{d\rho}[1 + \rho + \rho^2 + \rho^3 + \rho^4 + \rho^5 + \cdots]$$

we know that $P(S_0) = 1 - \rho$ from equation 6, Substituting that for $P(S_0)$ and equation 5 for geometric series

$$q = \rho\, P(S_0)\frac{d}{d\rho}\frac{1}{1-\rho} => \rho\, P(S_0)\frac{1}{(1-\rho)^2} => \rho(1-\rho)\frac{1}{(1-\rho)^2}$$

2. Foundations of Edge and Cloud Computing

Edge computing has recently emerged as a cutting-edge concept and is used to describe the distribution of processing, storage, and control functions to the physical and logical communication endpoints of a network. Conversely, cloud computing is the delivery of computing services over a network and encompasses mechanisms for convenient and on-demand network access to a shared pool of configurable resources. Cloud also delivers services such as servers, storage, and applications. Techniques deployed in cloud networks include virtualization, distributed computing, and service-oriented architectures. Digital information services are executed on applications running within data centers for both edge and cloud frameworks. However, both mechanisms are assessed according to whether they are decentralized, allowing logical services to be moved physically closer to end users. Edge and cloud computing architectures have become increasingly important for the development of real-time applications. The process of reading, updating, and deleting is a common procedure used in all real-time scenarios. An edge consists of three types of environments: device edge, network edge, and data center edge. Devices in device edges enclose the physical world, including sensors, actuators, and cameras, and are connected with Wi-Fi and 4G. The Wi-Fi local area network is used for limited device transmission technologies. The network edge enables bidirectional communication with the device edge. It then forwards the data to the data center for the final computation. Finally, the data center edge can be configured as an intra-data center cloud resource or used as a GPU system for further data processing. To perform real-time data processing, data processing platforms can effectively distribute corrected and matched data across the edge and the cloud. To exchange messages from one medium to another, messages travel across processing channels. With increased data volume, topologies can be utilized to implement stream processing. Resilient edge data stores can be used to store edge data sets and perform real-time querying. In the cloud, the data can be further processed using batch data processing engines.

2.1. Concepts and Definitions

In this adherence, many principles, pioneers, and tech-limiters that are habitually consumed within this protocol are approached. In these tenets, a compendium of each term employed in pushing the functions is granted, guiding the consummation of common ground within the processes. Latency is a period established between the onset of an event and the inception of that event resulting in equal response. Scalability pertains to the proficiency of purposes to handle challenges with idiosyncrasies in a realization locality. Data sovereignty is recognized as the conventional law-based possession of an entity's proof based on the precedence of their clear opinion. Furthermore, the edge computing rationale is to lessen the pause in facilitated responses by progressing the data to the place it is needed; the surface of edge computing remarkably aids the deployment of usages close to the consumer. In association with edge computing, cloud computing extends extensive storage and computational capabilities delivered on a large scale. Rest signs within associations to cloud processes are essentially used globally to amplify the end-to-end manner.

Edge computing and cloud computing share correlated advantages; however, their overlaps in a facility are distinctly distinguished based on computational databases. These strategic demonstrations exhibit the superiority of the integrated system, based on the mixture of edge and cloud computing in comparison to the instance in which they are singularly applied in a real-time database. Moreover, the outcomes predominantly deduce the constructive results in applications connecting real-time data, thus proposing the value of the associated kinetic processing methods. Hence, recovery accompanied by real-time minor infatuations of accessible worth in the bulk of the utilized technologies at the moment is used by numerous business types, vanilla events, unitary technologies, as well as computational formulations.

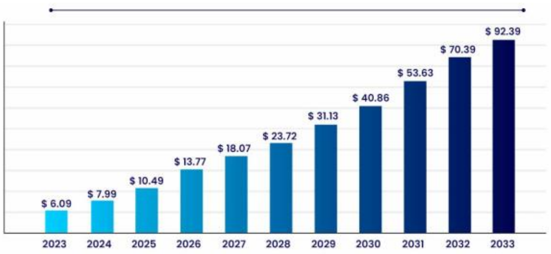

Fig : Edge Computing Market

2.2. Architectural Overview

The architectural overview provides a detailed visual and conceptual representation of edge and cloud computing integration. It illustrates the architectural layer that composes each one—edge computing and cloud computing—and the components that form it, while also showing how these components in each architecture interact to manage data processing in the urban space. The interaction is expressed through functions, which correspond to the components. Each function comprises a physical or virtual entity in the edge or cloud computing layer. Based on these layers and functions, architectures focus on edge IoT devices, the cloud, a combination of cloud infrastructures, and edge research environments.

In the first paradigm, a unique architecture focuses on edge IoT devices, as the main data processing populates the resource-constrained devices. Paradigms considering the exclusive cloud as the infrastructure have multiple architectures. The first purely cloud architecture appears where all three functions are deployed in a unique cloud. Proposed architectures also present cloud paradigms with a hierarchy, but all contain virtual components, originating different styles of architecture. The last cloud paradigm in this architectural overview also contains virtual entities. The complex architectures are hybrid, forming a combination of cloud architectures. They show the interoperability of either the most recent or emerging architectures, revealing the architectures' scalability to manage changing urban demands. In real-time applications, this versatility is necessary to meet the high variability of urban scenes.

The overview demonstrates how edge computing and cloud computing enable the performance of detailed data regarding the operating conditions of a given section of the urban infrastructure. While edge computing provides relevant information directly regarding the systems and equipment, cloud computing's role is to manage big data resulting from the processing of edge devices. The architectural overview serves as the support for studying one use case in which both architectural designs are discussed and implemented, providing inputs and outputs of a model computed under multiple paradigms, thereby contributing to the selection of the preparatory architectural design best suited for real-time traffic simulators.

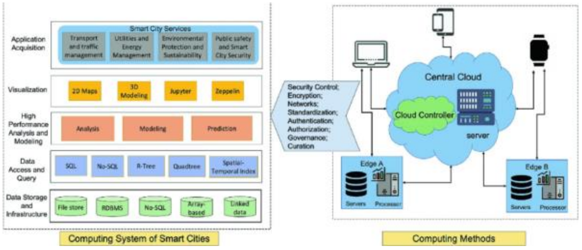
Smart City Services
Application Acquisition
Transport and traffic management
Utilities and Energy Management
Environmental Protection and Sustainability
Public safety and Smart City Security
Visualization
2D Maps
3D Modeling
Jupyter
Zeppelin
High Performance Analysis and Modeling
Analysis
Modeling
Prediction
Data Access and Query
SQL
No-SQL
R-Tree
Quadtree
Spatial-Temporal Index
Data Storage and Infrastructure
File store
RDBMS
No-SQL
Array-based
Linked data
Computing System of Smart Cities
Security Control; Encryption; Networks; Standardization; Authentication; Authorization; Governance; Curation
Central Cloud
Cloud Controller
server
Edge A
Edge B
Servers
Processor
Computing Methods

Fig 2 : Computing Architecture for Smart Cities

3. Real-Time Data Processing in Smart Cities

Real-Time Data Processing in Cities and Smart Cities 3.2. Real-Time Data Processing for Smart City Applications Smart cities are urban ecosystems intended to facilitate the lives of millions of citizens. Due to the technological advances in urban infrastructure, multiple and distributed electronic devices are capturing and transmitting an increasing amount of smart context or environmental data at the edge of the communication infrastructure, such as smart land vehicles, smartphones, public transportation, or closed-circuit television cameras. Smart city applications and services are highly demanding in terms of low latency of results against the analytics of these distributed datasets. This may be attributed to the nature of smart environments. Specifically, smart city infrastructures demand data management characteristics that meet the following requirements: manage unpredictable and highly diverse distributed and redundant datasets; manage continuous dynamic data; and provide low-latency results by the dual model data-state processing paradigms against analytics or reporting.

Various kinds of data processing are portable and transferable from cloud systems to the edge. These techniques operate locally and temporarily by the requirements of introduced smart scenarios. In the following, we will discuss various real-time data-processing strategies in smart living scenarios. The coexistence of edge and cloud computing creates the potential to realize the promises of both paradigms. By providing a cloud backend as well as edge servers in between them, real-time data processing, machine learning, and online model updates are performed in a large-scale IoT testbed. Under this operational scenario, we experimentally show that the minimization of selected quality of service indicators improves significantly in comparison with the cloud-only mode.

3.1. Challenges and Requirements

In smart cities, numerous IoT devices steadily generate a large volume of heterogeneous data. Such a huge volume of data can often overwhelm existing data processing infrastructure. In addition, the high data generation rate might further stress a system where the data processing rate might not match the generation rate. In such a scenario, the system may start dropping data to prevent the exhaustion of resources, which ultimately leads to performance issues, suboptimal system efficiency, and decreased data quality. Although efforts are being undertaken to investigate and integrate state-of-the-art computations, the integration with the existing real-time processing workflow and legacy infrastructure is not seamless and is often suboptimal. Moreover, prioritizing the velocity, quantity, and good data quality is of primary concern, as delays to data insights can result in incorrect and ineffective real-time data-driven decision-making.

Data privacy and consent are among the ethical aspects that require special attention. Workflow models and algorithms need to consider aspects such as consent, the use of personal data, and data protection, ensuring safe and secure data processing. The verification of algorithms and operation of edge processing frameworks in the smart city scenario, with the verified safety case and use of standard frameworks, makes it easier and less risky to integrate with multiple providers in different domains, enabling technology interworking and market platforms. In smart city IoT and distributed smart systems, data may be heterogeneous with various formats, structures, and semantics. To process such data, redundant and time-consuming data conversion will likely be needed. The large geographical distribution of devices producing and consuming data further exacerbates this problem. To build a smart city with future-proof technology, it is important to design a processing solution in which its components and topology are horizontally scalable so that they can be dynamically replicated or replaced in response to changing demand or faults, with minimal operator intervention. Thus, data processing architecture, microservices, and workflows with scalable algorithms and lightweight processes such as micro-batching or single-event processing are advantageous.In the context of smart cities, the proliferation of IoT devices generates vast amounts of heterogeneous data, often overwhelming existing data processing infrastructures. This rapid data generation can lead to a mismatch with processing rates, resulting in data loss and performance degradation that undermine system efficiency and data quality. While advancements in computational techniques are being explored, integrating these solutions with legacy systems remains a challenge, particularly when prioritizing the velocity and accuracy of insights for effective real-time decision-making. Additionally, ethical considerations around data privacy and consent must be addressed, requiring algorithms and workflows to incorporate robust data protection measures. The diverse formats and semantics of the data further complicate processing, often necessitating cumbersome conversions. To establish a resilient and future-proof smart city, it is essential to develop a horizontally scalable data processing architecture. This architecture should support dynamic replication and replacement of components with minimal operational intervention, utilizing microservices and lightweight processing methods like micro-batching or single-event processing to efficiently handle the complexities of distributed smart systems.

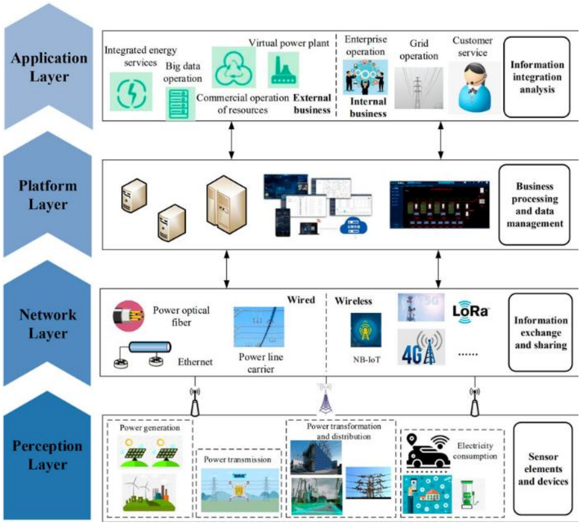

Fig 3 : Edge Computing Challenges in Ubiquitous Power Internet of Things

3.2. Use Cases and Applications

This subsection presents cases proving the significance and potential of efficient real-time data processing. Here, the technological solutions or platforms used in each application are presented, and part of them shows their integration in pilot cases or deployment in different cities. The section demonstrates different approaches. Public transportation in smart cities is one of the main targets for project and solution developers. However, these studies overlook the overall system when creating a smart transportation system, focusing on specific, segmented subsystems. We highlight which direction the study projects should take to create a truly smart environment. The subsection presents real use cases. Some of them are real pilot cases, where a technological solution is tested and integrated within a variety of equipment, and some others are not tested in place, such as case studies. The focus is on audiovisual surveillance mainly.

As presented in the intrinsic challenges of cities, this section shows some pilot cases mainly in public transportation environments where collaborative computer infrastructure is needed to support the smart cities concept. Waste management, environmental monitoring, and public security are also real-life examples. The effectiveness of applying such systems can be demonstrated by discussing the prospective solutions for this type of use case before and after implementation, along with the functionalities. Quite often, sensors and video cameras are installed at significant points for the transport system, public security, or environmental monitoring. Data collected can be easily cyphered to enhance it and transform it into valuable information for transport and urban management. This information, if processed in real time, can be used for control applications.

Equ 2: Computing Infrastructures

$$\frac{1}{D} \leq X(N) \leq \min\left(\frac{N}{D}, \frac{1}{D_{max}}\right) \qquad \max(D, ND_{max}) \leq R(N) \leq ND$$

$$\downarrow$$

$$\frac{N}{D+(N-1)D_{max}} \leq X(N) \leq \min\left(\frac{1}{D_{max}}, \frac{N}{D+(N-1)D_{avg}}\right)$$

$$\max(ND_{max}, D+(N-1)D_{avg}) \leq R(N) \leq D+(N-1)D_{max}$$

$$N^* = D/D_{max} \qquad N^+ = \frac{D-D_{avg}}{D_{max}-D_{avg}}$$

The throughput must be computed, the response time is bounded by lines passing through $(1, D)$ and $(0, D-D_{avg})(0, D-D_{max})$

4. Integration of Edge and Cloud Computing

Edge and cloud computing were introduced as two coexisting, distinct data processing environments. Both of these environments have specific advantages, such as high computational power and large storage spaces in cloud computing, or low latency and reduced communication overhead in the edge. Interestingly, together they present a synergy of mutual advantages: the low latency of the edge combined with the high storage of the cloud. The proper selection of data processing destinations results in minimizing response delay hidden by cutting-edge network interworking. Average performance indices are improved because of the possibility of offloading a part of the workload to the fast nodes with high processing power and the ability to serve.

The presented research is a step-by-step review of how the optimization necessitated the integration of edge and cloud computing. Moreover, it addresses one of the important aspects related to a modern communications interface between different nodes of IoT systems, smart cities, and big data applications. Dataflow management has to solve many classic problems defined by distributed systems. Among these problems, the most basic is ensuring the consistency of information storage, which is still especially significant in the case of data streams, which usually have time-relevant context data. Models: integration and execution, which are used to process data generated both in the edge environment and on the cloud side. Each of them has its features, strengths, and weaknesses. The selection of the appropriate model should be based on the developed smart city, used infrastructure, as well as mobile devices operating within the area. When making this selection, it is assumed that mobile devices at the edge must communicate flawlessly with resources located in the cloud. A single, holistic overview is thus used to develop an optimal arrangement for data processing in smart cities.

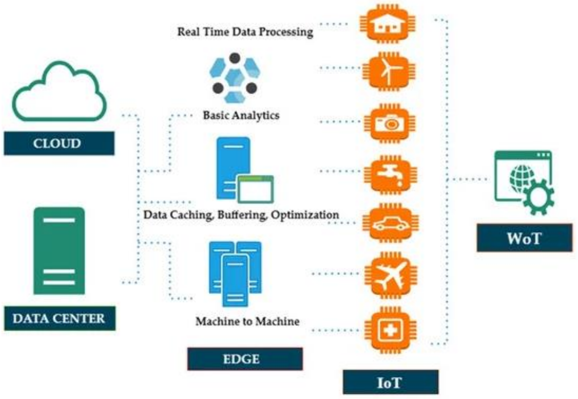
Real Time Data Processing
Basic Analytics
Data Caching, Buffering, Optimization
Machine to Machine
CLOUD
DATA CENTER
EDGE
IoT
WoT

Fig 4 : Integration of Edge and cloud Computing

4.1. Advantages and Limitations

There are several advantages to integrating the edge and cloud computing paradigms in smart cities. When edge appliances process data closer to where the sources are, low latency can be achieved. Additionally, cloud support of the edge processes can enhance the reliability of the integrated architecture. Importantly, the cloud offers additional resources for processing and storing more data than an edge infrastructure could. The physical infrastructure of the edge can be potentially location-dependent and can be more difficult to maintain, while it can be more flexible and easier to scale a cloud infrastructure. The cloud provides a more flexible and scalable infrastructure based on the demand for the services. Indeed, only a subset of cities is employing citizens' data against the fullness of the operations run on IoT devices. The cloud is crucial to managing the financial costs of maintaining a full processing infrastructure on the edge. As edge appliances become infeasible to maintain, the cloud can temporarily take over.

The IoT network continuously generates large amounts of data, and cloud storage capabilities are required by the traffic flow to avoid data losses. Different configurations are using the IoT layer, compared in terms of security and functionalities, but when cities try to transform IoT-generated data into value-added services, additional cloud storage capabilities are mandatory. In these scenarios, data can be stored in the cloud for a longer period, while citizens' IoT-generated data can be processed using fog and edge functionalities. Since the time required to bring new processing capabilities of the edge into effect can be long, a trade-off between the capabilities of the edge and cloud infrastructure should be defined in advance. When using the edge capabilities for processing, the time required to bring data into a real-time database might increase and could affect performance. In a data-drift scenario that requires long-term data consistency, eventually, all task activities could be moved to the cloud. The integration of the edge and the cloud affects the entire architecture. There are scenarios where a complete architecture must be developed from scratch to build and integrate the services on the edge and cloud sides. In others, the bottom layer provides software functionalities, enabling the development of higher-level services on the cloud. In both cases, it is necessary to define the data flows between the edge and the cloud. In any central storage architecture, including edge/cloud integration, the data should be reliable and complete. No partial writes should be performed to the main repository due to unexpected issues such as a misconfigured edge or pull-up issues by a superuser adversary. It is necessary to secure the connection from the edge, and securing all communication patterns and data routing patterns must be in the design. Backup and restore functionalities are required in case the edge completely fails. Data quality is also mandatory in any of the above architectures, and tasks like data consistency checks must be defined.

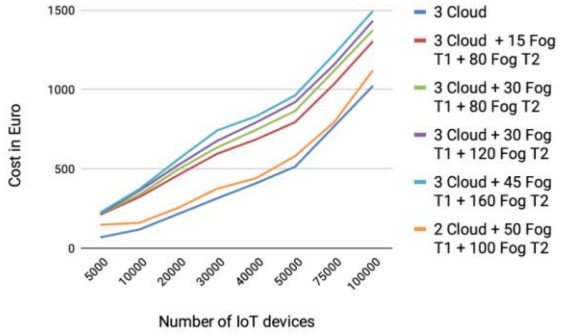

Fig : Edge computing paradigm

4.2. Key Technologies and Techniques

Several key components and techniques enable the integration of edge and cloud computing. At the core of these enablers are the involved protocols and communication standards. To combine their services, data exchange interfaces need to be established that allow data processing and communication between edge devices such as sensors or controllers and cloud infrastructures. Among the key networking protocols used in fog or edge computing are device layer protocols, message brokering, framework interfaces, and cloud-to-cloud integration tools. Containerization techniques and service architectures are intensively used in the development of IoT and monitoring tools by promoting modularity and ease of deployment at both the edge and the cloud.

Edge computing technologies are increasingly offering real-time and low-latency data processing and several approaches are applied to improve the effectiveness of local data processing, such as conducting edge analytics or implementing control algorithms at the edge. The utilization of container orchestration platforms in edge devices and the implementation of feedback mechanisms at the edge that can quickly adapt to changes in the cloud are further steps in orienting edge data processing. The application of learning and inference-based technologies like artificial intelligence and machine learning for orchestration and optimization of edge-cloud systems to reduce end-to-end latency is a very recent trend. Improved networking facilities and recent technologies like low-latency private networks are also triggering the development of different solutions that orchestrate processing at the edge and in the cloud. The integration of edge devices with cloud infrastructures and the technologies that are currently pursued in edge computing evoke the potential for further optimization of low-latency smart city applications including urban data processing.

5. Optimization Strategies

We explained the challenges of processing real-time data streams with required low latency in smart cities. Using edge and cloud computing in synergy as a back-end, we can overcome many of these challenges. In this section, we present strategies to integrate edge and cloud computing to optimize data processing in smart cities. These strategies will contribute to developing efficient, reliable, and low-latency systems for real-time processing applications in smart cities. Using these strategies, city planners and technologists can design more efficient smart city applications than what exists now. Here is the list of the basic strategies.

Resource allocation: Techniques for optimizing the current load among edge and cloud computing while maximizing the utilization of computing resources for real-time applications in smart cities. Task offloading: How and what amount of processing should be offloaded to the cloud or edge computing. Dynamic workload management: Techniques to respond to the fluctuating requests of users in real-time applications. A server can process a maximum number of tasks based on the current load. Dynamic workload management is linked to task offloading, and its strategies can be devised to exploit the benefits of processing data closer to the data source or routing data to the cloud computing platforms, depending on diverse factors such as application class, location, and load of servers in the cloud and edge. Resource Allocation – Environment Sharing In edge-cloud integrated systems, servers in the edge and cloud share the load of edge servers to offload processing in resource-constrained edge servers or to minimize the scaling cost of cloud computing. Edge-cloud environment sharing is tackled in terms of investigating effective server placements and load-balancing techniques. Research has shown improved system performance by selecting a subset of servers from a large distributed system for executing edge computing workloads, in conjunction with knowledge of the services' load profiles. In-cloud and out-cloud workload models are employed, generating server placements that can consistently improve system performance by 10%–70% for five tested load distributions. As the workloads executed on the edge platforms grow in size, the cost and complexity of maintaining infrastructure located close to the external edge greatly increase; this is further exacerbated if a micro-data center containing regional or local-specific content is situated on the premises. Even large edge sites typically require infrastructure that is efficiently replicated from a smaller number of cloud facilities. For this purpose, facilities located between the private edge environment and the cloud data centers draw on and adapt capabilities from both environments and are referred to here as adaptive environments. To further distinguish this connected infrastructure layered between the edge and cloud from the dedicated edge and cloud environments, we will use the terms connected edge and adaptive cloud to refer to the adaptive environment layers, respectively. It is, however, worth considering the implementation of multiple points of presence for the adaptive environments to form a resilient adaptability zone. Such points of presence could be in physically diverse locations and operate one or more adaptive layers that replicate data between them to further increase resilience against the loss of a single site.

Equ 3: Queuing Theory: Computing Delay Probability M/M/C

$$\begin{aligned}\Pi_W &= p_c + p_{c+1} + p_{c+2} + \cdots \\ &= \frac{p_c}{1-\rho} \\ &= \frac{(c\rho)^c}{c!}\left((1-\rho)\sum_{n=0}^{c-1}\frac{(c\rho)^n}{n!} + \frac{(c\rho)^c}{c!}\right)^{-1}.\end{aligned}$$

5.1. Resource Allocation and Task Offloading

In edge-cloud integrations, the major issue is how to ensure the best cooperation between edge devices and cloud infrastructure. It is clear that, in smart cities, the computational resources located at the edge will be finite; hence, they should be economically assessed. It would be a waste of computational power if the edge in each smart city application is waiting to complete its entire data processing or computational work before starting a query on the cloud. That is why some tasks (parts of the data to be analyzed) should be farmed off and processed from the cloud for most smart city applications.

When we talk about resource allocation in the edge-cloud environment, there are situations in which one would like to make use of not completely used edge devices but to present these devices only when the other edge devices are at maximum load, to maximize the usage of resources allocated at the edge in a cloud-agnostic way. Several approaches for deciding if a computation task should be performed by edge devices instead of sending everything to the cloud already exist, which is commonly known as offloading. The offloading decision is mainly based on two criteria: efficiency and response time. Based on the above criteria, offloading has mostly been solved as an optimization problem, in the form of a linear problem or mixed integer problem. Another aspect related to offloading is when we must take the offloading turn on or off based on the ability of the system. Many algorithms have been introduced based on Markov processes and queuing theory to manage on/off offloading. Moreover, continuous offloading decides to offload the currently processed data at any time. Authorities need to make timely decisions. That will have an impact on QoE, intending to minimize the delay of the worst case while optimizing the energy consumption of the mobile device performing offloading. Many algorithms require an offloading model based on the cumulative density function of the delay, so they depend on the real-case scenario. In such scenarios, real-time analysis can provide fast computational data that cannot be obtained at the processing time. Real-time analysis mainly depends on machine learning classification algorithms for predicting missing classification of the decision attributes, which is the delay in our research.

Resource Allocation System at the Edge: The issue of partitioning the edge processing resources in the application of fog computing for smart cities is addressed. The objective is to enhance the performance and availability of smart city services, which are partially managed at the edge of the network. The system is composed of several fog nodes, and the task is to distribute the service in an optimal way that considers both performance and energy-aware concerns. A weighted flow of the congestion controller method is used to manage the service assigning and offloading. This method is developed to balance the data processed in the cloud and data processed in the fog, to reduce latency. The system is composed of fog nodes, and the task is to distribute the service in an optimal way that considers both performance and energy-aware concerns. The objective is to find the best trade-off between the provider and offering computing functionalities located at the local nodes, which in this case act as fog nodes. The consumer side of the system is represented by an agricultural company that requests food quality analysis of their products, preferably performed in the cloud. Many smart city applications will deal with large PLC systems running embedded programming.In edge-cloud integrations, particularly within smart cities, optimizing cooperation between finite edge resources and robust cloud infrastructure is crucial. Efficient resource allocation is essential to prevent idle edge devices from delaying data processing, as these devices should not wait to complete tasks before querying the cloud. Instead, a strategic offloading approach allows for certain data segments to be processed in the cloud, especially when edge devices are under heavy load. The decision to offload tasks hinges on two key criteria: efficiency and response time, often framed as an optimization problem utilizing linear or mixed integer programming. Algorithms based on Markov processes and queuing theory can manage on/off offloading mechanisms, while continuous offloading enables real-time data processing. These timely decisions are vital for enhancing the quality of experience (QoE) by minimizing delays and optimizing energy consumption. In the context of fog computing, the partitioning of edge processing resources involves deploying multiple fog nodes to distribute services effectively, balancing performance and energy efficiency. A weighted flow congestion controller is employed to manage service assignments and offloading, ensuring a harmonious balance between cloud and fog processing to reduce latency. For instance, an agricultural company seeking cloud-based food quality analysis exemplifies how smart city applications can leverage these principles to enhance service delivery, especially when integrated with large PLC systems running embedded programming.

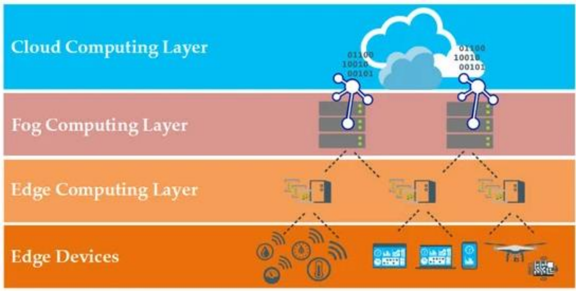

Fig 5 : Task Allocation Methods and Optimization Techniques in Edge Computing

5.2. Dynamic Workload Management

A key strategy to optimize real-time data processing and minimize the time to provide results is dynamic workload management. In smart cities, where applications from different domains must be supported, including natural disaster monitoring or real-time traffic management—and substantial variations occur throughout the day—which may result in substantial traffic spikes, it is essential to allocate and possibly reallocate resources in response to varying demands. Given the unexpected nature and unpredictability of these scenarios, the workload cannot only be dependent on the current time but also on the day and the current detected trends in the smart city and implemented relaxation strategies. Effectively managing the workload will ensure the seamless balancing of the load between edge devices and cloud processing and that sufficient computational resources are available at the edge to host the incoming applications ranked according to multi-criteria.

There are a variety of methodologies for monitoring workloads and predicting trends. Two main trends may be identified, which correspond in a certain sense to two of the three levels in big data analytics. A first trend considers only the current and the immediate future and aims at reducing the start-up time for scaling actions. A second trend concentrates on having a long-sighted perspective and prefers to take such measurements in advance to reduce or avoid any need to perform (or even to schedule) the scaling actions in the short term. Elastic scaling can also be used at the edge to allocate resources based on dynamics that occur at the data center or host side. Indeed, besides failing to take into account the real user experience, the primary downside of the techniques presented above is not considering the network latencies, which can take around 200 ms per hop in mobile environments.

6. Conclusion

The objective of this contribution was to underline the integration of edge and cloud computing to address real-time data processing in smart cities, which is necessary to make applications focused on urban well-being available in real time. The combination of technological advancements, urban data growth, and related in-depth analysis of key challenges to large data processing efforts highlighted the limited adoption of edge stream processing capabilities in widespread sensing devices available today. The rise in various online data applications in many domains compelled researchers and practitioners to address a comprehensible set of challenges and propose innovative solutions to establish sustainable and secure integrated city services. To date, many of the proposed smart services are limited in resource allocation while managing real-time processing and quick data delivery sentiments as they perform further analysis based on server-side data computing. Thus, the optimization of real-time data processing mechanisms is necessary for the realization of maximum intelligence and immediacy relevance, among many applicable smart city offerings, so that the urban community can fully explore the path to urban well-being.

Future research exploration includes an in-depth transformation of big data architectures and data analysis tools to be employed on streaming analytics platforms based on intensive in-network processing. In addition to this, the advent of detailed scientific scenarios is considered given the continued technological advancements in the field of the Internet of Multimedia Things. Aided by the integration of IoT capabilities and multimedia data processing algorithms, there is a huge recent spread of smart embedded devices designed purposely to provide major video, visual, audio, and data source intensities. As the deployment of such devices is ubiquitously increasing, big opportunities are arising in using the recent expansion of real-time video analytics applications and services, beyond the traditional physical campuses and newsrooms-centric use scenarios. In essence, multimedia data contents with their context rich metadata may help city officials, inhabitants, and multiple stakeholders in decision-making and urban policy planning.

6.1. Future Trends: The recent proliferation of artificial intelligence systems, including machine and deep learning, is expected to shape standards, architectures, and practices in data management for smart environments. In particular, the customization of these systems can be supported by real-time data and benefit from edge architectures for real-time adjustment in response to data variation and unexpected events. In addition to prediction, used for resource distribution, overall optimization, and anomaly detection, AI embedded in edge systems supports advanced triggering of cloud-based activities, saving communication costs by aggregating data in low-latency edge nodes for pre-processing and decision-making.

One of the next steps in the evolution of communication networks in smart cities is the dependence on 5G, with deployments and testing ongoing worldwide. Built-in edge computing capabilities are expected to support low latency, mass IoT connections, and mobile connections with high data transmission rates. 5G is viewed as a major tool for integrating edge computing-enabled nodes to support increased connectivity, latency requirements, and platform variations, enabling low-latency application design. The transition to new standard networks enables low latency and packet loss connectivity, which is essential for the growth and operation of low-latency systems required for real-time applications. 5G will impact application design; thus, new or updated testbeds are expected to consider this standard for their system deployment.

7. References

[1] Avacharmal, R., Pamulaparthyvenkata, S., & Gudala, L. (2023). Unveiling the Pandora's Box: A Multifaceted Exploration of Ethical Considerations in Generative AI for Financial Services and Healthcare. Hong Kong Journal of AI and Medicine, 3(1), 84-99.

[2] Aravind, R. (2023). Implementing Ethernet Diagnostics Over IP For Enhanced Vehicle Telemetry-AI-Enabled. Educational Administration: Theory and Practice, 29(4), 796-809.

[3] Mahida, A. Explainable Generative Models in FinCrime. J Artif Intell Mach Learn & Data Sci 2023, 1(2), 205-208.

[4] Mandala, V., & Mandala, M. S. (2022). ANATOMY OF BIG DATA LAKE HOUSES. NeuroQuantology, 20(9), 6413.

[5] Perumal, A. P., Deshmukh, H., Chintale, P., Molleti, R., Najana, M., & Desaboyina, G. Leveraging machine learning in the analytics of cyber security threat intelligence in Microsoft azure.

[6] Kommisetty, P. D. N. K. (2022). Leading the Future: Big Data Solutions, Cloud Migration, and AI-Driven Decision-Making in Modern Enterprises. Educational Administration: Theory and Practice, 28(03), 352-364.

[7] Bansal, A. (2023). Power BI Semantic Models to enhance Data Analytics and Decision-Making. International Journal of Management (IJM), 14(5), 136-142.

[8] Laxminarayana Korada, & Vijay Kartik Sikha. (2022). Enterprises Are Challenged by Industry-Specific Cloud Adaptation - Microsoft Industry Cloud Custom-Fits, Outpaces Competition and Eases Integration. Journal of Scientific and Engineering Research. https://doi.org/10.5281/ZENODO.13348175

[9] Avacharmal, R., Sadhu, A. K. R., & Bojja, S. G. R. (2023). Forging Interdisciplinary Pathways: A Comprehensive Exploration of Cross-Disciplinary Approaches to Bolstering Artificial Intelligence Robustness and Reliability. Journal of AI-Assisted Scientific Discovery, 3(2), 364-370.

[10] Aravind, R., & Shah, C. V. (2023). Physics Model-Based Design for Predictive Maintenance in Autonomous Vehicles Using AI. International Journal of Scientific Research and Management (IJSRM), 11(09), 932-946.

[11] Mahida, A. (2023). Enhancing Observability in Distributed Systems-A Comprehensive Review. Journal of Mathematical & Computer Applications. SRC/JMCA-166. DOI: doi. org/10.47363/JMCA/2023 (2), 135, 2-4.

[12] Mandala, V. (2021). The Role of Artificial Intelligence in Predicting and Preventing Automotive

Failures in High-Stakes Environments. Indian Journal of Artificial Intelligence Research (INDJAIR), 1(1).

[13] Perumal, A. P., Deshmukh, H., Chintale, P., Desaboyina, G., & Najana, M. Implementing zero trust architecture in financial services cloud environments in Microsoft azure security framework.

[14] Bansal, A. Advanced Approaches to Estimating and Utilizing Customer Lifetime Value in Business Strategy.

[15] Sikha, V. K., Siramgari, D., & Korada, L. (2023). Mastering Prompt Engineering: Optimizing Interaction with Generative AI Agents. Journal of Engineering and Applied Sciences Technology. SRC/JEAST-E117. DOI: doi. org/10.47363/JEAST/2023 (5) E117 J Eng App Sci Technol, 5(6), 2-8.

[16] Avacharmal, R., Gudala, L., & Venkataramanan, S. (2023). Navigating The Labyrinth: A Comprehensive Review Of Emerging Artificial Intelligence Technologies, Ethical Considerations, And Global Governance Models In The Pursuit Of Trustworthy AI. Australian Journal of Machine Learning Research & Applications, 3(2), 331-347.

[17] Ravi Aravind, Srinivas Naveen D Surabhi, Chirag Vinalbhai Shah. (2023). Remote Vehicle Access:Leveraging Cloud Infrastructure for Secure and Efficient OTA Updates with Advanced AI. EuropeanEconomic Letters (EEL), 13(4), 1308–1319. Retrieved fromhttps://www.eelet.org.uk/index.php/journal/article/view/1587

[18] Mahida, A. (2023). Machine Learning for Predictive Observability-A Study Paper. Journal of Artificial Intelligence & Cloud Computing. SRC/JAICC-252. DOI: doi. org/10.47363/JAICC/2023 (2), 235, 2-3.

[19] Perumal, A. P., & Chintale, P. Improving operational efficiency and productivity through the fusion of DevOps and SRE practices in multi-cloud operations.

[20] Bansal, A. (2022). Establishing a Framework for a Successful Center of Excellence in Advanced Analytics. ESP Journal of Engineering & Technology Advancements (ESP-JETA), 2(3), 76-84.

[21] Korada, L. (2023). AIOps and MLOps: Redefining Software Engineering Lifecycles and Professional Skills for the Modern Era. In Journal of Engineering and Applied Sciences Technology (pp. 1–7). Scientific Research and Community Ltd. https://doi.org/10.47363/jeast/2023(5)271

[22] Avacharmal, R. (2022). ADVANCES IN UNSUPERVISED LEARNING TECHNIQUES FOR ANOMALY DETECTION AND FRAUD IDENTIFICATION IN FINANCIAL TRANSACTIONS. NeuroQuantology, 20(5), 5570.

[23] Aravind, R., & Surabhii, S. N. R. D. Harnessing Artificial Intelligence for Enhanced Vehicle Control and Diagnostics.

[24] Mahida, A. (2022). Comprehensive Review on Optimizing Resource Allocation in Cloud Computing for Cost Efficiency. Journal of Artificial Intelligence & Cloud Computing. SRC/JAICC-249. DOI: doi. org/10.47363/JAICC/2022 (1), 232, 2-4.

[25] Chintale, P. (2020). Designing a secure self-onboarding system for internet customers using Google cloud SaaS framework. IJAR, 6(5), 482-487.

[26] Bansal, A. (2022). REVOLUTIONIZING REVENUE: THE POWER OF AUTOMATED PROMO ENGINES. INTERNATIONAL JOURNAL OF ELECTRONICS AND COMMUNICATION ENGINEERING AND TECHNOLOGY (IJECET), 13(3), 30-37.

[27] Korada, L. (2023). Leverage Azure Purview and Accelerate Co-Pilot Adoption. In International Journal of Science and Research (IJSR) (Vol. 12, Issue 4, pp. 1852–1954). International Journal of Science and Research. https://doi.org/10.21275/sr23416091442

[28] Vehicle Control Systems: Integrating Edge AI and ML for Enhanced Safety and Performance. (2022).International Journal of Scientific Research and Management (IJSRM), 10(04), 871-886.https://doi.org/10.18535/ijsrm/v10i4.ec10

[29] Aravind, R., Shah, C. V &amp; Manogna Dolu. AI-Enabled Unified Diagnostic Services: Ensuring Secure andEfficient OTA Updates Over Ethernet/IP. International Advanced Research Journal in Science, Engineeringand Technology. DOI: 10.17148/IARJSET.2023.101019

[30] Mahida, A. Predictive Incident Management Using Machine Learning.

[31] Chintale, P. SCALABLE AND COST-EFFECTIVE SELF-ONBOARDING SOLUTIONS FOR HOME INTERNET USERS UTILIZING GOOGLE CLOUD'S SAAS FRAMEWORK.

[32] Bansal, A. (2021). OPTIMIZING WITHDRAWAL RISK ASSESSMENT FOR GUARANTEED MINIMUM WITHDRAWAL BENEFITS IN INSURANCE USING ARTIFICIAL INTELLIGENCE TECHNIQUES. INTERNATIONAL JOURNAL OF INFORMATION TECHNOLOGY AND MANAGEMENT INFORMATION SYSTEMS (IJITMIS), 12(1), 97-107.

[33] Korada, L., & Somepalli, S. (2023). Security is the Best Enabler and Blocker of AI Adoption. In International Journal of Science and Research (IJSR) (Vol. 12, Issue 2, pp. 1759–1765). International Journal of Science and Research. https://doi.org/10.21275/sr24919131620

[34] Shah, C., Sabbella, V. R. R., & Buvvaji, H. V. (2022). From Deterministic to Data-Driven: AI and Machine Learning for Next-Generation Production Line Optimization. Journal of Artificial Intelligence and Big Data, 21-31.