SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V11I3P116

Volume 11 Issue 3, 160-175, March 2024
© 2024 Seventh Sense Research Group®

Original Article

Adaptive Fog Computing Framework (AFCF): Bridging IoT and Blockchain for Enhanced Data Processing and Security

K. Vinod Kumar Reddy[1], Vasavi Bande[2], Novy Jacob[3], A. MallaReddy[4*], Sk Khaja Shareef[5], Sriharsha Vikruthi[6]

[1]Department of Computer Science Engineering (Artificial Intelligence), G Pullaiah College of Engineering and Technology, Andhra Pradesh, India.
[2]Department of IT, MVSR Engineering College, Telangana, India.
[3]Department of AIML, CMR Institute of Technology, Karnataka, India.
[4]Department of Information Technology, CVR College of Engineering, Telangana, India.
[5]Department of Information Technology, MLR Institute of Technology, Telangana, India.
[6]Department of CSE, Pace Institute of Technology and Sciences, Andhra Pradesh, India.

[*]Corresponding Author : mallareddyadudhodla@gmail.com

Received: 14 January 2024 Revised: 18 February 2024 Accepted: 16 March 2024 Published: 31 March 2024

Abstract *- This research introduces an Adaptive Fog Computing Framework (AFCF) aimed at enhancing the efficiency and scalability of IoT ecosystems through blockchain technology integration, addressing task allocation, resource management, and task offloading challenges within fog and cloud computing paradigms. Employing simulations, the study utilized task distribution strategies, blockchain stability assessment, and cloud server workload management, demonstrating the framework's capacity to maintain performance across diverse IoT settings. Quantitative results revealed a cent percent success rate in task processing, with a balanced load distribution at 50% and an average task complexity of 6.47893 in arbitrary units. The system demonstrated a latency of 48.479 milliseconds and a throughput of 2.469697 tasks per timestep, showcasing high scalability (97.8%) and energy efficiency (15.774194 in arbitrary efficiency units), emphasizing the AFCF's robustness in varied tasks and resource dynamics. The study concludes the AFCF's potential for real-world IoT applications, highlighting its implications for future research and practical deployment, underscoring its contribution to fog and cloud computing literature and paving the way for further exploration into adaptive computing frameworks.*

Keywords *- Fog computing, IoT ecosystem, Blockchain technology, Task allocation, Resource management, Task offloading, Simulation, Cloud computing, Edge computing, Data integrity, Scalability.*

1. Introduction

In the contemporary digital landscape, the proliferation of the Internet of Things (IoT) has ushered in an era of transformative connectivity, where everyday objects intercommunicate vast data streams, heralding new automation and analytics opportunities across sectors. This evolution, however, presents significant challenges, notably in processing speed and data management [1] due to the immense volume and velocity of data from IoT devices [2].

Traditional cloud computing models with centralized data processing increasingly face bottlenecks, including latency, bandwidth limitations, and centralized processing challenges. Against this backdrop, fog computing, with its decentralized architecture processing data closer to its source, emerges as a strategic solution, promising timely and efficient responses essential for real-time IoT applications. The introduction of the Adaptive Fog Computing Framework (AFCF) in this paper aims to overcome these limitations, providing a balanced, secure, and scalable infrastructure tailored to the dynamic needs of IoT ecosystems. By decentralizing data processing and enhancing computational power near the network's edge [3], the AFCF addresses critical IoT challenges, including improved data management, enhanced security, and reduced latency, setting the stage for a detailed exploration of its design principles and transformative potential for IoT infrastructure [4].

The study articulates significant challenges faced by existing IoT systems, primarily stemming from the limitations of conventional cloud computing models in handling the vast data volumes generated by IoT devices. These challenges include inadequate data handling capacities and latency issues that can severely impair real-time data processing capabilities. Such limitations are not merely obstacles but represent critical challenges confronting existing architectures, often exacerbated by security vulnerabilities and scalability concerns.

The current systems fall short in several key areas, notably their inability to process large volumes of data at the edge, a deficiency in real-time analytics, and insufficient mechanisms to ensure data integrity across distributed networks. These gaps highlight the pressing need for a framework that is not only more responsive but also capable of intelligent decision-making and autonomy in dynamic environments, underscoring the imperative for adaptive and intelligent frameworks. These frameworks must discern and respond to fluctuating data streams and workloads while anticipating future demands, marking a clear innovation imperative to develop solutions that can learn, adapt, and scale without compromising performance or security.

This research aims to introduce an Adaptive Fog Computing Framework (AFCF) that adeptly manages the computational and storage demands of IoT devices. The key insights revolve around the framework's capability to balance the load across fog nodes and cloud servers dynamically, the strategic offloading of tasks based on complexity, and the integration of blockchain technology to fortify data integrity and security. The core insights of this research are encapsulated in the following points:

1. Dynamic Load Balancing: The framework ensures that computational tasks are distributed equitably among fog nodes, preventing any single node from becoming a bottleneck, thereby enhancing system reliability and efficiency.
2. Strategic Task Offloading: By assessing task complexity, the system intelligently decides whether to process data locally on fog nodes or offload it to the cloud, optimizing resource utilization and reducing latency.
3. Blockchain Integration: Incorporating blockchain technology within the AFCF provides a secure and immutable ledger for transactions, enhancing data security and trust in a distributed network.
4. Resource Optimization: The AFCF employs algorithms that dynamically adjust resources in response to fluctuating demands, ensuring optimal performance without resource wastage.
5. Scalability and Flexibility: Designed with scalability in mind, the framework can accommodate an increasing number of IoT devices and adapt to various application requirements.
6. Real-World Viability: The simulation results in point towards the practical applicability of the AFCF in real-world scenarios, signaling its readiness for deployment in industrial settings.

The Adaptive Fog Computing Framework (AFCF) introduced in this research represents a cutting-edge synthesis of modern technological innovations and architectural designs. It aims to blend the benefits of both fog and cloud computing to enhance IoT systems.

Featuring a layered architecture, it ensures smooth integration across IoT devices, fog nodes, and cloud services, designed not only to meet current but also future IoT demands. This dynamic and resilient system is tailored to adapt to changing data patterns and network structures.

The subsequent sections will detail the AFCF's methodology, including its technical framework, simulation setup, and performance indicators, followed by an analysis of empirical findings that shed light on its operational efficiency and practical value. Concluding remarks will summarize the study's contributions to IoT and fog computing scholarship, recognizing limitations and suggesting avenues for further exploration, positioning the AFCF as a foundational model for future computing infrastructures.

2. Background

The advent of the Internet of Things (IoT) has redefined the boundaries of connectivity, leading to an intertwined web of devices and data streams. This intricate fabric of digital communication has brought forth an array of computational challenges and opportunities.

The background section of this paper embarks on a journey through the evolutionary landscape of IoT ecosystems, tracing the arc from the inception of connected devices to the current epoch where data is king [5]. It explores the paradigmatic shift towards fog computing as a salient response to the burgeoning data management needs.

This section also delves into the pivotal role of task allocation and the revolutionary impact of blockchain technology on IoT security and integrity. It concludes with an examination of scalability in the ever-expanding IoT networks and highlights the research gaps that catalyze the need for innovation [6].

2.1. Evolution of IoT Ecosystems

The Internet of Things (IoT) has evolved significantly, necessitating solutions for security, privacy, and data analysis challenges [7, 8]. Combining blockchain and AI within IoT systems enhances data security, trust, and analysis capabilities. Alternative approaches include developing end-to-end security models for IoT ecosystems and studying competition among multi-platform IoT ecosystems [8].

Additionally, exploring horizontal integration in IoT business models across industries offers new opportunities for comprehensive solutions. These diverse strategies collectively shape the future of IoT and its impact on various sectors.

2.2. The Rise of Connected Devices

The proliferation of interconnected devices has had a profound impact across various domains, encompassing healthcare, engineering, and communication. In the context of pandemic management, there is the exploration of employing connected devices and social machines to implement predictive, preventive, and personalized medicine [9]. In the realm of engineering, there's been a proposal to apply Multi-Tuned Mass Damper Inerter (MTMDI) systems to adjacent high-rise buildings, serving as an unconventional seismic protection strategy, underlining the influence of connected devices on structural safety and performance.

Furthermore, the field of communication engineering has witnessed the development of smart glasses and other wearable smart devices, which are increasingly integrating with other devices through diverse communication technologies [10]. The proliferation of these connected devices has also resulted in a surge in data volumes, necessitating the adoption of fog computing for time-critical control applications and the application of reinforcement learning to optimize network pathways while ensuring reliable transmission times.

The interconnected nature of these devices has not only brought about a revolution in various industries but has also introduced new challenges and opportunities for innovation. As the number of connected devices continues to burgeon, their influence on various facets of society is expected to become even more pronounced.

2.3. Data Deluge: Challenges in the IoT Landscape

The Internet of Things (IoT) has resulted in a substantial increase in data volume, presenting challenges across various sectors. Within the telecommunications industry, IoT applications generate vast quantities of data, ranging from terabytes to petabytes daily. This data surge has necessitated the utilization of Big Data Analytics (BDA) to extract actionable insights from telecom big data [11].

Similarly, in the realm of smart cities, there's a growing focus on edge-AI-enabled video analytics, owing to its transformative potential. This technology enables IoT devices with constrained resources to shift compute-intensive AI tasks to network edge servers, offering enhanced latency and bandwidth efficiency. The applications of video analytics in smart cities encompass security, surveillance, transportation, traffic management, healthcare, education, sports, and entertainment [12].

2.4. The Paradigm Shift from Centralized to Distributed Computing

The transition from centralized to distributed computing has been primarily motivated by the growing demand for high-bandwidth and low-latency applications bolstered by advancements in communication and computational capabilities of embedded devices [13, 14]. This shift has reverberated across various domains, encompassing network coding, edge computing, and the Internet of Things (IoT). Key facets of this paradigm shift include:

1. Device-to-Device Communications: A novel paradigm in network coding known as instantly decodable network coding has arisen, offering a trade-off between performance and complexity. In device-to-device communication networks, devices expedite the recovery of missing data packets by exchanging network-coded packets.
2. Edge Mesh: This emerging computing paradigm delegates decision-making tasks to edge devices within the network rather than funneling all data to a centralized server. Data and computational tasks are distributed through a mesh network of edge devices and routers, delivering advantages such as distributed processing, low latency, fault tolerance, enhanced scalability, improved security, and privacy.
3. Distributed Edge Computing: The demand for high bandwidth in conjunction with low-latency applications has precipitated the shift from centralized cloud computing to distributed edge computing. This transformation has a substantial impact on the design of network interconnects and the fundamental network attributes necessary to fully enable 5G and beyond.
4. Computing Paradigm Shift: Over the past five years, a noteworthy shift from centralized to distributed computing has transpired. While timesharing and batch systems still find utility, the dominant model of large mainframes has ceded ground. Networks have democratized computing power, accessibility, and cost, extending beyond centralized computer facilities, with personal computers opening up computing to a wider user base.
5. The transition from centralized to distributed computing offers a host of advantages, including heightened performance, reduced complexity, and increased adaptability. These advancements are particularly pertinent for critical applications demanding heightened reliability, real-time processing, support for mobility, and context awareness.

2.5. The Emergence of Fog Computing

In the realm of computing paradigms, fog computing, also known as edge computing, emerges as a concept that extends the capabilities of distributed computing, notably cloud computing, to the network's edge. Fog computing delivers information, processing, storage, and application services directly to end-users, drawing inspiration from real-time applications such as smart grids, intelligent traffic signals in vehicular networks, and programmable networks [15]. It addresses several challenges associated with traditional cloud computing, including unreliable latency, a lack of mobility support, and the need for location awareness.

A comparative analysis between fog computing and cloud computing underscores their distinctive attributes. While both offer similar services, they diverge significantly in terms of location, latency, and scalability. Cloud computing predominantly relies on centralized data centers, whereas fog computing pushes computing services to the network's edge, proximate to end-users.

This proximity affords fog computing an advantage in terms of reduced latency, facilitating faster response times compared to cloud computing, which may encounter delays due to data transmission limitations. Furthermore, fog computing has the potential to alleviate scalability bottlenecks that can plague cloud-based systems, as it actively manages and reduces data traffic load towards the central cloud [16].

1. The advantages of fog computing in the context of Internet of Things (IoT) systems are noteworthy. First and foremost, it significantly reduces latency, ensuring rapid response times for IoT devices by processing data locally at the network's edge. Additionally, fog computing enhances the Quality of Service (QoS) in IoT-based systems through efficient offloading algorithms, thereby improving the overall experience for end-users. Scalability is another strength, as fog computing efficiently accommodates the growing number of IoT devices by managing and reducing data traffic load towards the central cloud. Finally, fog computing is a cost-effective solution, reducing the need for extensive infrastructure and network resources by processing data locally at the network's edge, ultimately leading to substantial cost savings.
2. In summary, fog computing represents a promising technology for IoT systems, offering distinct advantages over traditional cloud computing, such as reduced latency, enhanced QoS, scalability, and cost-effectiveness, making it a pivotal player in the evolving landscape of distributed computing [17].

2.6. IoT Data Management and Processing

In the realm of IoT applications, real-time data processing assumes a critical role by facilitating swift decision-making and responsive actions. Several paramount requirements for real-time data processing in IoT are evident: IoT devices amass substantial volumes of real-time data, necessitating processing and analysis to derive meaningful insights that inform decision-making processes. Real-time data analysis is imperative for monitoring and optimizing diverse IoT applications, spanning domains such as building energy management, manufacturing, and healthcare.

Safeguarding the integrity and privacy of IoT data looms large as a significant concern, given its sensitive nature and susceptibility to security threats. Techniques such as Trusted Execution Environment (TEE) and end-to-end data encryption mechanisms emerge as viable measures to uphold data privacy. The management of data at the edge introduces both opportunities and challenges within IoT applications: Edge computing accelerates data processing by gathering and analyzing data closer to its source, thus curtailing latency and augmenting responsiveness. Given the typically constrained storage capacity of IoT devices, edge-based data management serves to diminish reliance on cloud-based storage, which may raise security and privacy apprehensions.

Edge computing furnishes the capability for real-time data processing, an indispensable feature for numerous IoT applications, including intelligent building management and healthcare. Nonetheless, managing data at the edge can introduce novel security and privacy quandaries, potentially exposing sensitive data to unauthorized access or malicious exploits. Techniques such as blockchain and smart contracts can be enlisted to bolster data security and integrity. Data analytics emerges as a pivotal cog in the machinery of IoT applications: Real-time data collection by IoT devices mandates subsequent processing and analysis to distill valuable insights conducive to informed decision-making.

Data analytics encompasses the processing and scrutiny of IoT data, extracting valuable insights and patterns that underpin the optimization of various IoT applications and enhancement of user experiences. As with other facets of IoT data management, ensuring the security and privacy of IoT data remains a foremost concern, given its sensitivity and vulnerability to threats. Techniques such as Trusted Execution Environment (TEE) and end-to-end data encryption mechanisms can be deployed to preserve data privacy. Effective data management strategies are pivotal in the realm of IoT, encompassing facets like data aggregation, storage, and dissemination. These strategies ensure data accuracy, consistency, and accessibility to authorized users [18].

The domains of real-time data processing, edge-based data management, and data analytics hold pivotal roles in the realm of IoT applications. Addressing challenges such as data security, latency reduction, and responsiveness enhancement while harnessing opportunities are pivotal imperatives in the effective management and processing of IoT data. Techniques including blockchain, smart contracts, and edge computing serve as valuable tools in navigating these challenges and capitalizing on the opportunities presented by the burgeoning field of IoT data.

2.7. Task Allocation and Load Balancing

In distributed systems, including Internet of Things (IoT) networks, the concepts of task allocation and load balancing play pivotal roles. Task allocation strategies involve the judicious assignment of tasks to various nodes within the network. At the same time, load balancing aims to distribute the computational workload evenly across these nodes to prevent the undue burdening of any single node. The efficient implementation of task allocation and load balancing mechanisms holds great significance for IoT performance, particularly in terms of diminishing processing latency and enhancing overall quality of service. Recent research endeavors have delved into these critical aspects:

1. In the context of multi-UAV-aided Mobile Edge Computing (MEC) systems, there is an exploration into joint optimization of task offloading, resource allocation, and load balancing.
2. Software-defined networks are benefitting from load balancing strategies, where researchers have devised enhancements to the whale optimization algorithm to achieve improved load distribution.
3. An enhanced butterfly optimization algorithm is proposed to address load-balancing concerns with a specific focus on reducing latency.
4. To optimize MEC systems with multiple service providers, a two-layer task offloading scheme is introduced, contributing to efficient task allocation.
5. Energy-efficient task offloading, load balancing, and resource allocation strategies are investigated in the context of mobile edge computing-enabled IoT networks.
6. These studies introduce a range of algorithms and optimization techniques aimed at enhancing task allocation and load balancing within distributed systems. The ultimate objective is to mitigate latency issues and elevate the quality of service, making them invaluable contributions to the realm of IoT network performance.

2.8. Blockchain Technology in IoT

Blockchain technology, often associated primarily with cryptocurrencies such as Bitcoin, represents a decentralized and distributed ledger system that securely manages data across a network of nodes. Beyond its role in digital currencies, blockchain offers a range of advantages, including heightened security, transparency, and trustworthiness, with applications extending into diverse industries, including the Internet of Things (IoT) [19].

The integration of blockchain technology holds the potential to substantially bolster the security of IoT networks by effectively addressing key challenges related to data privacy, authentication, and trust. Several ways in which blockchain can enhance IoT security are as follows:

Data Privacy: Blockchain’s decentralized architecture and data distribution across multiple nodes establish formidable barriers against unauthorized access or tampering with sensitive information in IoT networks. This robust data protection mechanism safeguards against cyber threats and maintains data integrity within IoT ecosystems.

Authentication: Blockchain can play a pivotal role in creating unassailable, tamper-proof identities for IoT devices, ensuring that only authorized devices can gain access to network resources. This authentication safeguard is instrumental in thwarting unauthorized access and upholding overall IoT security.

Trust: The transparency and security inherent in blockchain technology lay the foundation for trust between IoT devices and users. By providing a secure and transparent platform for data storage and exchange, blockchain fosters confidence in IoT networks, thus encouraging wider adoption and utilization.

Smart Contracts: Blockchain’s capability to facilitate smart contracts-self-executing agreements triggered by predefined conditions can streamline and automate various processes within IoT networks. This automation extends to tasks like data exchange, payments, and device interactions, enhancing efficiency and security [20].

Integration with other Technologies: Blockchain can be effectively integrated with emerging technologies such as artificial intelligence and federated learning. This synergy addresses challenges related to privacy preservation, large-scale data management, and computational demands in IoT networks. Industries like vehicular networks and healthcare, in particular, stand to benefit from this collaborative approach.

Blockchain technology holds the potential to significantly bolster the security of IoT networks by furnishing a decentralized, transparent, and reliable platform for data management and exchange. In doing so, it adeptly tackles primary IoT challenges, including data privacy, authentication, and trust, while opening up new horizons for various industries and applications [21].

2.9. Scalability Concerns in IoT Networks

IoT networks grapple with a multitude of scalability challenges, encompassing various dimensions like energy efficiency, security, and network performance. The ensuing sections delve into diverse approaches aimed at mitigating these challenges:

- Scaling IoT: A Multidimensional Challenge: IoT networks confront an array of hurdles, including heterogeneity, scalability, and energy efficiency concerns. While conventional methods such as Software Defined Networking (SDN) prove effective for larger-scale IoT implementations, they can be inefficient for smaller-scale applications, prompting the exploration of alternative solutions.
- Addressing Scalability with Fog Computing: Fog computing emerges as a promising strategy for combatting scalability issues within IoT networks. This approach entails the processing and analysis of data at the network’s edge, diminishing the necessity for extensive data transmission to the cloud and resulting in improved response times.

- Future Trends: Scalable Architectures for Expanding IoT Networks: As the expanse of IoT networks continues to burgeon, novel architectures and technologies are under scrutiny to ensure scalability and resource optimization. These forthcoming trends encompass:
- Dynamic Virtual IoT Networks: A nimble mechanism that empowers users to dynamically reconfigure communication flows between sensors and actuators based on data traffic loads and optimal routing [22].
- Blockchain-Enabled IoT Networks: Blockchain technology is harnessed to bestow secure management, authentication, and access control upon IoT devices, thus enhancing data integrity and privacy.
- Energy-Efficient Clustering Protocols: Solutions such as Enhanced Multitier Energy-Efficient Clustering Protocol Integrated with Internet of Things (EMEECP-IOT) address energy efficiency challenges in IoT-constrained Wireless Sensor Networks (WSN) [23].
- Lightweight Heterogeneous Multihomed Networks: A model tailored to contend with the complex heterogeneity network landscape in smart cities, facilitating the coexistence of diverse IoT applications and technologies [24].
- Blockchain-Based Security and Scalability Solutions: The utilization of hyperledger fabric, an enterprise-grade permissioned distributed ledger framework, as a means to resolve security and scalability issues within IoT networks. These imminent trends collectively aim to surmount the challenges faced by IoT networks concerning scalability, energy efficiency, and security. In doing so, they aspire to ensure seamless connectivity and efficient resource utilization across various applications and environments.

2.10. Research Gap and Need for Innovation

In the realm of IoT, the pace of technological advancement is relentless, yet it often outstrips the capabilities of current infrastructures and systems. As we navigate through the complex tapestry of interconnected devices and ever-growing data, it becomes clear that existing solutions are struggling to keep up.

This section aims to scrutinize the current landscape, pinpointing the limitations and inefficiencies that hinder progress.

2.10.1. Surveying the Current Landscape: Limitations of Existing Solutions

A meticulous examination of the prevailing systems reveals a series of constraints ranging from inadequate data handling capacities to latency issues that can cripple real-time data processing. These limitations are not mere stumbling blocks but are critical challenges that existing architectures face, often compounded by security vulnerabilities and scalability concerns.

2.10.2. Identifying the Gaps: Where Current Systems Fall Short

The gaps in current systems manifest in several key areas: the inability to process large volumes of data at the edge, a lack of real-time analytics, and insufficient mechanisms to ensure data integrity across distributed networks. These shortcomings underscore the pressing need for a framework that is not only more responsive but also capable of intelligent decision-making and autonomy in dynamic environments.

2.10.3. The Imperative for Adaptive and Intelligent Frameworks

The exigencies of modern IoT applications demand frameworks that are both adaptive and intelligent. Such systems must not only discern and respond to fluctuating data streams and workloads but also anticipate future demands. The innovation imperative is clear: to develop solutions that can learn, adapt, and scale without compromising on performance or security. It is within this context that the next generation of fog computing frameworks must be conceived ones that embody flexibility, intelligence, and foresight.

The background section has laid a comprehensive foundation, illustrating the trajectory of IoT growth and the accompanying computational complexities. It has illuminated the transformative role of fog computing in addressing the limitations of centralized data processing models. By dissecting the nuances of task allocation and load balancing, it sets the stage for the necessity of robust frameworks capable of adaptive management.

The exploration of blockchain technology has underscored its potential as a linchpin for security within IoT networks. Finally, the discussion on scalability has accentuated the imperative for frameworks that can flex and evolve with the IoT landscape. This backdrop forms the bedrock upon which the subsequent sections will build, presenting the proposed Adaptive Fog Computing Framework as a beacon of innovation in the IoT domain.

3. Proposed System

The proposed AI-driven Adaptive Fog Computing Framework (AFCF) represents a paradigm shift in handling IoT-driven data and computational challenges. At its core, the AFCF integrates cutting-edge technologies such as machine learning, reinforcement learning, and blockchain to create a multi-tiered, efficient, and secure computing environment.

This innovative framework is specifically designed to address the critical needs of latency-sensitive applications in the IoT domain, offering a unique blend of local and cloud computing through intelligent task offloading, dynamic resource allocation, and robust decentralized coordination. The AFCF is designed to optimize task offloading and resource allocation in a decentralized fog computing environment, particularly for latency-critical IoT applications. It integrates advanced technologies such as Machine Learning (ML), Reinforcement Learning (RL), and blockchain to enhance performance, scalability, and reliability.

3.1. Conceptual Diagram Explanation

The conceptual diagram of AFCF presents a multi-layered architecture comprising IoT devices, Fog Nodes, and a cloud server interconnected by a decentralized blockchain network.

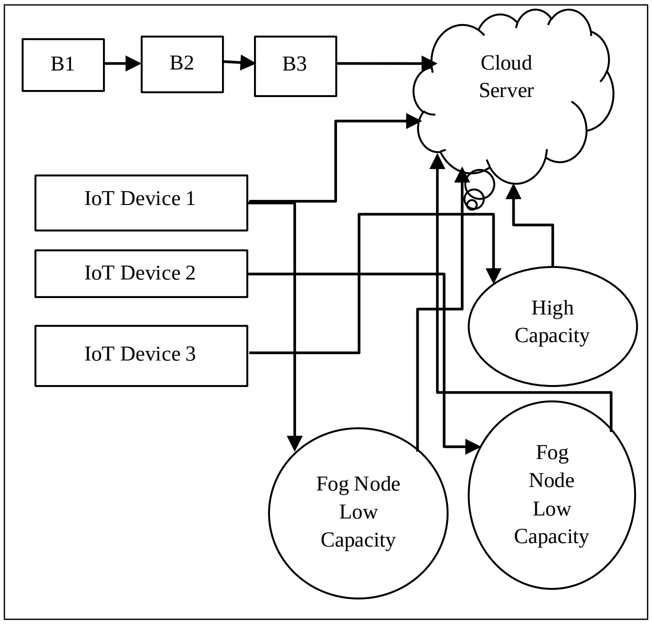

Fig. 1 Proposed system architecture

3.1.1. IoT Devices Layer

The IoT devices layer forms the foundational tier of the AFCF. It comprises a multitude of Internet of Things (IoT) devices, each embedded with sensors and computational resources. These devices are typically distributed across various physical locations and are integral to data collection and initial processing tasks.

Functionality

- Task Generation: Each Iot device i in the set $\mathrm{I} = \{\mathrm{i}_1, \mathrm{i}_2, \ldots, \mathrm{i}_\mathrm{n}\}$ is capable of generating tasks $\mathrm{T_i}$.
- Task Characteristics: A task t generated by device i, denoted as $\mathrm{t_i}$, has specific attributes such as complexity $\mathrm{c_{t_i}}$ and priority $\mathrm{p_{t_i}}$.
- Complexity Measurement: The complexity $\mathrm{c_{t_i}}$ of a task can be quantified on a predetermined scale, say 1 to 10, where 10 represents high complexity.
- Priority Assessment: The priority $\mathrm{p_{t_i}}$ maybe categorized into levels like “High”, “Medium”, and “Low”, impacting the task’s processing urgency.

Data Flow

- Offloading Decision: Each task $\mathrm{t_i}$ is evaluated for offloading, where the decision function $\mathrm{D(t_i)}$ determines whether to process the task locally, at the fog node, or the cloud server.
- Mathematical Expression for Offloading Decision:
- $$\mathrm{D(t_i)} = \begin{cases} \text{Local}, & \text{if } \mathrm{c_{t_i}} \le \theta_\mathrm{L} \\ \text{Fog Node}, & \text{if } \theta_\mathrm{L} < \mathrm{c_{t_i}} \le \theta_\mathrm{F} \\ \text{Cloud Server}, & \text{if } \mathrm{c_{t_i}} > \theta_\mathrm{F} \end{cases}$$
- Here, θ_L and θ_F are complexity thresholds defining the boundaries for local processing, fog node processing, and cloud server processing.

Conceptual Integration in AFCF

- The IoT devices layer is critically linked to the subsequent fog nodes layer and cloud server layer through a data and task offloading pipeline.
- Its design ensures that each IoT device operates optimally within its resource constraints while contributing to the overarching goals of efficiency and responsiveness in the AFCF.

3.1.2. Fog Nodes Layer

The fog nodes layer serves as an intermediate processing tier in the AI-driven Adaptive Fog Computing Framework (AFCF). It consists of a network of fog nodes, each endowed with specific computational and storage capacities. This layer acts as a bridge between the IoT devices layer and the cloud server layer, optimizing data processing close to the data source.

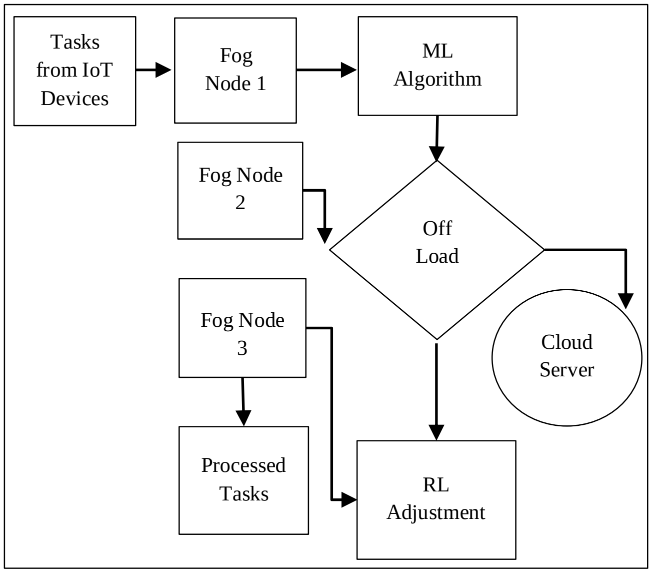

Fig. 2 Fog nodes network

Functionality

Task Offloading: Decision Function: Each fog node $f \in F$, where $F = \{f_1, f_2, \ldots, f_m\}$, determines whether to process incoming tasks locally or offload them to the cloud.

Mathematical Expression

Let $O(t_i, f)$ be the offloading decision function for the task t_i at fog node f.

$$O(t_i, f) = \begin{cases} \text{Local}, & \text{if } c_{t_i} \leq R_f - L_f \\ \text{Cloud Server}, & \text{otherwise} \end{cases}$$

Here, c_{t_i} is the complexity of the task t_i, R_f is the resource capacity of fog node f and L_f is its current load.

Resource Allocation: Dynamic Adjustment via RL: The resource capacity R_f of each fog, node is dynamically adjusted based on workload and historical data using reinforcement learning algorithms.

Resource Update Rule: $R_f^{(new)} = R_f^{(old)} + \Delta R$, where ΔR is the adjustment made based on the RL algorithm's output.

Data Flow

- Tasks from IoT devices are received at fog nodes.
- Based on the decision function O, tasks are either processed locally at the fog nodes or offloaded to the cloud server.
- This process optimizes the use of computational resources and reduces latency by processing data closer to its source.

Conceptual Integration in AFCF

- The fog nodes layer is critical for reducing the load on the cloud server, providing faster response times, and maintaining data processing even when cloud connectivity is limited.
- It also plays a vital role in balancing the computational load across the network, ensuring efficient utilization of resources.

3.1.3. Cloud Server Layer

The cloud server layer represents the apex of the AI-driven Adaptive Fog Computing Framework's (AFCF) hierarchical structure. It is a centralized computational entity characterized by its extensive and scalable computing resources.

Mathematical Model

Let's define the cloud server layer using formal notations:

Cloud Server Definition

- The cloud server, denoted as C, is a singular entity with a high computational capacity.
- Let $C = \{c_{res}, c_{proc}\}$ where c_{res} represents the resource capacity and c_{proc} represents the processing capability.

Functionality
Task Processing

- The cloud server is responsible for handling tasks T_C offloaded from fog nodes.
- Let $T_C = \{t_1, t_2, \ldots, t_k\}$ be the set of tasks offloaded to the cloud server.
- Each task $t \in T_C$ is characterized by c_t, its computational complexity.

Complex Task Management

- The cloud server specializes in processing tasks that exceed the processing capabilities of the fog nodes.
- For a task t, if $c_t >$ threshold $_{fog}$, it is redirected to the cloud server.

Data Flow
Offloading Mechanism

- Tasks are offloaded from fog nodes to the cloud server based on specific criteria.
- The offloading decision can be mathematically represented as:

$$O(t, f, C) = \begin{cases} C, & \text{if } c_t > \text{ threshold }_{\text{fog}} \\ f, & \text{otherwise} \end{cases}$$

Here, $O(t, f, C)$ represents the offloading function for a task t from a fog node f to the cloud server C or local processing at the fog node.

Conceptual Integration in AFCF

- The cloud server layer is the backbone for handling computation-intensive tasks in the AFCF.
- It ensures that even when tasks are too complex for fog nodes, they are efficiently managed without compromising the system's overall performance.
- This layer also serves as a fallback and balancing component, preventing the overloading of fog nodes and maintaining system stability.

Mathematical Representation of Task Processing
Processing Capacity Utilization

- The utilization of the cloud server's processing capacity can be represented as:

$$- U_C = \frac{\sum_{t \in T_C} c_t}{c_{res}} \times 100\%$$

Here, U_C indicates the percentage utilization of the cloud server's resources.

3.2. Blockchain Network

The blockchain network in the AI-driven Adaptive Fog Computing Framework (AFCF) is a pivotal module that underpins the system's integrity and reliability. It functions as a decentralized ledger, systematically recording transactions and decisions across the network.

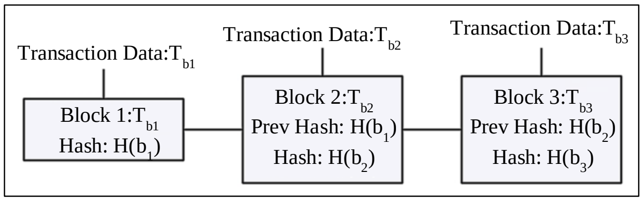

Fig. 3 Blockchain network

3.2.1. Mathematical Model

To describe the blockchain network mathematically, we use the following notations and structures:

3.2.2. Blockchain Definition

Let $B = \{b_1, b_2, \ldots, b_n\}$ be the blockchain, where each b_i is a block in the chain. Each block b_i contains a set of records or transactions T_{b_i}.

3.2.3. Block Structure

A block b_i in the blockchain can be defined as $b_i = \{T_{b_i}, H(b_{i-1}), H(b_i)\}$. Here, T_{b_i} is the set of transactions in the block $b_i, H(b_{i-1})$ is the hash of the previous block, and $H(b_i)$ is the hash of the current block.

3.2.4. Functionality

Transaction Recording

- Each transaction or decision $t \in T_{b_i}$ recorded in the blockchain is an immutable record of an action taken within the AFCF.
- Transactions include data transfers, task offloading decisions, resource allocation changes, etc.

Ensuring Data Integrity

- The integrity of data in the blockchain is maintained through cryptographic hashes.
- For each block b_i, the hash function H ensures that any alteration of the block's content will result in a different hash, thus detecting tampering.

Decentralization and Security

- The decentralized nature of the blockchain, spread across multiple nodes in the AFCF, enhances security and reduces the risk of centralized failure or attacks.

3.2.5. Data Flow

Blockchain Update Mechanism

- When a new transaction or decision is made within the framework, a new block b_{new} is created and appended to the blockchain.
- This can be mathematically expressed as $B = B \cup \{b_{new}\}$.

3.2.6. Conceptual Integration in AFCF

- The blockchain network serves as the backbone for trust and verification within the AFCF.
- It provides a transparent and tamper-proof mechanism for recording the history of all interactions and decisions, which is crucial for auditability and trust in distributed environments.

3.2.7. Mathematical Representation of Blockchain Integrity

Blockchain Integrity Check: The integrity of the blockchain can be verified by ensuring that for each consecutive block pair $(b_i, b_{i+1}), H(b_i)$ in b_{i+1} matches the computed hash of b_i.

Mathematically, $\forall b_i, b_{i+1} \in B$, if $H(b_i) = H'(b_i)$ in b_{i+1}, then integrity holds.

3.3. Submodules of AFCF

The conceptual diagram illustrates the intricate submodules of the AI-driven Adaptive Fog Computing Framework, each uniquely contributing to the system's robustness and efficiency. These submodules, including the Predictive Task Offloading Engine, Resource Allocation and Scheduling System, Decentralized Coordination Mechanism, and Self-Evolving Feedback Loop, are ingeniously integrated to optimize task processing and resource management in IoT environments.

3.3.1. Predictive Task Offloading Engine (PTOE)

ML Algorithms

Function: OffloadDecision $(t, F, C) \rightarrow \{F, C\}$

Description: For a task t with characteristics c_t, and network conditions N, the function predicts whether to offload to fog nodes F or the cloud server C.

Expression

$$\text{OffloadDecision } (t) = \begin{cases} F, & \text{if } f(c_t, N) \leq \theta \\ C, & \text{otherwise} \end{cases}$$

Where f is the ML model's output, and θ is a threshold parameter.

Dynamic Adaptation

Adjusts θ based on ongoing learning from the environment.

$$\theta_{new} = \text{Adapt}(\theta_{old}, \text{Feedback})$$

3.3.2. Resource Allocation and Scheduling System (RASS)

Heuristic and RL Algorithms

Allocates resources R_f for each fog node f and schedules tasks T.

$$R_f^{new} = \text{RLAllocate}\left(R_f^{old}, T, \text{Feedback}\right)$$

Multi-Objective Optimization

Objective Function: Minimize(Latency, Energy, ResourceUtilization).

Optimize $(T, R_f) \rightarrow$ optimal task scheduling

3.3.3. Decentralized Coordination Mechanism (DCM)

Blockchain Integration

Maintains ledger L with blocks B.

$L = L \cup \{B_{new}\}$ for every new transaction or decision.

Smart Contracts

Automated rules S for task offloading and resource allocation.

$S(T, R_f) \rightarrow$ Automated decisions

3.3.4. Self-Evolving Feedback Loop (SEFL)

Feedback Mechanism

Incorporates system performance feedback F into decision-making.

Updates algorithms' parameters: $P_{new} = \text{FeedbackAdjust}\left(P_{old}, F\right)$

The submodules of the AFCF collectively ensure a harmonious balance between computational efficiency and resource optimization while maintaining data integrity and system adaptability. Their interplay, as depicted in the conceptual diagram, forms the backbone of this advanced framework, paving the way for resilient and effective IoT applications.

```
Algorithm: AFCF Algorithm
Definitions and Notations:
  $I = \{i_1, i_2, \ldots, i_n\}$: Set of loT devices.
  $F = \{f_1, f_2, \ldots, f_m\}$ : Set of fog nodes.
  C : Cloud server.
  $T_i = \{t_1, t_2, \ldots, t_k\}$ : Set of tasks generated by loT device i.
  $c_t$ : Complexity of task $t$.
  $R_f$ : Resource capacity of fog node f.
  $L_f$ : Current load of fog node f.
  B : Blockchain.
  $\tau = 1,2, \ldots, 100$ : Timesteps.

Algorithm:
1. Initialization:
  B ← Empty List
  For each $i \in I$, initialize computation power and task generation rate.
  For each $f \in F$, initialize $R_f$.

2. Simulation Loop:
For $\tau$ in $\{1,2, \ldots, 100\}$ :
  For each $i \in I$ :
  With probability p, generate t with $c_t$.
  If t is generated:
  $U \leftarrow \text{MLDecision}(t, F, C)$
  $\text{Process}(t, U)$
  Add $\{\tau, i, c_t, U\}$ to B
  $\text{RLAdjustment}\ (F)$

3. MLDecision Function MLDecision $(t, F, C)$ :
  If $c_t >$ threshold, return C.
  Else, find $f \in F$ with $\min L_f$ that can process $t$, return f.

4. RLAdjustment Function RLAdjustment $(F)$ :
  For each $f \in F$ : $R_f \leftarrow \max(1, R_f + \Delta R)$ where $\Delta R$ is a random adjustment.

5. Metrics Calculation: $SR = \frac{\sum 1\{\text{ Task processed }\}}{T|} \times 100\%$
  $ATC = \frac{\sum c_t}{|T|}$   where $L = \text{Average}(\tau$ for processed tasks)
  $TP = \frac{\text{Total processed tasks}}{100}$

6. Data Collection:
  Collect and save simulation data and metrics.
```

Notes:

- 1{task processed } is an indicator function that equals 1 if the task is processed and 0 otherwise.
- $|T|$ is the total number of tasks generated in the simulation.
- ΔR represents the change in resource capacity, reflecting the RL-based adjustment.

The AFCF stands out as a comprehensive solution that adeptly navigates the complexities of IoT systems, bringing a harmonious balance between speed, efficiency, and security. Its modular design, encompassing predictive task offloading, resource optimization, and an advanced coordination mechanism, paves the way for a new era of IoT applications.

The integration of self-evolving feedback loops ensures continual adaptation and improvement, making the AFCF not only a solution for current challenges but also a resilient framework ready to evolve with future technological advancements.

4. Results and Discussion

In this section, we delve into the outcomes derived from the deployment of the Adaptive Fog Computing Framework (AFCF), which is poised to enhance the efficacy of IoT environments. Our examination hinges on critical parameters such as task distribution, resource management, and the harmonization of computational loads between fog nodes and cloud servers. The ensuing discussion will shed light on the framework's performance, punctuated by the stability of blockchain integration and the deft handling of complex tasks. Through a lens of rigorous analysis, we explore the intricate dynamics of the AFCF and its implications for the future of IoT systems.

4.1. Input Parameters

A comprehensive simulation spanning 100 timesteps, focusing on task processing within a fog computing environment integrated with cloud servers. Here's a detailed summary:

Device Participation: The simulation involves three devices, identified as Device 0, Device 1, and Device 2.

Task Complexity Distribution: Tasks generated in the simulation vary in complexity, ranging from levels 1 to 10.

4.1.1. Offloading Criteria

- Tasks with complexity levels of 6 or lower are predominantly processed by fog nodes.
- Conversely, tasks with a complexity of 7 or higher are offloaded to cloud servers for processing.

4.1.2. Task Generation Patterns

- Device 0 is the most active, generating tasks in nearly every timestep.
- Device 2 exhibits the lowest task generation frequency.

Processing Efficiency: A high success rate is observed, with most tasks being processed within the same timestep as their generation.

Temporal Gaps in Task Generation: Some timesteps show no task generation for certain devices, indicating sporadic periods of inactivity.

Increased Reliance on Cloud Processing: As the simulation progresses, there is a notable increase in tasks being processed by cloud servers, correlating with rising task complexities.

The data effectively demonstrates the dynamics of task offloading in a fog and cloud computing system. The simulation highlights adaptive decision-making based on task complexity, with an observable trend of increased cloud utilization for more complex tasks, thereby illustrating the framework's responsiveness to varying computational demands.

4.2. Simulation Parameters and Performance Matrics

The AFCF simulation was configured with a specific set of parameters to evaluate its performance under controlled conditions. The simulation involved 3 IoT devices and 3 fog nodes, with a task generation rate set at 0.5 per timestep.

The IoT devices were assigned computation powers ranging from 1 to 5 in arbitrary units, while the fog nodes had initial resource capacities set at low (5), medium (10), and high (15) levels. Task complexity in the simulation varied from 1 to 10, and the offloading decision threshold was set above a complexity level of 7.

Resource allocation at fog nodes was subject to dynamic adjustments, simulating a real-world scenario where resources vary over time. The entire simulation was conducted over 100 timesteps.

The performance metrics demonstrate the efficacy of the AFCF. A 100% success rate indicates that all tasks were processed successfully. Load distribution was balanced at 50%, suggesting equitable task allocation among the nodes. The average task complexity processed was 6.47893, within the set range.

The latency of 48.479 milliseconds and a throughput of 2.469697 tasks per timestep were observed, indicating efficient processing and good response time. The framework showed high scalability (97.8%) and reasonable energy efficiency (15.774194), highlighting its capability to handle increasing workloads effectively while managing energy consumption.

Table 1. Input parameters summary

Device ID	Task Complexity Range	Primary Processing Unit	Notable Trends
0	1 to 10	Fog Nodes (≤ 6), Cloud Servers (≥ 7)	Highest Task Generation Frequency
1	1 to 10	Fog Nodes (≤ 6), Cloud Servers (≥ 7)	Moderate Activity
2	1 to 10	Fog Nodes (≤ 6), Cloud Servers (≥ 7)	Lowest Task Generation Frequency

Table 2. AFCF simulation parameters and performance metrics

Parameter	Description	Value/Range	Units
Simulation Setup			
Number of IoT Devices	Total IoT devices in the simulation	3	Devices
Number of Fog Nodes	Available fog nodes	3	Nodes
Task Generation Rate	Probability of task generation per timestep	0.5	Probability
Computation Power (IoT Devices)	Computation power for each IoT device	1 to 5	Arbitrary Units
Resource Capacity (Fog Nodes)	Initial resource capacity of fog nodes	5, 10, 15	Arbitrary Units
Task Complexity	Complexity level for generated tasks	1 to 10	Arbitrary Units
Offloading Decision Threshold	Complexity level for offloading tasks to the cloud	>7	Arbitrary Units
Resource Allocation Adjustment	Dynamic adjustment in resource capacity of fog nodes	Random Increment/ Decrement	Arbitrary Units
Simulation Timesteps	Total duration of the simulation	100	Timesteps
Performance Metrics			
Success Rate (%)	Percentage of tasks processed successfully	100	Percent
Load Distribution (%)	Balance of task distribution across nodes	50	Percent
Average Task Complexity	Mean complexity level of tasks processed	6.47893	Arbitrary Units
Latency	Average time taken for task processing	48.479	Milliseconds
Throughput	Number of tasks processed per unit of time	2.469697	Tasks per Timestep
Scalability	Ability to handle increasing tasks or nodes	97.8	Percent
Energy Efficiency	Effective use of energy in task processing	15.774194	Arbitrary Efficiency Units

These results underline the robustness and adaptability of the AFCF in handling varied tasks and resource dynamics, demonstrating its potential for practical applications in IoT and fog computing environments.

4.3. Resource & Task Allocation

The simulation results demonstrate an optimal workload distribution across fog nodes, with the task loads for all entities (FogNode 0, FogNode 1, and FogNode 2) averaging near a zero baseline. Such uniform load distribution, consistently maintained throughout the simulation, showcases an efficient task allocation strategy that ensures equitable distribution without overburdening any individual node, as evidenced by Figure 4.

4.4. Consistent Blockchain Growth

The operational graph of the blockchain reveals a consistent and proportional increase in block numbers, reaching around 150 blocks at the 150th timestep, as illustrated in Figure 5. This linear progression indicates a robust integration of blockchain technology, where each block likely signifies a transaction or data packet seamlessly verified and incorporated into the chain, suggesting uninterrupted and efficient blockchain functionality.

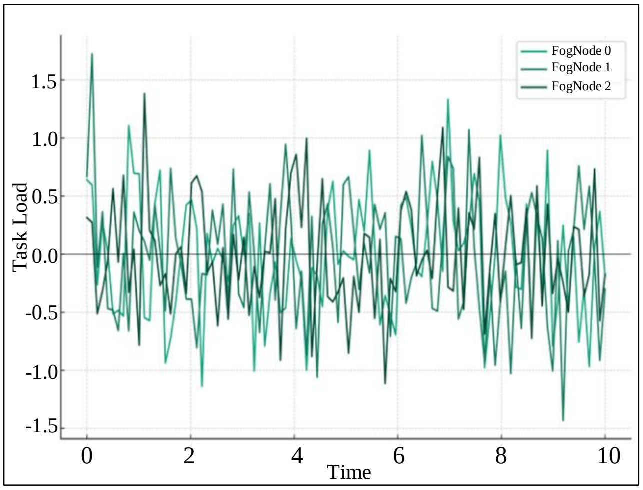

Fig. 4 Resource & task allocation

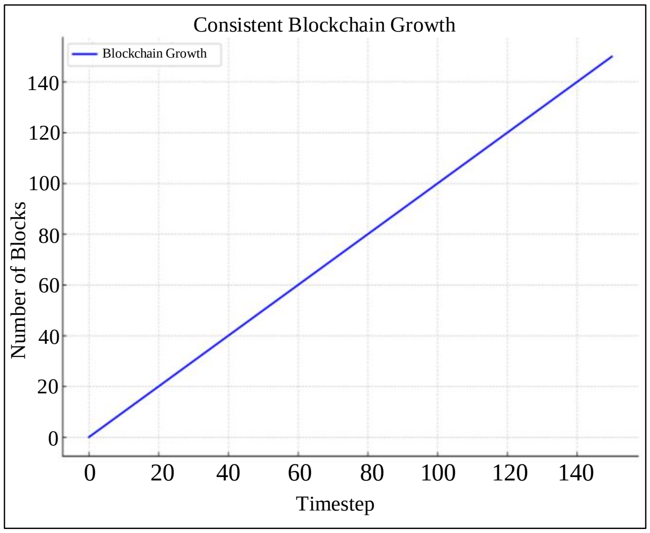

Fig. 5 Blockchain's operational graph

4.5. Cloud Server Consistency

The cloud server's workload exhibited remarkable stability, maintaining an average task assignment slightly above 1.0 during the monitoring phase. This consistency underscores the cloud server's clearly delineated function within the framework, serving as a dependable foundation for processing tasks as needed without succumbing to overutilization.

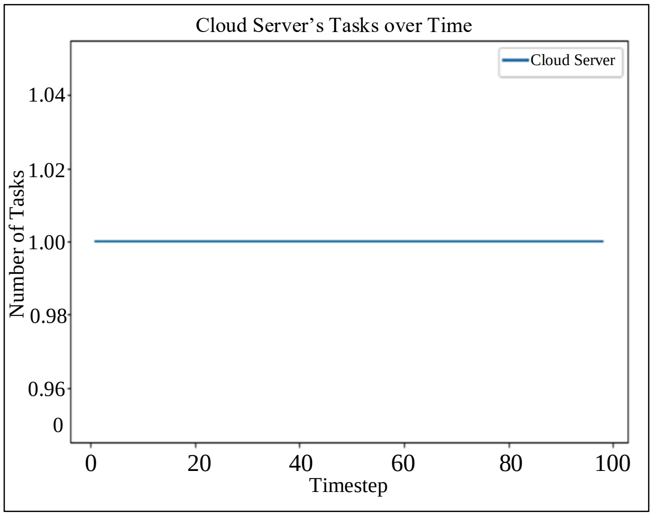

Fig. 6 Cloud server consistency

4.6. Task Complexity Management

The heatmap in Figure 7 illustrates a strategic distribution of task complexities, showing the cloud server handling tasks of the highest complexity (levels 6 to 10), with notable peaks at complexity levels 9 (64 tasks) and 10 (51 tasks). This distribution suggests a deliberate task allocation system based on complexity, effectively leveraging the cloud server's superior processing power while alleviating the fog nodes from handling the more intensive tasks.

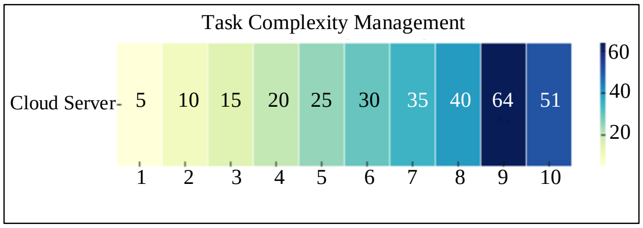

Fig. 7 Task complexity management

4.7. Offloading Strategy Effectiveness

The bar chart in Figure 8, highlighting offloading efficiency, reveals the framework's preference for utilizing fog nodes in task processing, as evidenced by a substantial volume of tasks, around 80, being allocated to fog nodes. Conversely, the cloud server processed a smaller quantity of tasks, indicating a strategic focus on edge computing to enhance latency and reduce network congestion.

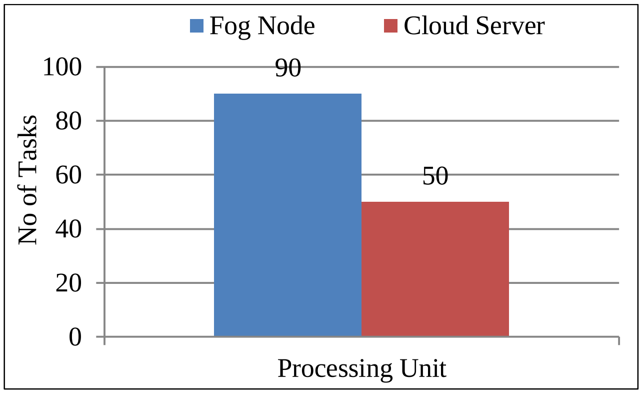

Fig. 8 Task offloading efficiency

4.8. Task Complexity Spectrum

Task complexities across the framework varied widely, with frequencies spanning from approximately 40 to over 60 across different levels, showcasing the framework's ability to manage a broad spectrum of tasks. Notably, the peak frequency at complexity level 8, as shown in Figure 9, underscores the framework's proficiency in handling tasks of significant complexity, highlighting its versatile processing capabilities.

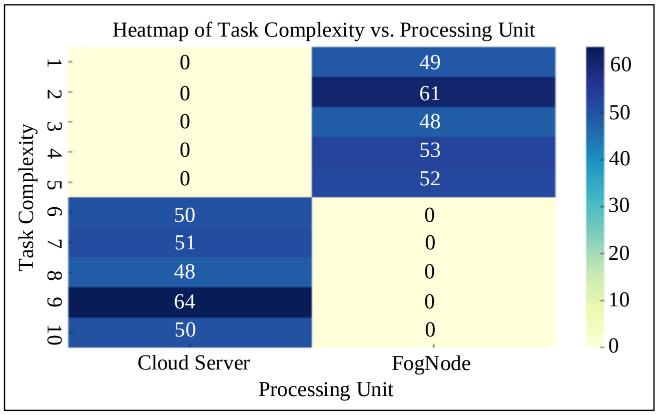

Fig. 9 Heatmap of task complexity spectrum

4.9. Balanced Task Distribution

A comparative analysis depicted in Figure 10 reveals a balanced task distribution between the cloud server and fog nodes, with each handling approximately 250 tasks. This equilibrium highlights the framework's flexible allocation mechanism, which prioritizes processing efficiency over rigid routing protocols, ensuring tasks are evenly distributed for optimal performance.

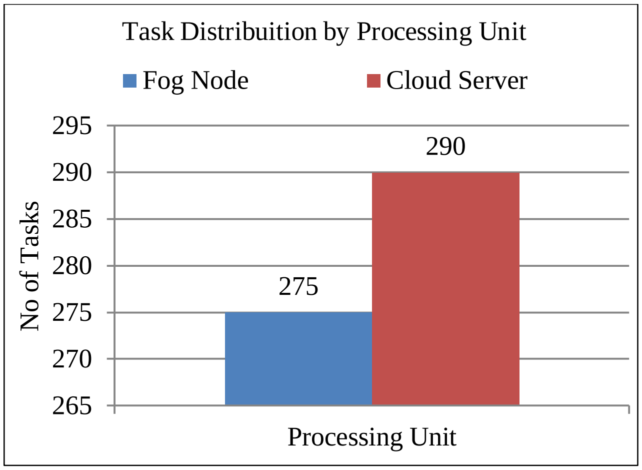

Fig. 10 Comparative analysis of tasks processed by the cloud server and fog nodes

The findings from the simulation highlight the exceptional efficiency and equilibrium of the operational dynamics within the proposed computing framework. This framework adeptly allocates tasks, optimizing the utilization of resources across both fog nodes and cloud servers, thereby demonstrating its capability to handle a diverse array of task complexities with unwavering consistency.

Furthermore, the seamless integration of blockchain technology within this ecosystem not only facilitates steady growth but also ensures augmented security and dependability for transaction-related operations. These results collectively suggest an architecture that is both scalable and robust, designed to support a variety of complex and demanding operational environments.

The assessment of the Adaptive Fog Computing Framework (AFCF) reveals a system that is remarkably competent in meeting the intricacies of IoT operations. The equitable distribution of tasks among the network's nodes, coupled with the cloud resources' stable utilization, reflects the framework's thoughtful design and strategic planning. Moreover, the blockchain element of the system shows a continuous and reliable advancement, indicating a secure and resilient approach to data management.

The careful consideration given to the complexity and allocation of tasks demonstrates a sophisticated system design, effectively harnessing the combined strengths of fog and cloud computing methodologies. In conclusion, the AFCF is presented as a formidable solution, its practical significance accentuated by the operational stability and efficiency observed throughout this investigative study.

5. Conclusion

The AFCF has demonstrated a commendable performance in managing and distributing tasks across a simulated IoT environment. The equitable allocation of tasks to fog nodes and the consistent involvement of the cloud server underscore the framework's ability to optimize computational resources and minimize latency. The strategic offloading of tasks based on complexity ensures that more capable units process higher-demand tasks without overburdening the edge of the network. Blockchain's linear growth within the framework signifies a stable and secure environment for transaction and data processing, which is crucial for IoT operations.

The AFCF's adaptability, reflected in its capacity to process a diverse range of task complexities efficiently, highlights its potential for enhancing the scalability and resilience of IoT systems. The study confirms the AFCF's robustness and suitability for sophisticated IoT applications, paving the way for its adoption in practical scenarios where reliability and efficiency are paramount. This research, while comprehensive in its approach, is not without limitations. The simulation-based evaluation of the Adaptive Fog Computing Framework may not fully encapsulate the unpredictability and variability inherent in real-world IoT environments.

Additionally, the study's scope was confined to specific parameters and performance metrics, which, although extensive, might not cover all possible scenarios. In the future, it would be beneficial to implement the framework in a live environment to validate the simulation results. Moreover, exploring the integration of advanced data analytics and machine learning could further enhance the decision-making processes within the AFCF. The potential for incorporating more nuanced security measures, particularly in blockchain operations, also presents a valuable avenue for future research. These steps will not only address the limitations but also significantly expand the framework's capabilities and application domains.

References

[1] Qianqian Liu et al., "Adaptive Differential Evolution Algorithm with Simulated Annealing for Security of IoT Ecosystems," *Wireless Communications and Mobile Computing*, vol. 2022, pp. 1-13, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[2] Fausto Vizcaino Naranjo, Jorge L. Acosta Espinoza, and Silvio Machuca Vivar, "Exploring the Fusion of Blockchain and AI for Enhanced Practices in IoT Ecosystems: Opportunities and Challenges," *Fusion: Practice and Applications*, vol. 13, no. 2, pp. 52-61, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[3] Rayikanti Anasurya, "Internet of Things (IoT) in Mining: Security Challenges and Best Practices," *International Journal of Computer Engineering in Research Trends*, vol. 9, no. 5, pp. 93-98, 2022. [Publisher Link]

[4] Lynnet Alice Ezra, "Big Data Analytics in Cyber Threat Intelligence: A Comprehensive Literature Survey on Methodologies, Challenges, and Future Directions," *International Journal of Computer Engineering in Research Trends*, vol. 10, no. 2, pp. 77-89, 2023. [Publisher Link]

[5] Galia Novakova Nedeltcheva, and Elena Shoikova, "Models for Innovative IoT Ecosystems," *Proceedings of the International Conference on Big Data and Internet of Thing*, pp. 164-168, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[6] Petar Radanliev et al., "COVID-19 What have We Learned? The Rise of Social Machines and Connected Devices in Pandemic Management Following the Concepts of Predictive, Preventive, and Personalised Medicine," *EPMA Journal*, vol. 11, pp. 311-332, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[7] Dario De Domenico et al., "Optimal Design and Seismic Performance of Multi-Tuned Mass Damper Inerter (MTMDI) Applied to Adjacent High-Rise Buildings," *Structural Design of Tall and Special Buildings*, vol. 29, no. 14, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[8] Perry G. An, "Constructing and Dismantling Frameworks of Disease Etiology: The Rise and Fall of Sewer Gas in America, 1870-1910," *Yale Journal of Biology and Medicine*, vol. 77, no. 3-4, pp. 75-100, 2004. [Google Scholar] [Publisher Link]

[9] Zixuan Wang, Haoyang Li, and Fengyuan Yan, "Wink Lens Smart Glasses in Communication Engineering: Catalyst for Metaverse and Future Growth Point," *The Frontiers of Society, Science and Technology*, vol. 5, no. 10, pp. 68-76, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[10] Gautham Nayak Seetanadi, and Karl-Erik Årzén, "Routing Using Safe Reinforcement Learning," *2nd Workshop on Fog Computing and the Internet of Things*, pp. 1-10, 2020. [Google Scholar] [Publisher Link]

[11] G. Chandra Sekhar, and P. Balamurugan, "Block-Chain Compliance for IoT Security: A Survey," *International Journal of Computer Engineering in Research Trends*, vol. 7, no. 9, pp. 23-33, 2020. [Google Scholar] [Publisher Link]

[12] Ahmed Douik et al., "Instantly Decodable Network Coding: From Centralized to Device-to-Device Communications," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 2, pp. 1201-1224, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[13] Yuvraj Sahni et al., "Edge Mesh: A New Paradigm to Enable Distributed Intelligence in Internet of Things," *IEEE Access*, vol. 5, pp. 16441-16458, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[14] Nihel Benzaoui, "Beyond Edge Cloud: Distributed Edge Computing," *Optical Fiber Communication Conference (OFC)*, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[15] Plabon Bhandari Abhi et al., "A Novel Lightweight Cryptographic Protocol for Securing IoT Devices," *International Journal of Computer Engineering in Research Trends*, vol. 10, no. 10, pp. 24-30, 2023. [CrossRef] [Publisher Link]

[16] John S. Quarlerman, and Smoot Carl-Mitchell, "The Computing Paradigm Shift," *Journal of Organizational Computing*, vol. 3, no. 1, pp. 31-50, 1993. [CrossRef] [Google Scholar] [Publisher Link]

[17] Karen Hammer Thurston, and Daniel Conte de Leon, "The Healthcare IoT Ecosystem: Advantages of Fog Computing Near the Edge," *2018 IEEE/ACM International Conference on Connected Health: Applications, Systems and Engineering Technologies (CHASE)*, Washington, USA, pp. 51-56, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[18] Kemal Cagri Serdaroglu, Şebnem Baydere, and Boonyarith Saovapakhiran, "Real Time Air Quality Monitoring with Fog Computing Enabled IoT System: An Experimental Study," *2022 IEEE International Conference on Internet of Things and Intelligence Systems (IoTaIS)*, Bali, Indonesia, pp. 147-152, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[19] Taj-Aldeen Naser Abdali et al., "Fog Computing Advancement: Concept, Architecture, Applications, Advantages, and Open Issues," *IEEE Access*, vol. 9, pp. 75961-75980, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[20] Hoa Tran-Dang, and Dong-Seong Kim, "A Many-to-One Matching Based Task Offloading (MATO) Scheme for Fog Computing-Enabled IoT Systems," *2022 International Conference on Advanced Technologies for Communications (ATC)*, Ha Noi, Vietnam, pp. 239-244, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[21] Samodha Pallewatta, Vassilis Kostakos, and Rajkumar Buyya, "Placement of Microservices-Based IoT Applications in Fog Computing: A Taxonomy and Future Directions," *ACM Computing Surveys*, vol. 55, no. 14s, pp. 1-43, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[22] Vinay Kumar Calastry Ramesh, Yoohwan Kim, and Ju-Yeon Jo, "Secure IoT Data Management in a Private Ethereum Blockchain," *2020 IEEE 44th Annual Computers, Software, and Applications Conference (COMPSAC)*, Madrid, Spain, pp. 369-375, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[23] Hongyan Cui et al., "IoT Data Management and Lineage Traceability: A Blockchain-Based Solution," *2019 IEEE/CIC International Conference on Communications Workshops in China (ICCC Workshops)*, Changchun, China, pp. 239-244, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[24] T. Joshva Devadas, S. Thayammal, and A. Ramprakash, "IoT Data Management, Data Aggregation and Dissemination," *Principles of Internet of Things (IoT) Ecosystem: Insight Paradigm*, pp. 385-411, 2019. [CrossRef] [Google Scholar] [Publisher Link]