applied sciences

MDPI

Article

On-Chain/Off-Chain Adaptive Low-Latency Network Communication Technology with High Security and Regulatory Compliance

Yu Jin †, Daming Huang † and Chen Tian *

State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093, China; yjin@smail.nju.edu.com (Y.J.); huangdm@nju.edu.cn (D.H.)
* Correspondence: tianchen@nju.edu.cn
† These authors contributed equally to this work.

Abstract

The rapid advancement of blockchain technology has introduced a new paradigm for constructing trusted digital economic infrastructure. However, its large-scale adoption remains constrained by dual challenges: on-chain and off-chain communication efficiency and security assurance. This paper addresses the universal demands of blockchain in complex application scenarios by proposing a low-latency, high-security, adaptive, and regulatory-compliant network communication technology bridging on-chain and off-chain systems. A hierarchical "device–edge–chain" communication architecture based on edge gateways is designed to address the critical challenge of achieving one-second on-chain processing for tens of millions of data entries. Experimental validation demonstrates that the system sustains transaction throughput at the scale of at least 10 million while consistently maintaining sub-second latency thresholds. Furthermore, implemented fault tolerance mechanisms ensure reliable operation through dynamic path switching and capacity-aware load redistribution. This architecture systematically resolves the performance–security–regulatory compliance trilemma inherent in conventional blockchain systems deployed within complex real-world environments.

Keywords: blockchain; network; SDN; on-chain/off-chain communication

Academic Editor: Juan-Carlos Cano

Received: 22 May 2025
Revised: 22 July 2025
Accepted: 30 July 2025
Published: 12 August 2025

Citation: Jin, Y.; Huang, D.; Tian, C. On-Chain/Off-Chain Adaptive Low-Latency Network Communication Technology with High Security and Regulatory Compliance. *Appl. Sci.* **2025**, *15*, 8880. https://doi.org/10.3390/app15168880

1. Introduction

Blockchain technology has been widely adopted across diverse domains including finance, supply chains, healthcare, and intellectual property due to its decentralized and tamper-resistant characteristics. However, its performance limitations significantly hinder large-scale practical deployment.

Among the core metrics for evaluating blockchain system performance, throughput and latency are particularly critical. Throughput, measured in transactions per second (TPS), quantifies the number of transactions a system can process within a unit time frame, while latency refers to the time required for transaction submission to network confirmation. These metrics directly determine a blockchain system's practicality and scalability—high throughput enables support for large-scale commercial applications, and low latency is essential for time-sensitive scenarios such as high-frequency trading. Current mainstream public chains exhibit throughput magnitudes lower than centralized systems (e.g., Bitcoin's [1] 7 TPS vs. Visa's 24,000 TPS), with transaction confirmation latencies often spanning minutes or hours, constituting the primary barrier to blockchain adoption.

Efforts to enhance throughput and reduce latency have yielded mixed results. For public chains, off-chain scaling solutions (e.g., Lightning Network [2]) alleviate mainchain congestion by diverting high-frequency transactions to external channels, while sharding techniques partition networks into parallel subchains (shards) to improve throughput [3–6]. Consortium chains, leveraging geographically compact node deployments [7], achieve higher throughput and lower latency. However, public chains inherently face lower upper bounds in performance due to node distribution and trust constraints, with true high-performance blockchains predominantly emerging from consortium chains that sacrifice partial decentralization. Even with coordinated optimizations in hardware–software integration and consensus algorithms [8–10], consortium chains struggle to surpass 100,000 TPS thresholds, rendering them inadequate for high-frequency trading or large-scale infrastructure scenarios.

Current research on blockchain on-chain/off-chain data interaction remains nascent. No existing solution adequately addresses the comprehensive requirements of large-scale heterogeneous terminals, sub-second latency (1 s on-chain processing), high throughput (tens of millions of transactions), adaptive communication, and regulatory-compliant workflows. To bridge these gaps—stemming from inherent blockchain limitations, inadequate on-chain/off-chain interaction schemes, and unmet complex demands—this study pioneers a holistic on-chain/off-chain network communication framework. The proposed solution systematically addresses these challenges through a device–edge–chain hierarchical architecture anchored by software-defined edge gateways, integrating high-efficiency data processing, intelligent network management, and embedded regulatory mechanisms.

Beyond technical efficiency, this work also emphasizes compliance with governmental regulations, which are increasingly critical in blockchain-based infrastructures, especially in public sectors such as finance, energy, and healthcare. Regulatory authorities often impose strict requirements on data transmission, retention, and traceability to ensure auditability and lawful data use. Our architecture is therefore designed with built-in support for transparent logging and auditable workflows, which not only enhance trustworthiness but also align with national regulatory standards for information governance. These features position the proposed system as a viable solution for deployment in government-regulated or policy-sensitive environments.

The major contributions of this paper are summarized as follows:

- A novel high-performance hierarchical communication architecture: We propose a hierarchical device–edge–chain architecture centered on software-defined edge gateways, specifically designed to bridge the efficiency and trust gap between massive off-chain devices and on-chain blockchain systems and achieve the critical performance target of processing tens of millions of data entries per second.
- Robust fault tolerance and scalability: We incorporate dynamic path switching and capacity-aware load redistribution mechanisms mediated by software-defined networking (SDN), ensuring high availability and scalability to handle link failures and fluctuating workloads effectively.
- End-to-end regulatory compliance: We develop a sidechain-based auditing system integrated within the architecture, providing transparent, tamper-proof logs for data provenance and workflow execution, ensuring adherence to regulatory requirements across the entire communication path.

The remainder of this paper is organized as follows: Section 2 reviews blockchain fundamentals and scalability challenges. Section 3 presents the system design, including the device–edge–chain architecture and key mechanisms such as SDN integration, trusted data processing, and regulatory auditing. Section 4 describes the experimental setup and evaluates performance and fault tolerance. Section 5 discusses observed bottlenecks and

optimization directions. Finally, Section 6 concludes this paper and outlines potential future research directions.

2. Background and Related Work

2.1. Blockchain Fundamentals

Blockchain technology constitutes a decentralized data management architecture rooted in distributed ledger technology (DLT). Its fundamental innovation resides in constructing tamper-resistant and transparent transaction recording systems through cryptographic algorithms and consensus mechanisms. Traditional centralized systems depend on singular authorities (e.g., banks, governments) for data storage and verification, introducing risks of single-point failures and elevated trust costs. Blockchain innovates by distributing data storage across network nodes, where each transaction undergoes collective validation before being chronologically linked into blocks to form a chain structure. This architecture integrates three core components: distributed ledgers, enabling global data synchronization, consensus mechanisms (e.g., Proof of Work (PoW), Proof of Stake (PoS), as well as more recent protocols like Practical Byzantine Fault Tolerance (PBFT), Delegated Proof of Stake (DPoS), and HotStuff [11–13]), ensuring inter-node state consistency, and smart contracts, supporting automated business execution through programmable logic.

2.2. Blockchain Scalability

Blockchain scalability denotes the capacity to maintain operational efficiency amidst expanding user bases, transaction throughput, and data volumes. With the explosive growth of applications like DeFi, NFTs, and the metaverse, network congestion and surging transaction costs have intensified, driving academic and industrial efforts toward innovative solutions including sharding, directed acyclic graphs (DAGs), and Layer 2 scaling. These technologies aim to transcend the "blockchain trilemma" constraints by enhancing performance while preserving decentralization and security.

Sharding technology horizontally partitions networks into multiple parallel subchains (shards) to achieve localized and parallelized transaction processing. Ethereum 2.0 exemplifies sharding implementation, designed with 64 network shards managed by independent validator committees, coordinated through a beacon chain for cross-shard communication, targeting 10^4 TPS throughput. Critical technical advancements include transaction sharding (allocating transactions to specific shards by account addresses), state sharding (maintaining independent state databases per shard), and data availability sampling (light nodes verifying shard data integrity through random sampling). Nevertheless, sharding confronts challenges including cross-shard transaction atomicity guarantees, inter-shard state synchronization delays, and reduced per-shard security. OmniLedger [5] proposes a scalable distributed ledger maintaining long-term security in permissionless environments through the Atomix atomic commit protocol. RapidChain [6] implements an efficient consensus algorithm within committees using block pipelining for optimal throughput. Additional works [3,4] enhance sharded blockchain security and throughput through novel consensus protocols and architectural designs.

Directed Acyclic Graph (DAG) technology restructures data organization to overcome the serialization bottlenecks inherent in chain models, supporting asynchronous transaction verification and parallel confirmation. In DAG networks, each new transaction validates and links to multiple historical transactions, forming mesh-like topologies instead of linear chains, thereby eliminating block size constraints. The Tangle network by IOTA [14] exemplifies this through its "transaction-verifies-transaction" mechanism, enabling feeless micropayments with theoretically increasing throughput. Nano [15] employs a block–lattice architecture assigning independent chains to each account, achieving

asynchronous confirmation speeds exceeding 1000 TPS. Vite [16] adopts Nano's foundation while introducing a global snapshot chain for total ordering consistency. Meshcash [17] requires honest nodes to generate new blocks via Proof of Work (PoW), referencing all terminal blocks in their view. NEZHA [18] proposes a DAG-based concurrency control scheme that resolves conflicts arising from concurrent read–write operations on identical addresses during parallel transaction processing.

3. Materials and Methods

This study adopts a design-oriented methodology to address the communication and control challenges in blockchain-based systems. The proposed approach involves three key stages: (1) architectural design of a communication framework that integrates software-defined networking (SDN), edge computing, and blockchain consensus; (2) implementation of a prototype system to validate the feasibility of the design; and (3) performance evaluation through experimental testing on a controlled testbed. This structured methodology enables both conceptual innovation and practical verification, ensuring that the proposed framework can be realistically deployed in decentralized environments with dynamic network conditions.

3.1. System Design Analysis

First, confronting the dual challenges of blockchain performance bottlenecks and low-latency concurrent on-chain processing for tens of millions of data entries, our solution identifies the core issue as the terminal data generation rate exceeding blockchain processing capacity by approximately 1000:1, coupled with stringent end-to-end latency constraints. Rather than pursuing thousandfold improvements in core blockchain performance, we innovatively propose edge-side intelligent data dimensionality reduction calibrated to blockchain capabilities. Specifically, the framework deploys data compression and aggregation modules at edge gateways, featuring standardized processing for diverse data types (e.g., IPFS hashes, metadata, raw data) with predefined 100-byte on-chain entry benchmarks. The key innovation lies in domain-knowledge-driven semantic compression mechanisms, where edge gateways leverage application-specific knowledge bases (e.g., power systems) to achieve unprecedented compression ratios (e.g., 10:1 through temporal feature aggregation and redundant field consolidation). Concurrently, high-efficiency data aggregation algorithms optimize transaction packaging density. Through synergistic compression–aggression coordination, three-orders-of-magnitude data reduction is achieved. To ensure 1 s latency targets, adaptive on-chain processing dynamically adjusts compression strategies based on real-time monitoring of transmission delays and computational overheads. Furthermore, Anycast-based edge caching services alleviate mainchain access pressure, effectively reducing off-chain retrieval latency.

Second, addressing reliability and security in untrusted edge environments, we recognize edge gateways as critical trust anchors. Traditional solutions often neglect off-chain processing reliability. Our innovation integrates Trusted Execution Environments (TEEs), specifically Intel SGX, into edge gateways. All sensitive operations (data aggregation, semantic compression) execute within SGX enclaves, ensuring tamper resistance even against compromised host OSs. This extends prior work like [19] to complex edge scenarios. For data transmission, optimized cryptographic protocols (TLS/DTLS or SM algorithms) secure device–edge and edge–chain links. Crucially, our security lifecycle management ensures decrypted data immediately enters enclave processing before re-encryption, forming an end-to-end protection chain that eliminates exposure windows.

Third, targeting adaptive resilience requirements, we implement software-defined networking (SDN) at edge gateways. Through SDN controller-mediated management

(logically centralized), gateways gain programmable awareness of network states (load, connectivity). The controller dynamically optimizes traffic routing and resource allocation based on real-time metrics from gateway agents, synergizing with adaptive on-chain techniques. Advanced fault tolerance mechanisms enable automatic load redistribution during gateway overload and sub-500 ms failover to backup nodes during failures. This surpasses static SDN implementations [20–23] through deep integration with edge–blockchain interaction patterns.

Finally, addressing regulatory gaps in on-chain/off-chain communication, we design a multi-layer auditing system. Dual-network edge gateways feature isolated regulatory channels connecting to central platforms. A dedicated security sidechain stores tamper-evident audit proofs. Three-tier storage architecture combines the following: (1) raw signed events (local edge storage with 30-day retention), (2) semantic abstracts (central platform analysis), and (3) cryptographic proofs (sidechain preservation). Regulators access unified analytics through a security module querying all storage tiers, achieving comprehensive oversight without operational compromises.

3.2. Architecture Design

Building upon the preceding analysis, our proposed low-latency, high-security, adaptive, and regulatory-compliant on-chain/off-chain network communication framework centers on a three-tier device–edge–chain architecture anchored by software-defined edge gateways, as illustrated in Figure 1. Within this architecture, edge gateways fulfill four critical roles: (1) trusted entry points for massive heterogeneous device access to blockchain networks, (2) processing hubs for on-chain data aggregation/compression and off-chain data distribution, (3) core nodes enabling adaptive network management and fault tolerance, and (4) interfaces for communication auditing and regulatory system integration.

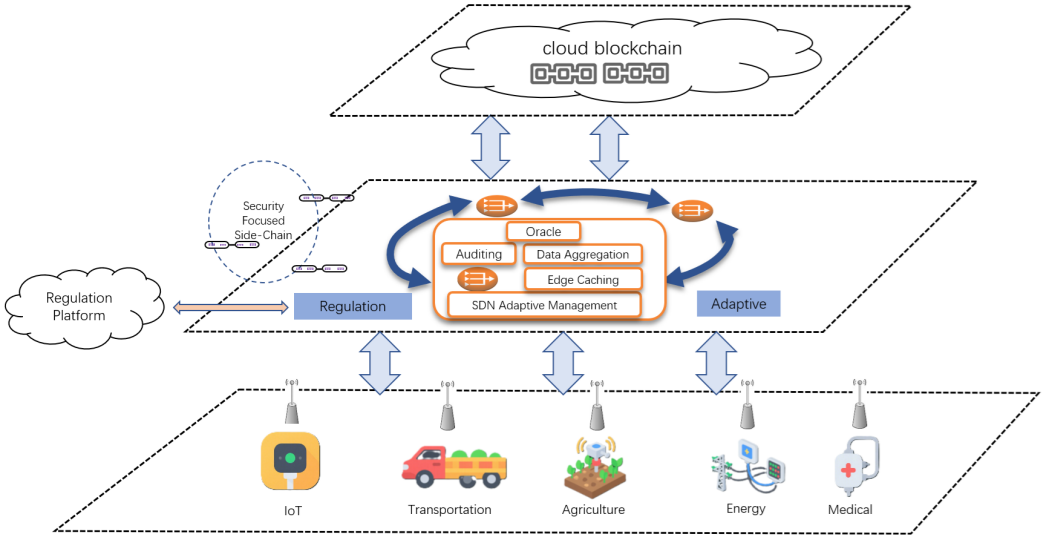


Figure 1. Device–edge–chain hierarchical communication architecture centered on software-defined edge gateways.

To realize these functionalities, edge gateways are designed as software-defined entities integrating SDN controller/agent capabilities. This design grants the edge network global visibility and flexible orchestration capacities, enabling dynamic routing adjustments, load balancing, and sub-second failover based on real-time network states and service demands. Such capabilities ensure system-wide adaptability and high fault tolerance.

Addressing performance and security requirements, the framework implements three core mechanisms within edge gateways: (1) lightweight secure transmission protocols (device–edge and edge–chain links), (2) Trusted Execution Environment (TEE)-protected data processing pipelines ensuring reliability and confidentiality, and (3) synergistic semantic compression algorithms with edge caching services. These components collectively resolve high-throughput and low-latency challenges while maintaining security guarantees.

For comprehensive regulatory compliance, the architecture establishes a dual-network auditing system comprising the following: (1) security-focused sidechains for tamper-proof audit logging, (2) hierarchical storage strategies (local edge storage, centralized platform analysis, and on-chain preservation), and (3) dedicated regulatory channels. Edge-gateway-embedded auditing components capture communication events, generating structured logs with semantic abstracts to optimize storage and chainspace utilization.

By integrating edge computing, software-defined networking, trusted computing, and blockchain regulatory technologies, this architecture forms a dynamically scalable, performance-stable, and end-to-end auditable communication framework. It systematically addresses the multifaceted challenges confronting existing solutions in complex application scenarios.

3.2.1. SDN Architecture

Each management domain comprises an intelligent edge gateway, multiple frontend SDN switches, and associated communication devices. Terminals connect to the intelligent edge gateway via frontend switches, which directly interface with cloud-based blockchain nodes. The SDN controller for each management domain resides on dedicated servers, while inter-domain switch connectivity is facilitated through external interfaces, as depicted in Figure 2.

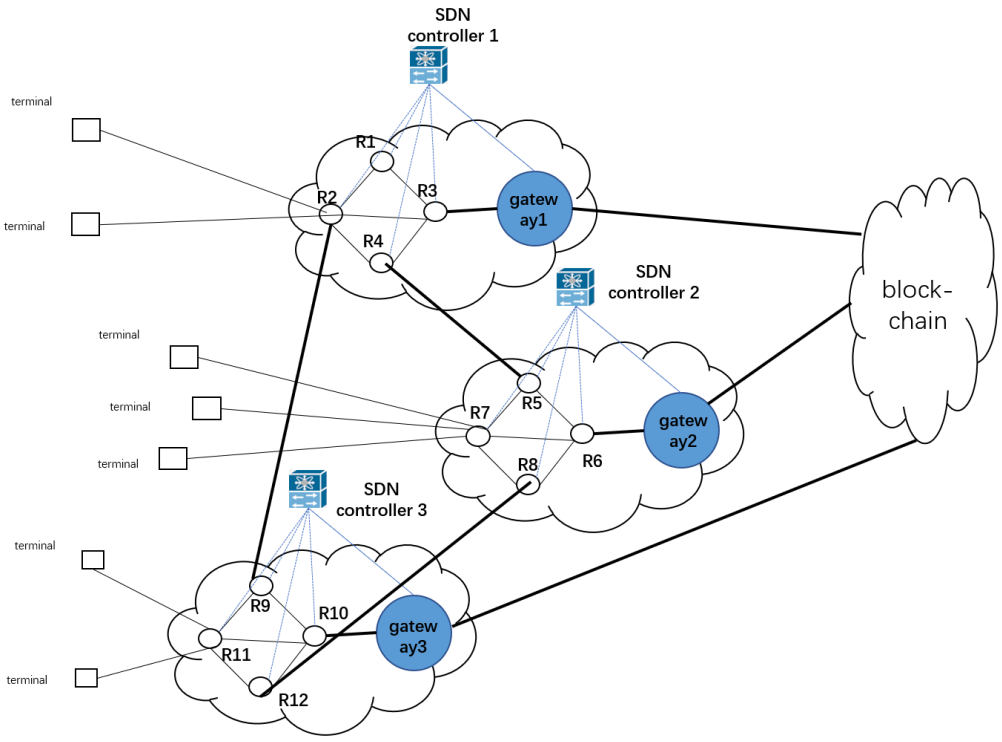


Figure 2. SDN topological structure.

The management domain SDN controller architecture integrates six core modules: (1) identity authentication, (2) QoS path construction, (3) network state updating, (4) reputation management, (5) self-status reporting, and (6) inter-controller QoS negotiation.

The QoS path construction module further incorporates path repair algorithms, concurrent QoS path generation mechanisms, and cross-domain QoS negotiation protocols.

Integration with the security-focused sidechain enables reliable inter-controller communication and decentralized SDN network operations. The sidechain performs three critical functions: (1) terminal identity management through smart contracts, (2) global network state storage (including domain-specific topology and controller reputation metrics), and (3) security validation for end-to-end (E2E) paths constructed by domain controllers.

QoS transmission paths generated by controllers are translated into flow rules and deployed to relevant switches, completing E2E path establishment. Identity credentials and verification data for all domain controllers are uniformly registered on the sidechain via an identity management smart contract. Cross-domain communications and on-chain data submissions require blockchain-verified identity authentication, ensuring both chain data integrity and inter-controller communication security.

The global network state aggregates all management domain states, each comprising four elements: (1) intra-domain node interconnection topology, (2) resource utilization metrics, (3) QoS capability profiles, and (4) inter-domain routing status. Domain controllers model local states as weighted directed graphs, with the complete network state emerging from federated graph integration across domains.

For state synchronization, domain controllers implement a privacy-aware update protocol: (1) propagating latest local states to active peer controllers via authenticated channels, and (2) submitting state hash values to the sidechain for verification. To balance efficiency and security, controllers may periodically query the sidechain for global updates or directly request state information from peers, validated against sidechain records.

3.2.2. Edge Gateway Architecture

The edge gateway comprises four core components: (1) a trusted data processing module, (2) ab auditing component, (3) an edge caching service, and (4) an off-chain module. The trusted data processing module performs decryption, summarization, compression, and aggregation operations on on-chain-bound data. The oracle off-chain module conducts credibility assessments for data before initiating secure transmission to the blockchain. The auditing component captures traffic information to generate regulatory logs, storing raw logs locally while submitting semantically abstracted summaries to the security sidechain. Figure 3 illustrates the system architecture.

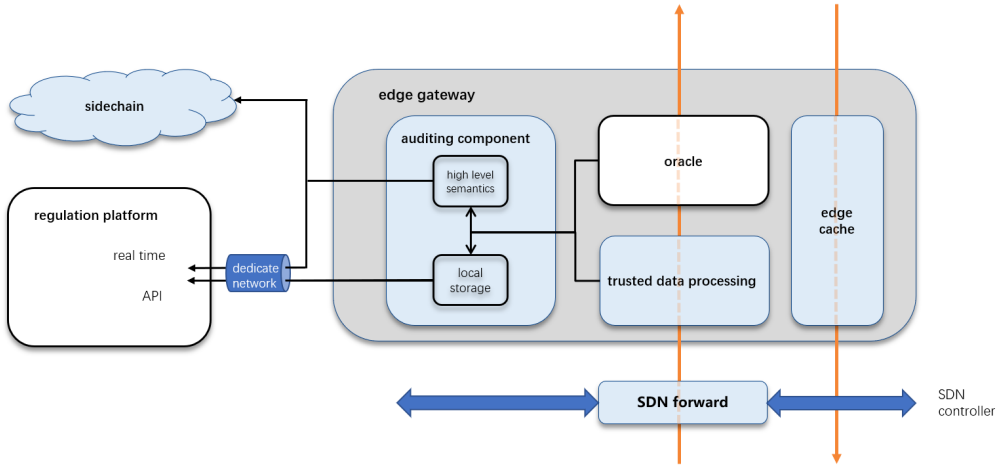


Figure 3. Edge gateway system architecture.

The gateway employs high-performance network servers to handle terminal requests, utilizing lightweight user-space handlers to manage hundreds of thousands of concurrent

on-chain requests. Upon receiving requests, handlers write them to processing queues and acknowledge terminal connections immediately, enabling higher concurrency through early connection termination.

The processing queue utilizes AVX512-optimized SHA-256 algorithms for efficient transaction hashing. Processed transactions bifurcate into two categories: (1) domain-specific data requiring semantic compression with provided algorithms, and (2) structured time-series data from high-frequency sources (e.g., $\geq$10 Hz). For time-series data, the compression engine merges identical metadata fields and encodes values using "base value + offset" schemes. Compression latency is strictly capped at 200 ms to meet end-to-end deadlines. Re-hashing follows compression to generate new transaction identifiers.

All processed transactions adopt a standardized triple structure: (transaction, index, hash). These triples are stored in a high-performance distributed database with dual indexing on index and hash fields for efficient retrieval.

Concurrently, transactions enter the data aggregation pipeline through a shared-memory ring buffer interface. Multiple worker threads consume transaction hashes from the buffer, constructing Merkle trees in parallel. Each thread maintains local tree leaves until either reaching capacity thresholds (node saturation) or timeout triggers (maximum 500 ms). Workers then compute hierarchical hashes bottom-up until root hash generation. This workflow is formally detailed in Algorithm 1.

Algorithm 1 Hash tree aggregate algorithm.

Input: $RingBuffers$
$MaxLeaves$: max leaves of a tree (default 32,768)
$Timeout$: batch timeout (default 100 ms)
Output: $Forest$: generated hash tree

init global hash chain $Chain \leftarrow \emptyset$
create work thread pool $Workers[1..N]$
allocate private mempool for each worker $MemPool_i$
for each worker $W_i \in Workers$ **do**
  while system running **do**
    $LocalLeaves \leftarrow \emptyset$
    $Timer \leftarrow Time.Now()$
    while $|LocalLeaves| \leq MaxLeaves$ **and** $Time.Now() < Timer + Timeout$ **do**
      read hash from $RingBuffers$
    end while
    build hash tree T:
    for $level \leftarrow 1$ **to** $\log_2(MaxLeaves)$ **do**
      for $j \leftarrow 0$ **to** $|Nodes_{level-1}|/2$ **do**
        $parentHash \leftarrow \text{SHA256}(Nodes_{level-1}[2j] \parallel Nodes_{level-1}[2j+1])$
        $Nodes_{level} \leftarrow Nodes_{level} \cup \{parentHash\}$
      end for
    end for
    $rootHash \leftarrow Nodes_{top}[0]$
    if $Chain \neq \emptyset$ **then**
      $chainedRoot \leftarrow \text{SHA256}(\text{[For chainedRoot in Chain If chainedRoot.isLeaf()]} \parallel rootHash)$
    else
      $chainedRoot \leftarrow rootHash$
    end if
    $Chain \leftarrow Chain \cup \{chainedRoot\}$
    update DB
  end while
end for

To ensure data integrity while preventing worker blocking, the aggregation system implements a novel chaining mechanism. Rather than linear blockchain-style linking, each new root hash connects to all unlinked predecessors, forming a tamper-evident directed acyclic graph (DAG). This enables non-blocking concurrent appends while maintaining cryptographic audit trails.

Completed Merkle trees return to the gateway with corresponding root hashes. The gateway updates database records with tree positional metadata (hash, root, path) and submits trees to oracles for validation. Upon receiving blockchain-confirmed (root, hash, block) mappings, the gateway finalizes database updates to enable efficient off-chain queries.

3.2.3. Data On-Chaining Process

The on-chaining workflow integrates intelligent routing, secure transmission, and real-time auditing to ensure end-to-end verifiability and timeliness from terminals to blockchain, as illustrated in Figure 3. Key stages include the following:

Upon transaction initiation, software-defined networking (SDN) enables intelligent traffic orchestration. The nearest SDN switch forwards requests to its controller, which performs real-time network analysis evaluating the following: (1) edge gateway load metrics (CPU utilization, memory footprint, network queue depth), and (2) link quality parameters (latency, packet loss). A dynamic weighted algorithm selects optimal gateways, triggering OpenFlow rule updates for failover routing when thresholds are exceeded (e.g., CPU > 80%).

At the edge gateway, Transport Layer Security (TLS) decryption precedes Trusted Execution Environment (TEE)-based processing: (1) semantic-aware lossless compression through deep feature extraction, and (2) multi-source aggregation via enhanced Merkle tree structures. Concurrently, the auditing subsystem captures dual-layer traces, transaction-level primitives (timestamps, data fingerprints) and network-layer metadata (source IPs, transmission durations), storing categorized records locally while pushing critical features to regulatory platforms via secure channels.

Post-processing, oracle networks perform compliance verification against smart contract-defined rules (data schema validation, business logic checks). Validated transactions transmit via optimized HTTP/3 (QUIC-based implementation) to cloud blockchain nodes, leveraging 0-RTT connection establishment and multiplexing capabilities to minimize network latency.

3.2.4. Data Off-Chaining Process

The off-chaining mechanism employs edge-priority architecture with Redis caching and distributed query optimization, as shown in Figure 3's off-chain path.

Terminal requests route through SDN controllers to proximal edge gateways. Local Redis caches with LRU eviction policies and auto-expiry mechanisms enable millisecond-level responses for cache hits. Missed queries trigger distributed resolution: (1) metadata index services (built on distributed key-value stores) provide blockchain coordinates (block heights, transaction indices, Merkle tree paths), and (2) P2P queries prioritize least recently used neighboring gateways via intelligent routing.

Cloud blockchain nodes return full blocks containing target data. Gateways execute three-tier validation upon receipt: (1) block header hash verification, (2) Merkle proof validation for transaction inclusion, and (3) digital signature checks for integrity. Validated data populates Redis caches with time-to-live (TTL) settings while returning complete datasets (raw data, blocks, Merkle trees) to terminals.

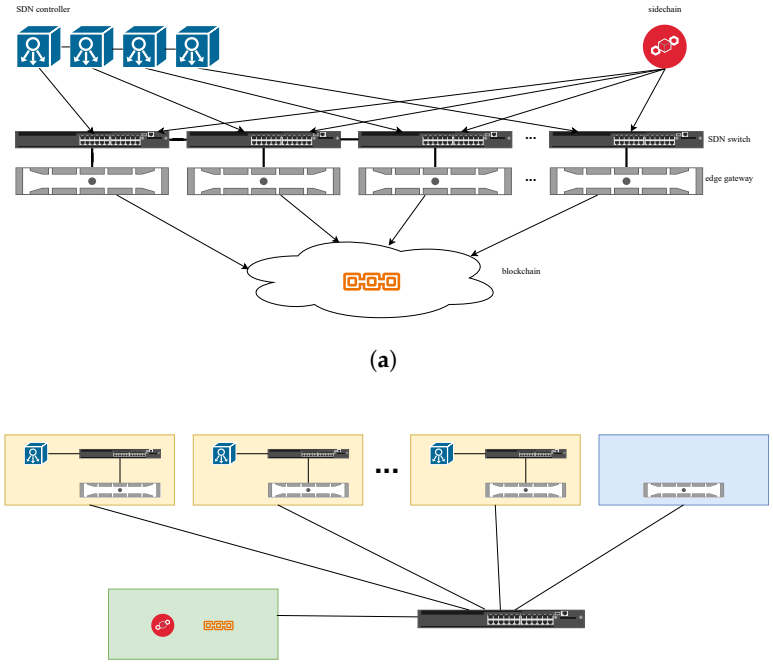
4. Experimental Evaluation

We fully implemented the proposed on-chain/off-chain communication system based on SDN and edge gateway clusters. This section evaluates system performance (throughput, latency) and fault tolerance capabilities.

4.1. Experimental Configuration

4.1.1. Network Topology

The experimental topology comprises logical and physical layouts shown in Figure 4. The logical design (Figure 4a) places an SDN switch before each edge gateway, with interconnected SDN controllers and all gateways connecting to security sidechains and cloud blockchain nodes. Physical deployment (Figure 4b) allocates each (gateway, switch, controller) group to dedicated servers (yellow boxes), with blockchain servers in green boxes. Two load generators and cross-rack VPN tunneling complete the setup.



(**a**)

(**b**)

Figure 4. On-chain/off-chain communication test topology. (**a**) Logical test topology. (**b**) Physical deployment.

Given single-gateway throughput limits (300,000 TPS from stress tests), 35 physical servers host edge gateways to achieve 10 million TPS capacity. Two additional servers generate test requests, with one dedicated blockchain server, totaling thirty-eight nodes.

4.1.2. Parameter Settings

To accommodate heterogeneous CPUs (Intel Xeon Silver 4214/4110, E5-2650 v4), we standardize parallel processing configurations: four ring buffer groups and four worker threads per gateway. Both cloud blockchain and security sidechain use FISCO BCOS implementation.

4.1.3. Traffic Generation

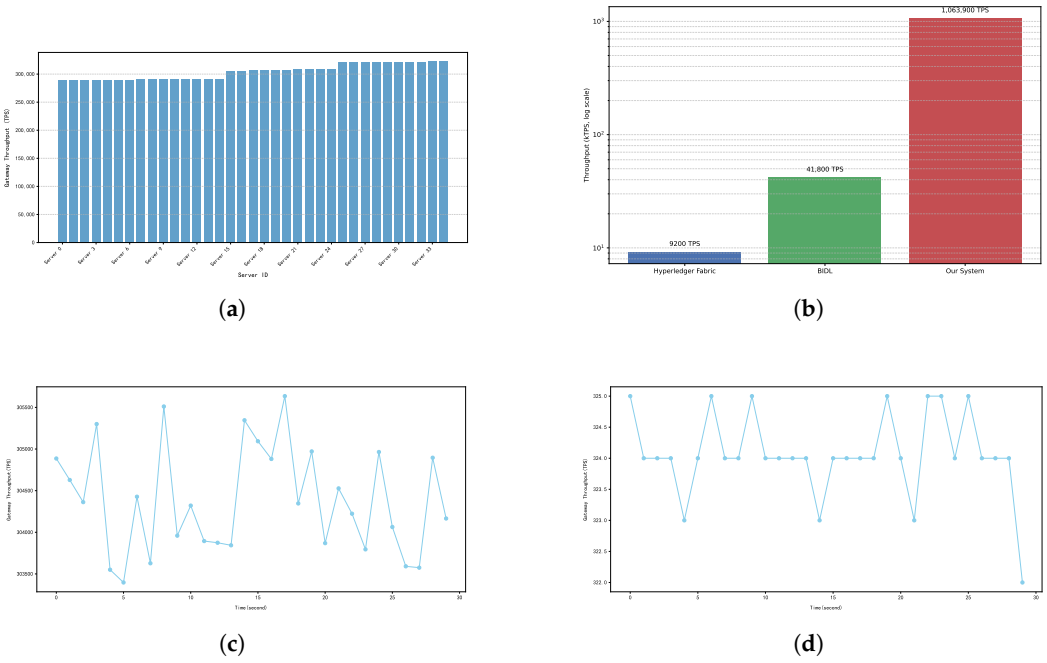
Performance tests employ go-stress-testing for uniform request distribution across gateways. Fault tolerance tests use the Locust framework with programmable traffic patterns. Each request contains randomly sized payloads: 100–400 bytes for transaction content and 10–40 bytes for indices.

4.1.4. Test Limitations

The experimental environment cannot support intended high-performance distributed databases. Gateway–database interactions remain unimplemented, with MySQL tests showing 97% failure rates (only about 9000 TPS sustained). Off-chain query performance depends entirely on database implementations and is thus excluded from the results. Audit functionality validation is similarly omitted due to dependency constraints.

4.2. System Performance Evaluation

The system's capacity to achieve 10 million TPS throughput was evaluated using 35 edge gateways over a 2 min test period. Results are presented in Figure 5.



(**a**) (**b**)

(**c**) (**d**)

Figure 5. Performance evaluation of on-chain/off-chain communication system. (**a**) Aggregate on-chain throughput across edge gateways. (**b**) Throughput comparison of blockchain systems (log scale). (**c**) 30 s Throughput of single edge gateway. (**d**) 30 s blockchain layer throughput.

Figure 5a displays the aggregate throughput across all gateways, revealing three performance tiers among the 35 nodes (Groups 0–14, 15–24, and 25–34). This stratification stems from heterogeneous CPU architectures and single-core performance variations across servers. The cumulative throughput reaches 10,639,666 TPS with individual gateways averaging approximately 300,000 TPS.

Figure 5b presents a throughput comparison between our proposed system and representative blockchain architectures, including Hyperledger Fabric (HLF) and BIDL, plotted on a logarithmic scale. The results clearly demonstrate the performance advantage of our architecture, which achieves over 10 million TPS—approximately 250× that of HLF (9.2 k TPS) and over 25× that of BIDL (41.8 k TPS). This significant improvement validates the effectiveness of our hierarchical edge–chain architecture in supporting ultra-high-throughput applications.

Figure 5c details the stabilized throughput of a representative gateway after initial transient fluctuations, maintaining 304,000 TPS with ±1000 TPS variance through 1 s sampling intervals.

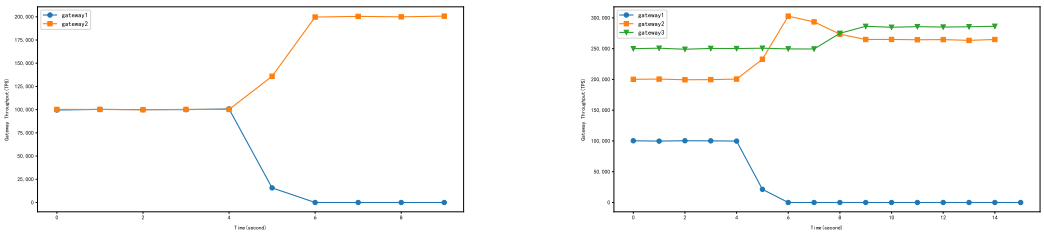
Figure 5d demonstrates cloud blockchain throughput stabilizing at 324 TPS. Given the Merkle tree's maximum leaf node capacity (32,768 entries per root hash), this correlates to

324 × 32,768 = 10,616,832 effective transactions, validating the aggregation mechanism's design efficacy.

4.3. Fault Tolerance Evaluation

Figure 6a validates the system's fundamental fault detection and path switching capabilities. Two edge gateways initially handle 100,000 TPS each (generated by Locust framework). At the 4th second, Gateway 1 is shut down to simulate link/gateway failure. Experimental results show that Gateway 1's throughput plummets at the 4th second and stabilizes at zero from the 5th second onward, indicating failure occurrence during the 4th second. Gateway 2's throughput begins increasing at the 4th second and stabilizes after the 5th second, fully taking over Gateway 1's traffic. The combined throughput of both gateways during the 4th second falls below 200,000 TPS due to detection latency between actual failure and SDN controller awareness. The controller monitors gateway status through 500 ms interval heartbeat signals from gateway agents. The sub-500 ms detection window causes temporary packet loss for requests routed to Gateway 1 before flow table updates complete. Full traffic migration to Gateway 2 occurs within 1 s after the controller deploys predefined backup path rules.

Figure 6b demonstrates redundant path selection capabilities. Three gateways operate with initial loads of 100,000/150,000/250,000 TPS, respectively. Gateway 1 failure at the 4th second triggers the following: Gateway 2's throughput surges from the 4th second, peaks at the 6th second, then gradually declines before stabilizing after the 9th second. Gateway 3's throughput increases from the 8th second onward, reaching stability after the 9th second. Analysis reveals that the SDN controller initially redirects all Gateway 1 traffic to Gateway 2 (similar to Figure 6a), but their combined load (100 k + 150 k = 250 k TPS) exceeds Gateway 2's 150 k TPS capacity. Throughput degradation occurs at the 7th second due to overloaded burst processing. Algorithm 1 then calculates available capacity ratios (Gateway 2: 150 k → 50 k residual, Gateway 3: 250 k → 50 k residual) and proportionally redistributes traffic through updated flow tables deployed at the 8th second. This enables complete traffic by Gateways 2/3 through intelligent load splitting.



(**a**) (**b**)

Figure 6. Fault tolerance test results. (**a**) Single failover scenario. (**b**) Redundant path selection scenario.

5. Discussion

This work employs cryptographic algorithm-based secure transmission protocols to ensure communication security during data on-chaining processes. Additionally, edge gateways compute hash values as transaction indices following data decryption. Performance analysis under high-load conditions identifies data decryption and hash computation as the primary system bottlenecks, collectively consuming over 80% of total CPU resources. Emerging smart network interface cards (e.g., Mellanox CX6) demonstrate hardware-level capabilities for cryptographic operations and hash acceleration. By offloading these computationally intensive tasks to dedicated hardware, future implementations could substantially enhance edge gateway throughput and overall system efficiency.

Deployment Considerations in Cost-Sensitive Environments

While the proposed architecture demonstrates strong performance and reliability through a multi-node SDN and edge gateway deployment, its real-world applicability in cost-sensitive environments warrants careful analysis. Deploying and maintaining dozens of edge gateways, along with SDN infrastructure and regulatory sidechains, may introduce substantial capital and operational expenditures. This could pose adoption barriers for small-to-medium enterprises or resource-constrained governmental agencies.

To address this, several cost-optimization strategies can be employed. First, edge gateways can be implemented using lightweight virtualization or container-based instances running on commodity hardware, reducing hardware requirements. Second, a tiered deployment model can be adopted, where high-performance gateways are selectively deployed in critical regions, while lightweight nodes handle less time-sensitive data. Third, existing infrastructure—such as 5G MEC servers or regional data centers—can be repurposed as edge nodes to minimize deployment overhead.

Furthermore, core system modules, including auditing and TEE-based processing, can be modularized and selectively enabled based on application-specific requirements and compliance levels. In future work, we plan to explore serverless edge deployments, cloud–edge hybrid offloading models, and integration with network function virtualization (NFV) frameworks to enable more flexible and cost-effective deployment in diverse scenarios.

6. Conclusions and Future Work

This paper presents a comprehensive on-chain/off-chain communication framework tailored for blockchain systems operating under high-throughput and low-latency requirements. By introducing a hierarchical device–edge–chain architecture centered on software-defined edge gateways, the proposed system systematically addresses performance, adaptability, security, and regulatory compliance challenges.

Experimental evaluations validate the system's capability to process over 10 million transactions per second (TPS) with sub-second end-to-end latency. Specifically, the aggregation mechanism enables a single edge gateway to sustain 300,000 TPS, while the entire system—scaling across 35 physical gateways—achieves a peak throughput of 10.64 million TPS. Fault-tolerance tests confirm that the SDN-based architecture can detect and recover from gateway failures within 1 s, with dynamic load redistribution ensuring continued service availability under stress conditions.

The key technical contributions include the following:

- An edge-centric semantic compression and aggregation pipeline achieving up to 1000:1 effective data reduction, bridging the gap between device data generation and blockchain processing capacity;
- A TEE-protected processing environment at the edge, ensuring confidentiality and integrity even under untrusted infrastructure;
- An SDN-controlled network layer that enables sub-second failover, capacity-aware routing, and global network visibility;
- A regulatory-compliant auditing subsystem based on a dedicated sidechain and multi-layer storage model.

Future work will be focused in several directions. First, we plan to integrate smart network interface cards (SmartNICs) to offload cryptographic and hashing operations, aiming to further boost per-node throughput. Second, the current architecture can be extended to support multi-chain environments, introducing cross-chain communication protocols and compatibility layers. Third, we will implement and evaluate the off-chain querying subsystem using high-performance distributed databases and caching mechanisms. Finally, enhancing the audit subsystem with real-time analytics and AI-driven

anomaly detection will further strengthen the regulatory framework and enable intelligent compliance monitoring.

Author Contributions: Gateway design and implementation, Y.J.; SDN design and implementation, D.H.; system architecture design, C.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported in part by the National Key Research and Development Program of China under grant number 2022YFB2702800 and The Key Program of Natural Science Foundation of Jiangsu under grant number BK20243053.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System. 2008. Available online: https://bitcoin.org/bitcoin.pdf (accessed on 1 July 2025).
2. Poon, J.; Dryja, T. The Bitcoin Lightning Network. 2016. Available online: https://lightning.network/lightning-network-paper.pdf (accessed on 1 July 2025).
3. Wang, J.; Wang, H. Monoxide: Scale out blockchains with asynchronous consensus zones. In Proceedings of the 16th USENIX Symposium on Networked Systems Design and Implementation (NSDI 19), Boston, MA, USA, 26–28 February 2019; USENIX Association: Berkeley, CA, USA, 2019; pp. 95–112.
4. Hong, Z.; Guo, S.; Li, P.; Chen, W. Pyramid: A layered sharding blockchain system. In Proceedings of the IEEE INFOCOM 2021—IEEE Conference on Computer Communications, Virtual, 10–13 May 2021; pp. 1–10.
5. Kokoris-Kogias, E.; Jovanovic, P.; Gasser, L.; Gailly, N.; Syta, E.; Ford, B. Omniledger: A secure, scale-out, decentralized ledger via sharding. In Proceedings of the 2018 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 20–24 May 2018; pp. 583–598.
6. Zamani, M.; Movahedi, M.; Raykova, M. Rapidchain: Scaling blockchain via full sharding. In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, CCS '18, Toronto, ON, Canada, 15–19 October 2018; Association for Computing Machinery: New York, NY, USA, 2018; pp. 931–948.
7. Qi, J.; Chen, X.; Jiang, Y.; Jiang, J.; Shen, T.; Zhao, S.; Wang, S.; Zhang, G.; Chen, L.; Au, M.; et al. Bidl: A High-Throughput, Low-Latency Permissioned Blockchain Framework for Datacenter Networks. In Proceedings of the ACM SIGOPS 28th Symposium on Operating Systems Principles, Virtual, 26–29 October 2021; pp. 18–34. [CrossRef]
8. Sun, G.; Jiang, M.; Khooi, X.; Li, Y.; Li, J. NeoBFT: Accelerating Byzantine Fault Tolerance Using Authenticated In-Network Ordering. In Proceedings of the ACM SIGCOMM 2023 Conference, New York, NY, USA, 10 September 2023; pp. 239–254. [CrossRef]
9. Wei, X.; Cheng, R.; Yang, Y.; Chen, R.; Chen, H. Characterizing Off-path SmartNIC for Accelerating Distributed Systems. In Proceedings of the 17th USENIX Symposium on Operating Systems Design and Implementation (OSDI 23), Boston, MA, USA, 10–12 July 2023; pp. 987–1004. Available online: https://www.usenix.org/conference/osdi23/presentation/wei-smartnic (accessed on 27 March 2025).
10. Zhou, Y.; Wang, Z.; Dharanipragada, S.; Yu, M. Electrode: Accelerating Distributed Protocols with eBPF. In Proceedings of the 20th USENIX Symposium on Networked Systems Design and Implementation (NSDI 23), Boston, MA, USA, 17–19 April 2023; pp. 1391–1407. Available online: https://www.usenix.org/conference/nsdi23/presentation/zhou (accessed on 27 March 2025).
11. Castro, M.; Liskov, B. Practical Byzantine fault tolerance. In Proceedings of the Third Symposium on Operating Systems Design and Implementation (OSDI), New Orleans, LA, USA, 22–25 February 1999; pp. 173–186.
12. Larimer, D. Delegated Proof-of-Stake (DPoS). 2014. Available online: https://bitshares.org/technology/delegated-proof-of-stake-consensus/ (accessed on 22 July 2025).
13. Yin, M.; Malkhi, D.; Reiter, M.; Gueta, G.; Abraham, I. HotStuff: BFT Consensus in the Lens of Blockchain. In Proceedings of the 2019 ACM Symposium on Principles of Distributed Computing (PODC), Toronto, ON, Canada, 29 July–2 August 2019; pp. 347–356.
14. IOTA Stiftung. About IOTA. 2025. Available online: https://docs.iota.org/about-iota (accessed on 27 March 2025).

15. Nano Foundation. Nano Documentation. 2025. Available online: https://docs.nano.org/protocol-design/introduction/ (accessed on 27 March 2025).
16. Vite Labs Limited. ZERO GAS LAYER-1. 2018. Available online: https://vite.org/ (accessed on 27 March 2025).
17. Bentov, I.; Hubáček, P.; Moran, T.; Nadler, A. Tortoise and hares consensus: The meshcash framework for incentive-compatible, scalable cryptocurrencies. In *Cyber Security Cryptography and Machine Learning*; Dolev, S., Margalit, O., Pinkas, B., Schwarzmann, A., Eds.; Springer International Publishing: Cham, Switzerland, 2021; pp. 114–127.
18. Xiao, J.; Zhang, S.; Zhang, Z.; Li, B.; Dai, X.; Jin, H. Nezha: Exploiting concurrency for transaction processing in dag-based blockchains. In Proceedings of the 2022 IEEE 42nd International Conference on Distributed Computing Systems (ICDCS), Bologna, Italy, 10–13 July 2022; pp. 269–279.
19. Zhang, F.; Cecchetti, E.; Croman, K.; Juels, A.; Shi, E. Town crier: An authenticated data feed for smart contracts. In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, CCS '16, Vienna, Austria, 24–28 October 2016; Association for Computing Machinery: New York, NY, USA, 2016; pp. 270–282.
20. Atwal, K.S.; Bassiouni, M. Softaccess: Cloud-based software defined virtualized wireless mobile access networks. In Proceedings of the 2017 5th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud), San Francisco, CA, USA, 6–8 April 2017; pp. 96–101.
21. Kim, H.; Schlansker, M.; Santos, J.R.; Tourrilhes, J.; Turner, Y.; Feamster, N. Coronet: Fault tolerance for software defined networks. In Proceedings of the 2012 20th IEEE International Conference on Network Protocols (ICNP), Austin, TX, USA, 30 October–2 November 2012; pp. 1–2.
22. Li, H.; Li, Q.; Jiang, Y.; Zhang, T.; Wang, L. A declarative failure recovery system in software defined networks. In Proceedings of the 2016 IEEE International Conference on Communications (ICC), Kuala Lumpur, Malaysia, 22–27 May 2016; pp. 1–6.
23. Wu, D.; Arkhipov, D.I.; Asmare, E.; Qin, Z.; McCann, J.A. Ubiflow: Mobility management in urban-scale software defined iot. In Proceedings of the 2015 IEEE Conference on Computer Communications (INFOCOM), Hong Kong, China, 26 April–1 May 2015; pp. 208–216.