SvedbergOpen
DISSEMINATION OF KNOWLEDGE

International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: https://www.svedbergopen.com/

Research Paper Open Access

A Blockchain-Decoupled QoS Framework for Secure and Low-Latency Tactile Internet over SDN

Mohit Marwaha[1*], Dr. Rajeev Bedi[2]

[1]Department of Computer Science, Sardar Beant Singh State University, Gurdaspur, Punjab, India
[2]I.K. Gujral Punjab Technical University, Jalandhar, Punjab, India

Abstract
Network security and Quality of Service (QoS) assurance pose dual challenges that conventional Software-Defined Networking (SDN) architectures address only in isolation. This paper proposes a unified SDN-Blockchain-Tactile Internet framework. It introduces the Blockchain Decoupled QOS Scheduling Algorithm (BQSA). This formal scheduling mechanism enforces latency-class-aware traffic prioritization while decoupling Hyperledger Fabric consensus operations from the data plane forwarding critical path. The proposed architecture integrates a Ryu SDN controller with a permissioned Hyperledger Fabric Blockchain operating under Raft consensus, augmented by Tactile Internet-inspired traffic classification and fast path forwarding for delay-sensitive flows. BQSA is formally specified using pseudocode, with an O(n log n) complexity analysis, and a control plane latency-bound theorem that guarantees Blockchain validation overhead does not propagate to Tactile flow forwarding delay. Experimental evaluation in a Mininet 2.3.0 emulation environment with Hyperledger Fabric v2.5.14 demonstrates that the proposed framework (C3) achieves a latency of 16.0 ms, jitter of 2.8 ms, throughput of 96.0 Mbps, and packet loss of 1.0%, outperforming both traditional SDN (C1) and SDN-Blockchain (C2) baselines across all four QoS dimensions and comparing favorably with representative values reported in prior SDN Blockchain literature. Blockchain-based flow rule integrity validation achieves a 96-97.5% detection rate for unauthorized rule modification with an average response time of 11-13 ms. These results establish that architectural decoupling of Blockchain validation from forwarding-critical operations is a viable and effective strategy for building networks that are simultaneously secure, auditable, and QoS-responsive.

Keyword: Software-Defined Network (SDN), Blockchain, Tactile Internet, Network security, BQSA and Flow rule integrity.

I. Introduction

Recent industry analysis predicts a dramatic increase in global network traffic over the next decade. Some reports like Nokia's Global Network Traffic Report, forecast that worldwide network traffic will grow five to nine times by 2033, primarily driven by AI-powered applications, video streaming, and adoption of 5G and use cases [1]. Similarly, research by Sofrecom anticipates that total fixed-line internet traffic will double and mobile data traffic will triple by 2030, with 5G networks expected to carry upto 75% of Global mobile data by 2028 [2]. These forecasts have been confirmed by research organizations such as GRAND View Research, which highlight the increasing importance of implementing advanced network analytics solutions to address the increase in the volume of traffic [3]. Modern communication networks are becoming more complicated and dynamic, thus requiring a novel way to effectively handle the Data traffic. Traditionally, networking relies on hardware solutions along with distributed control methods that are rather limited and vulnerable.

Software-Defined Networks represent one of the most viable paradigms that have been proposed to address the aforementioned problems, thanks to its ability to improve the programmability of the network, as well as its ability to offer traffic control through the controller. The main reason behind this is the ability of SDN to decouple the control plane and the data plane within the network, enabling the administrator to manage the operation of the network under real-time circumstances. Fig. 1 illustrates the basic design structure wherein a centralized controller can connect with switches that are located in a distributed manner using OpenFlow. This helps in efficient management of the network and deployment of services and policies.

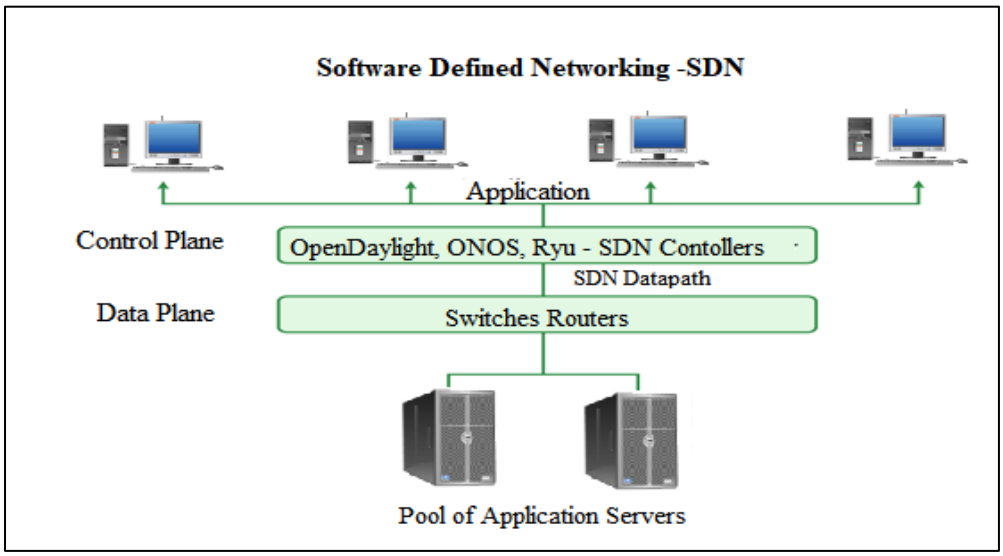

Fig. 1. Basic Architecture of Software-Defined Network

Real-time traffic monitoring and the ability to programmatically isolate and quarantine suspicious traffic enhance an organization's ability to contain and recover from threats; however, it is not enough to fight the threats that can come from unauthorized changes, rule tampering, and flow spoofing, etc to the network process. These vulnerabilities directly impact the reliability and performance of latency-sensitive applications [4]. These Vulnerabilities need dynamic security frameworks that will not only provide security against unauthorized entry and data but also ensure the integrity and availability of the system.

In Blockchain, the consensus mechanism guarantees that all nodes in the network have a shared and unaltered view of the transactions, which are important for auditing, compliance, and preventing unauthorized changes to network configurations [5].

The Tactile Internet, which provides haptic feedback and real-time interactivity with latency below 1 millisecond, represents the future of communication networks. It goes beyond conventional data transfer by offering fast and sensory experiences, which is critical in applications that require responsiveness [6]. By combining the Tactile Internet-inspired principles with SDN and Blockchain, we introduce a framework that secures the network and optimizes real-time performance, though not yet meet the strict sub-millisecond Tactile Internet standards in the current emulation-based evaluation (Section 5).

SDN programmability allows for the dynamic allocation of resources and the enforcement of differentiated forwarding policies, while a permissioned Blockchain secures control plane operations through immutable, consensus-validated logging. Tactile Internet-grade applications impose hard QoS constraints, such as latency below 1 ms in ideal physical deployment, minimal jitter and near-zero packet loss that cannot tolerate the additional control plane overhead that Blockchain consensus typically introduces. The central engineering challenge addressed by this paper is therefore not simply combining these three technologies but doing so in a way that prevents the security layer from becoming a QoS bottleneck.

To address this challenge, we introduced the Blockchain Decoupled QoS Scheduling Algorithm (BQSA), a formal scheduling mechanism embedded in the SDN control plane that classifies flows by latency class, enforces priority-based forwarding, and routes Blockchain validation operations through a non-blocking side channel so that consensus latency does not propagate to tactile flow forwarding delay. The primary contributions of this paper are as follows:

1. A unified six-component SDN-Blockchain-Tactile internet architecture integrating Ryu. Hyperledger Fabric with Raft consensus and tactile internet traffic modules in a single experimental framework.
2. The BQSA algorithm is a formally specified, complexity-analyzed scheduling mechanism with a provable control plane latency bound that decouples Blockchain validation from data plane forwarding critical path.
3. Experimental QoS evaluation across four metrics (latency, jitter, throughput, packet loss) comparing three configurations: traditional SDN, SDN-Blockchain, and the proposed SDN-Blockchain-Tactile internet framework in a reproducible Mininet-based testbed.
4. A flow rule integrity evaluation demonstrating that Blockchain-based consensus validation detects 96-97.5% of unauthorized rule modification attempts with sub-13 ms response time, establishing the security baseline for future attack-aware extensions.

II. Background and Related Works

Prior foundational work establishes a three-layer SDN architecture built on OpenFlow, demonstrating significant programmability and flexibility for dynamic network management. Despite these advances, such a framework lacks mechanisms for distributed security, decentralized trust, and comprehensive Quality of Service support such as latency, jitter, throughput and packet loss control that real-world deployments demand. While SDN provides centralized control and dynamic flow management, Blockchain introduces decentralized trust and tamper-proof logging and Tactile Internet applications impose stringent QoS requirements, including low latency, minimal jitter and high reliability. A comparative summary of the most relevant studies is presented in **Table 1**. Notably, frameworks such as HB-SDN-IOT [7] and SmartSecChain-SDN [8] emphasize security and trust but stop short of explicitly optimizing QoS for latency-sensitive tactile workloads; this critical gap establishes a core motivation for a unified SDN-Blockchain-tactile framework.

Table 1. Literature analysis based on methodology, advantages, disadvantages, and Gaps identified

Reference	Methodology	Advantages	Disadvantages	Gaps identified
[7]	Hybrid Blockchain-SDN for secure routing in IoT uses proof of authority (PoA) and proof of work (PoW) for clustering	Reduces Energy consumption by 23%, achieves 96.8% packet delivery and is robust against Sybil attack	Complex Dual-layer architecture, High overhead in an adversarial environment. .	Limited to simulation, no real-time responsiveness
[8]	Blockchain integrated ML IDS (SmartSecChain-SDN) for SDN uses RF, XGBoost, etc	High Accuracy, immutable Logs	High Complexity in integration	Gap in 6G scalability and real-time testing
[9]	Federated Learning and enhanced resource allocation (Tactiflex) for TI in 6G.	Achieves high accuracy in time-series data.	High Complexity in FL: centralized limitation	Gaps in full-scale 6G testing, a real-world privacy evaluation is needed
[10]	Proposes “DistB-SDcloud” a distributed Blockchain-SDN architecture for secure cloud in smart IIot, integrates permissioned Blockchain for data integration/privacy and SDN for network durability and payload balancing, evaluated by simulation measuring, throughput, packet analysis and response time and attack monitoring.	Blockchain’s decentralized nature provides better confidentiality, integrity and trust. This improves cloud scalability and flexibility for IIoT, SDN enables stable load balancing and resilience, transparent, tamper-proof data sharing without third parties.	High-level conceptual architecture with simulation-only evaluation. Potential overhead from Blockchain consensus in real-time IIoT. SDN centralization risks persist despite distribution.	Security against real-world attacks remains limited, while centralized control architecture creates a potential single point of failure. In addition, privacy and data integrity protection are inadequate due to lack of distributed industrial scale real-world testing.
[11]	ML-based IDS with SNORT and Zeek in SDN evaluates DDOS detection	Low False positives real-time testbed evaluation.	Limited DDOS attack , high processing time	No Blockchain for distributed trust gaps in multi-attack scenarios
[12]	Multi-connection TI protocol (MTIP) for 5G, context-aware path selection	Low cost packets tradeoff reliability duplicates	High duplicate packets in strict reliability	Gaps in Blockchain for secure migration,6g extension
[13]	SDN-based Blockchain VANET security, encrypted flow rules	Removes 95% attacks , improves traffic coordination	High Energy overhead in mobile environments	Latency issues and limited simulation
[14]	Blockchain for SDN flow rule management uses BRS for failure recovery	Robust against controller failure, decentralized rules	High response time in failure	Security evaluation is limited

[15]	SDN-Blockchain for VANET routing encrypted SDN.	Remove 90% attacks and normalize traffic	Mobile overhead is high.	Gap in TI latency, Limited simulation
[16]	Survey of TI architecture, propose edge computing and Artificial intelligence for haptic communications	Comprehensive requirements for 1ms latency identify enablers like SDN and NFV (Network Function Virtualization)	Survey-based no new empirical methodology	Lack of structure for trust in distributed TI, Gaps in 6G integration
[17]	Survey on TI design, mind map of issues proposes SDN/NFV enablers	Comprehensive on latency (1 ms challenges), identifies enablers	Survey-based high-level gaps	No Blockchain for security, real-time testing needed
[18]	A survey on Blockchain IDS, issues, and challenges were explained	Distributed trust enhances reliability	High consensus cost	Gaps in federated learning integration
[19]	Blockchain-based CIDS with Hyperledger Fabric and SNORT IDS	Robust against insider attack, transparent sharing	Consensus overhead, limited to simulated environments	No anomaly detection, Scalability in large networks
[20]	Edge Blockchain as a service (BaaS) with game theoretic reward schemes for flow verification in SDN-IoT network, uses proof–of–work hybrid	Enhances trust between SDN control and data plane, maximizes joint profits of SDN operator and BAAS provider, provides immutable history for anomaly detection.	Moral Hazard issue in information asymmetry, high computational cost for Blockchain consensus in edge environments.	Limited evaluation of security policies, no consideration of scalability in large-scale critical infrastructure sectors.
[21]	Federated DL (Deep Learning) IDS for Blockchain-based Smart Transport System context-aware transformer.	High detection accuracy, privacy preserving.	High false positive anomalies	Gaps in real-time application and no multi-attack evaluation.
[22]	Integration of SDN into Fiber Wireless (FiWi) networks with a tactile internet, dynamic Wavelength and Bandwidth allocation (TI-DWBA) algorithm for resource provisioning.	Provide ultra-low latency (< 1ms) for tactile internet (TI) applications, support coexistence of Human Type Communication (HTC) and Machine Type Communication (MTC), Improve QoS for haptic communications	High computational overhead in estimating bandwidth for TI and non-TI traffic depends on global traffic conditions, which may lead to inefficiencies in dynamic environments.	Lack of real-world testing in heterogeneous 5G/B5G scenarios, no integration with Blockchain for security in distributed control.
[23]	Blockchain-based anomaly detection (BAD) using distributed ledgers for malicious activity logging in SDN	Distributed tamper-proof trusted and private, avoids central failure points and high accuracy in anomalies detection.	High overhead in real time processing limited to SDN specific threats	No evaluation against advanced evasion techniques, scalability in large network not addressed.
[24]	Survey on Blockchain in SDN; focuses on decentralization and security	Enhances confidentiality and integrity.	Survey based no empirical data.	Gaps in performance metrics, real world scalability.
[25]	Survey on Blockchain in SDN, security, and non-security applications	Decentralized authentication improve reliability.	Survey-based, high-level and conceptual	Gaps in real-time IIoT testing, privacy in SDN
[26]	Survey on IDS techniques datasets and evasion methods	Comprehensive taxonomy identifies ML challenges.	Survey-Based no new implementation	Gaps in Blockchain for distributed IDS high false positives in anomaly detection.

[27]	Blockchain based secure energy trading (BEST) for EVs in SDN.	Reduces moral hazards secures energy trading in Tactile IoT	Centralized SDN vulnerabilities, high consensus cost.	No real-time TI latency evaluation, gaps in multi- stakeholder integration.
[28]	Trust-based Blockchain for collaborative intrusion detection networks(CIDNs), combines proof-of-work and proof-of-stake for the consensus mechanism, evaluate IDS peers and IP sources	Enhances security against insider threats, provides transparency and accountability, weighted trust model adjusts sensitivity to behavioral changes	High consensus overhead in dynamic networks; Potential scalability issues in large CIDNs.	No real-time performance evaluation and lacks integration with SDN for broader network control.
[29]	Collaborative IDS using Blockchain with SNORT, Shares signature via Ryu SDN controller	Achieves 96% true positive rate and avoids single points of failure, also provides secure signature sharing	Limited to specific attacks, high overheads in heavy traffic	No anomaly-based detection, scalability in large networks not evaluated.
[30]	Blockchain-based web attack detection system, signature with Multichain	Real-time updates, detects SQL injection and XSS (cross-site scripting)	Limited to signatures, high update overhead	Gaps in anomaly detection and scalability
[31]	Collaborative Blockchain-based signature IDS (CBsigIDS) uses Blockchain for trusted signature database	Robust against insider attacks and improves detection accuracy.	High overhead in distributed updates/	Limited Signature- based anomaly gaps.
[32]	Blockchain-enhanced challenge-based collaborative intrusion detection networks (CIDNs) defeat insider attacks like PMFA (Passive Message Finger Printing attack)	Robust against advanced insider attacks and provides distributed trust	Overhead in Blockchain updates	Gap in IoT scalability

The above analysis reveals a consistent and structurally important gap across three research dimensions. Studies that prioritize security and trust, such as the collaborative IDS framework of Ujjan et al. [29], the Hyperledger Fabric-based CIDS of Gurung et al. [19], and the Blockchain-enhanced challenge-based CIDN of Meng et al. [32] , achieve robust policy enforcement but treat QoS as secondary, leaving latency, jitter, throughput and packet loss largely uncontrolled and unevaluated. Studies that prioritize Tactile Internet QoS, including the comprehensive survey of Promwongsa et al. [16], the multi-connection tactile protocol by Rico et al. [12], and the FiWi-TI resource allocation framework by Ganesan et al. [22], demonstrate measurable latency gain but introduce no decentralized trust mechanism for policy validation, making them vulnerable to the controller-level rule manipulation documented in Mukherjee et al. [14] and Choudhary and Dorle [13, 15]. Hybrid approaches such as Variagade et al. [7] and Rahman et al. [10] attempt partial integration of Blockchain and SDN but evaluate neither comprehensive QoS dimensions nor Tactile Internet traffic profiles, and none of these works formally characterize how Blockchain validation latency is prevented from propagating to time-critical forwarding paths. This propagating problem is not merely a performance concern; it is an architectural one. Any framework that routes Blockchain endorsement through the same control plane thread as forwarding decisions will impose consensus latency on every flow, including those with sub-millisecond QoS requirements. Addressing it requires a scheduling mechanism that explicitly separates the Blockchain validation path from the forwarding critical path and provides a formal guarantee of that separation.

The present work addresses three gaps through a unified SDN-Blockchain-Tactile Internet experimental framework built on Hyperledger Fabric with Raft consensus. It introduces the Blockchain Decoupled QoS scheduling algorithm (BQSA), formally specified and proven in Theorem 1 of Section 3.5. Notably, Vairagade et al. [7] explicitly acknowledge in their own conclusion that their framework "lacks the academic rigor expected in high-impact scholarly venues" due to the absence of formal security proofs, complexity analysis and statistical validation, a gap the present work directly addresses through Theorem 1 and the complexity analysis of Section 3.5.4

III. Methodology

3.1 Proposed Framework

The proposed framework integrates Software-Defined Networking (SDN), a private and permissioned Blockchain network (Hyperledger Fabric), and Tactile Internet (TI) applications into a unified architecture to achieve two main objectives

1. Secure rule validation and accountability.
2. Optimized Quality of Service (QoS) for latency-critical tactile applications.

Fig. 3 illustrates the overall system architecture defining a three-layer architecture for a Tactile Internet-enabled SDN framework integrated with Blockchain.

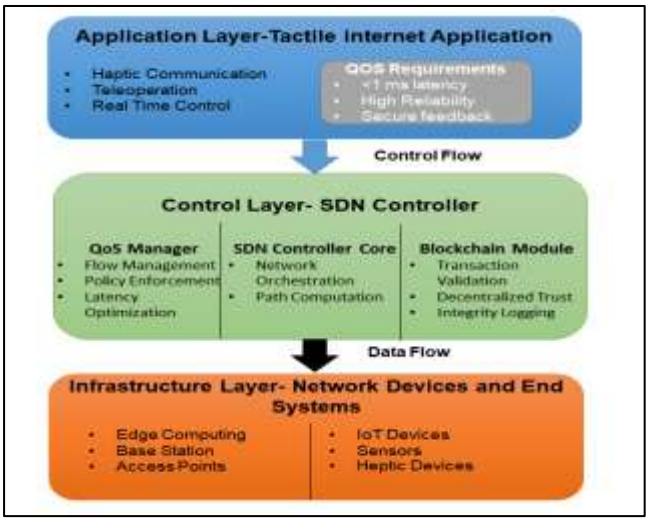

Fig. 3 Layered SDN-Blockchain-Tactile Internet architecture.

3.2 Traffic classification

At the application layer, incoming traffic flows are inspected and classified into two latency classes before any forwarding or validation decision is made. Tactile flows include haptic feedback signals, teleoperation commands and real-time control streams carrying stringent delay requirements that make them incompatible with the additional processing overhead of synchronous Blockchain endorsement. Non Tactile flow including background IoT signaling and bulk data transfers, can accommodate higher latency and therefore follow the standard synchronous validation path. Classification is performed by inspecting three fields in the packet header: the differentiated Service Code Point (DSCP) value, the application identifier extracted from deep packet inspection and QoS priority tags pre-assigned at the network edge. A flow is labelled TACTILE if its DSCP value equals Expedited Forwarding (EF), all other flows are labelled STANDARD. The output of this classification is the latency class C(F) ∈ {TACTILE, STANDARD} is the primary input to the BQSA scheduling decision described in Section 3.5.

The following algorithm formalizes the classification logic.

Procedure 1: Traffic Classification and Priority Assignment (Tactile vs Non-Tactile)

Input: Incoming network traffic packets

Output: Classified traffic with assigned priority levels

1. *For each incoming packet/flow F:*

a) *Extract header fields: {srcIP, dstIP, srcPort, dstPort, protocol}.*

b) *Extract traffic identifiers {DSCP /QoS _+tag, appID}.*

c) *Check the packet against tactile traffic signatures (Haptic, teleoperation, real-time control).*

2. *Classify the traffic type:*

a) *If (appID ∈. { Haptic, teleoperation, real-time control}) or (DSCP= EF) label as tactile traffic. Where DSCP is Differentiated Service Code Point and EF is Expedited Forwarding*

b) *Otherwise, label as Non-tactile traffic.*

3. *Assign priority and QoS handling:*

a) *If Tactile Traffic, set High Priority and map Low Latency Queue* Q_{LL}.

b) *If Non-Tactile Traffic, set Normal Priority and map Best Effort Queue* Q_{BE}

4. *Forwarding for SDN enforcement:*

a) *Tag the flow with the assigned priority label.*

b) Send the classification output to the SDN controller for rule installation and path selection.

This procedure ensures that latency-sensitive tactile traffic is prioritized at the network entry point.

3.3 SDN Rule Configuration

The Ryu SDN controller manages data plane forwarding through the OpenFlow match-action paradigm, dynamically generating flow rules in response to packet-in events and traffic classification outputs received from the module described in section 3.2. Three categories of flow rules are enforced as summarized in Table 2. Tactile traffic identified by a high-priority DSCP tag and confirmed by the classification module is matched by its combined header and QoS tag and forwarded immediately through the dedicated low–latency queue Q_LL, bypassing any blocking control plane operations. Standard non-tactile traffic is matched by default header fields and forwarded through the best-effort queue Q_BE following completion of the Blockchain validation step. Traffic exhibiting anomalous metadata, including unexpected source signatures, malformed headers or flow parameters inconsistent with installed policy, is redirected to the Blockchain endorsement pipeline for peer validation before any forwarding decision is issued. The priority threshold separating normal from anomalous traffic is determined by the policy set Π; any flow rule proposal whose parameters fall outside Π is treated as suspicious regardless of its latency class. This three-tier rule structure ensures that the enforcement of the security policy does not introduce forwarding delay for TACTILE- Class flows, a property that is formally guaranteed by the BQSA algorithm in Section 3.5.

Table 2. Flow Rule Configuration in the Proposed Framework

Traffic Types	Match Field	Action	Priority
Tactile Traffic	Header + QoS tag	Forwarding Via Low Latency Path	High
Non-Tactile Traffic	Header + Default Tag	Forwarding via standard Latency Path	Medium
Suspicious Traffic	Anomalous Metadata	Redirected to Blockchain check	Critical

3.4 Blockchain Integration Layer

In our proposed framework, we have integrated Blockchain in the SDN control layer, operating alongside the Ryu controller. In this architecture, Blockchain does not participate in packet forwarding; instead, it acts as a trusted control plane service that validates and records the flow rule decision before it is enforced in the data plane. A private and permissioned Blockchain in Hyperledger Fabric is deployed on Docker, providing decentralized trust and ensuring no unauthorized flow rules are enforced,

When the SDN controllers propose a rule update, the metadata is broadcast to Hyperledger peers for consensus validation; if rules are legitimate, they are committed to the immutable ledger and are enforced. Otherwise, it is rejected and flagged for inspection. This mechanism prevents rule injection attacks from happening, enhances transparency and guarantees tamper-proof policy enforcement.

Procedure 2: Blockchain-based Flow Rule Validation

Input Flow rule R proposed by Ryu controller.

Output: Validated and Enforced flow rule

1. *Broadcast R to Hyperledger peer nodes*
2. *For each peer Pi::*

a) Verify rule source and signature

b) Check compliance with security policies

3. *Collect votes { ACCEPT, REJECT }*
4. *If Accept ≥ Threshold*

Commit R to Blockchain ledger

Notify Ryu controller to enforce R

Else

Reject R and send alert to the QoS manager.

The above procedure ensures secure and tamper-proof validation of SDN flow rules using the blockchain consensus method.

In our Framework, we have deliberately chosen a private permissioned Blockchain to reduce consensus latency compared to a public Blockchain, making it more suitable for real-time tactile applications. The impact of Raft consensus on control plane latency can be formulated in terms of transaction ordering and replication delays. The total confirmation delay of a control plane transaction under Raft can be expressed as follows:

$D_{raft} = D_{leader} + D_{replication} + D_{commit}$ (1)

Where D_{leader} denotes the time required by the leader to receive and serialize the transaction. $D_{replication}$ represents the time taken the log entry to majority of follower nodes and D_{commit} corresponds to acknowledgment and final commit of the transaction.

For Raft with N ordering nodes, a transaction is committed once the acknowledgments are received from a quorum Q defined as

$$\mathbf{Q} = \left[\frac{\mathbf{N}}{\mathbf{2}}\right] \mathbf{+1} \qquad (2)$$

Thus, the replication delay is bounded by the slowest response among quorum nodes.

$$Dreplication = i \in Qmax(Di)$$

Within the proposed framework, Raft operates exclusively at the SDN control layer. For STANDARD- class flows, *D*raft contributes directly to per-flow validation latency before a forwarding decision is issued, as captured in Equation (1). For TACTILE-class flows, however, the Blockchain endorsement pipeline is invoked asynchronously, the forwarding rule is installed at the data plane switch before endorsement completes and the ledger commit occurs on a non-blocking side channel. This architectural choice means that *D*raft does not enter the forwarding critical path of any TACTILE flow. The mechanism that implements and formally guarantees this decoupling is the Blockchain Decoupled QoS scheduling Algorithm introduced in section 3.5.

3.5 BQSA: Blockchain Decoupled QoS Scheduling Algorithm

Procedures 1 and 2 establish the classification and validation logic independently. BQSA is the scheduling algorithm that unites them into a single control plane decision mechanism, determining for each incoming flow both the forwarding action and the Blockchain logging path, while preserving the QoS guarantee of section 3.3 and the security guarantee mentioned in section 3.4 simultaneously. BQSA differs from conventional SDN QoS scheduling (eg., strict priority queuing, weighted fair queuing, DSCP marking), which reorders already available packets to manage data plane congestion but cannot address delay caused by an unresolved control plane dependency. Prior Blockchain-SDN frameworks (section 2) route consensus validation and forwarding through the same synchronous thread, so downstream scheduling inherits the full consensus delay. BQSA instead decouples these two operations at the points of decision-making and, via Theorem 1, formally guarantees this decoupling holds irrespective of consensus latency, a property no prior QoS scheduling or Blockchain-SDN integration provides.

3.5.1 Notation

Let F denote an incoming flow characterized by the tuple:

$$F = (srcip, dstip, srcport, dstport, porto, DSCP\ app\ ID) \qquad (3)$$

Let C(F) ∈ {TACTILE, STANDARD} be the latency class from Procedure 1. Let V(F) ∈ {ACCEPT, REJECT} be the blockchain endorsement outcome from Procedure 2. Let Q_LL and Q_BE denote the low-latency and best-effort forwarding queues, respectively. Let D(F) denote the forwarding decision issued to the OvS switch. Let T_classify denote the execution time of Procedure 1, a fixed O(1) header field lookup bounded independently of network load. Let T_ryu denote southbound Api round trip latency for OpenFlow OFPT_FLOW_MOD delivery, bounded independently of Blockchain state.

3.5.2 Algorithm

Algorithm 1: BQSA- Blockchain Decoupled QoS Scheduling

Input: Flow F policy set Π, tactile signature set Σ, Hyperledger peer set P = {p_1, p_2, …, p_n}

Output: Forwarding decision D(F), Blockchain log entry ℓ(F)

1. *Classify F via procedure 1 using Σ → assign C(F)*
2. *Generate flow rules proposal R(F) from policy set Π*
3. *If C(f) = TACTILE then*
4. *Set D(F) ← FORWARD(Q_LL)*
5. *Install R(F) at OvS switch via OpenFlow OFPT_FLOW_MOD*
6. *Dispatch R(F) to Hyperledger endorsement Pipeline [asynchronous – non–blocking side–channel]*
7. *On endorsement completion: commit ℓ(F) ← ⟨F, R(F), V(F), timestamp⟩ to ledger*
8. *Else {C(F)= STANDARD}*
9. *Broadcast R(F) to peer set p[synchronous]*
10. *Collect V(p_i) from each $p_i \in P$*
11. *Compute quorum Q = ⌊N/2⌋ + 1 [from Equation (2)]*
12. *if |{p_i : V(p_i) = ACCEPT}| ≥ Q then*

13. Commit R(f) to ledger
14. Set D(F) ← FORWARD(Q_BE)
15. Install R(F) at OvS switch via OpenFlow OFPT_FLOW_MOD
16. Else
17. SET D(F) ← DROP;
18. Raise policy alert to QoS manager
19. End if
20. Commit ℓ(F) ← ⟨F, R(F), V(F), timestamp⟩
21. End if
22. Return D(F), ℓ(F)

3.5.3 Latency Bound Theorem

Theorem1. Under BQSA, the forwarding latency of any TACTILE- Class flow F is bounded by T_Classify + T_Ryu and is independent of Raft consensus latency *DRaft*

Proof: For TACTILE-class flows, Algorithm 1 executes line 4-5 before initiating Blockchain endorsement at line 6. Line 4sets the forwarding decision D(F) ← FORWARD(Q_LL) and line 5 installs rule R(F) at the OvS switch via OpenFlow OFPT_FLOW_MOD, completing in time T_ryu; the controller-to-switch round trip latency depends only on the southbound API communication delay and is bounded independently of any Blockchain state. Line 6 dispatches the endorsement request on a non-Blocking asynchronous side-channel, returning control to the BQSA event loop immediately without awaiting peer response. Data plane forwarding of subsequent packet matching R(F) therefore begins as soon as the OvS switch processes the OFPT_FLOW_MOD message at line 5 without waiting for Raft consensus to complete. The total TACTILE forwarding latency is therefore bounded by T_classify + T_ryu Equation (1), D_raft = D_leader + D_replication + D_commit; none of these three terms appear in the TACTILE critical path between lines 1 and 5 of Algorithm 1.

Corollary 1. For STANDARD-class flows, the total per-flow scheduling latency under BQSA is bounded by:
L_BQSA(STANDARD) ≤ T_classify + D_raft + T_ryu (4)
Where *Draft* is defined by Equations (1) and (2) and the inequality is strict because *Dleader,* the leader serialization component of *Draft,* partially overlaps with T_classify in the controller's processing pipeline, making Equation (4) a conservative upper bound rather than an exact equal.

3.5.4 Complexity Analysis

Time complexity: BQSA processes each incoming flow through two components regardless of path. Flow classification (line 1) performs a fixed field header lookup against a constant-size DSCP and app_ID table: O(1). Rule generation (line 2) traverses the policy set Π in the worst case: O(|Π|). For TACTILE-class flows, the on-critical-path work ends at line 5, with forwarding decision and OvS rule installation, giving an overall critical-path complexity of O(|Π|). The asynchronous endorsement dispatch at line 6 is off the critical path and does not contribute to forwarding latency. For STANDARD- class flows, peer broadcast (line 9), vote collection line 10 and vote counting (line 12) each scale with the number of ordering nodes N: O(N); quorum computation (line 11) and ledger commit (line 13,20) are O(1) under fixed block parameters. The overall per-flow complexity, taking the dominant term across both paths, is:

$$T_{BOSA} = O(|\Pi| + N) \quad (5)$$

Since both |Π| and N are fixed at deployment — N = 3 in the experimental configuration — T_BQSA is effectively constant in steady-state operation.

Space complexity: The controller maintains three runtime structures: a flow classification cache of size O(n_flows) for active concurrent flows, a peer response buffer of size O(N) per active STANDARD path validation, and a pending endorsement queue of size O(n_pending) bounded by the Hyperledger batch size parameter. Total space complexity is

$$S_{BOSA} = O(n_{flows} + N + n_{pending}) \quad (6)$$

In Practice, with N fixed at 3 and n_pending bounded by batch size, the dominant cost is the flow cache O(n_flow).

3.6 Workflow Summary

The complete operation of the proposed framework proceeds as follows. Each incoming flow F is inspected by the traffic classification module, which assigns latency class C(F) based on DSCP value and application identifier. The Ryu SDN controller generates a corresponding flow rule proposal R(F) drawn from the installed policy set Π. BQSA then determines the forwarding and validation path. TACTILE-class flows receive an immediate forwarding decision with asynchronous Blockchain logging, preserving their QoS guarantee; STANDARD-class flows undergo synchronous peer endorsement via Hyperledger Fabric before

forwarding is authorized, preserving the security guarantee. In both paths, every flow decision is committed to the immutable ledger as a tamper-proof audit record. The QoS monitoring module runs continuously alongside this pipeline, observing latency, jitter, and throughput at the OvS switch level and triggering dynamic rule adjustment via Ryu REST API when monitored values exceed configured thresholds, adjustments that re-enter the BQSA pipeline and are themselves subject to Blockchain validation. This closed-loop design ensures that neither security enforcement nor QoS responsiveness is sacrificed at the expense of the other. Fig 4 illustrates this complete workflow.

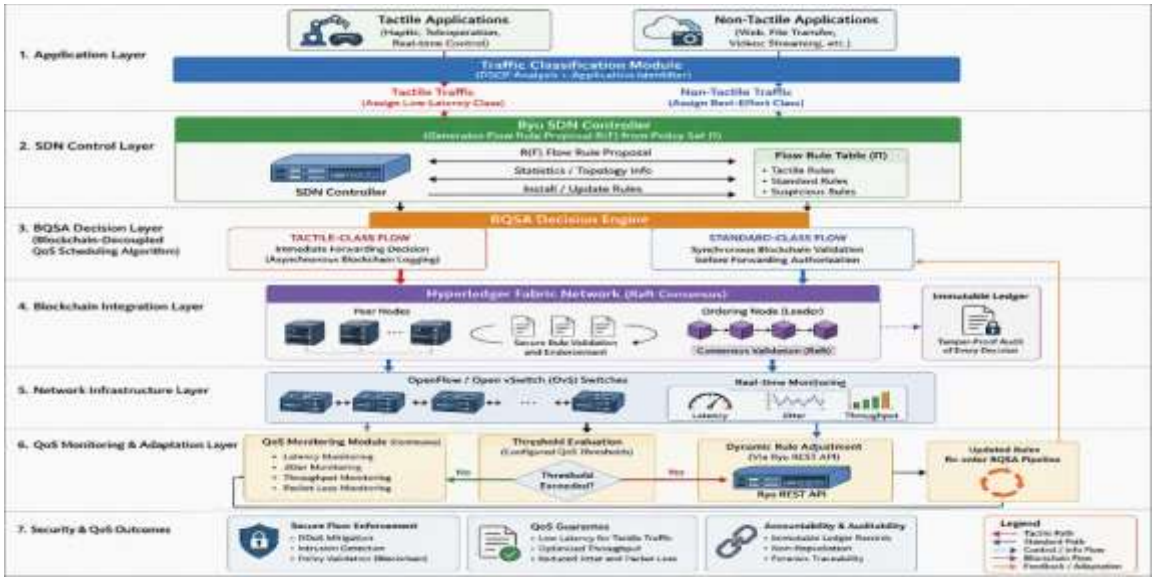

Fig. 4 Workflow of traffic classification, Blockchain-SDN rule enforcement and QoS monitoring.

IV. Experimental Evaluation and Results

This section presents an empirical assessment of our framework that combines Software-Defined Networking (SDN), Hyperledger Fabric (HLF) Blockchain, and Tactile Internet (TI) modules. Three configurations are evaluated.

1. C1-Traditional SDN (No Blockchain)
2. C2-SDN + Blockchain (Hyperledger Fabric only but without Tactile Internet scheduling)
3. C3- the proposed framework incorporating BQSA..

Table 3 presents the consolidated QoS results across all three configurations, and Table 4 provides a contextual comparison with representative values from related SDN-Blockchain literature and individual metric analysis follows in sections 4.2- 4.5

4.1Simulation and Test Setup

Environment: Experiments were conducted on a Windows 11 host using the official Mininet VM Version 2.3.0 for data plane emulation and WSL (Windows Subsystem for Linux) Ubuntu 20.04 for control and ledger plane services. Docker Engine (24x) hosted Hyperledger Fabric components and chain code (Go 1.20). **Fig. 6** shows the methodological flow of the proposed Blockchain- SDN- Tactile Internet Framework. The Ryu SDN controller (Python 3.8) ran with Mininet. Custom TI modules written in Go were deployed via gRPC. The BQSA scheduling module was deployed as a Ryu application, Intercepting a; packet- in- events and applying the classification and validation logic of Algorithm 1 before forwarding decisions were issued. All performance metrics were averaged over five experimental runs to ensure consistency and reduce stochastic variation**. Fig.5** shows the experimental testbed topology, and Fig. 6 illustrates the methodological flow of the framework.

Network Topology: A three-tier fat tree-like topology (k = 4) was emulated in Mininet with 16 hosts and 8 Open vSwitches. Link bandwidth was set to 100 Mbps with a 1ms base delay per hop, and tc (Linux traffic control) was configured. The SDN controller Ryu (Python 3.8) is running in WSL. For fault tolerance, the controller maintained a standby state replication

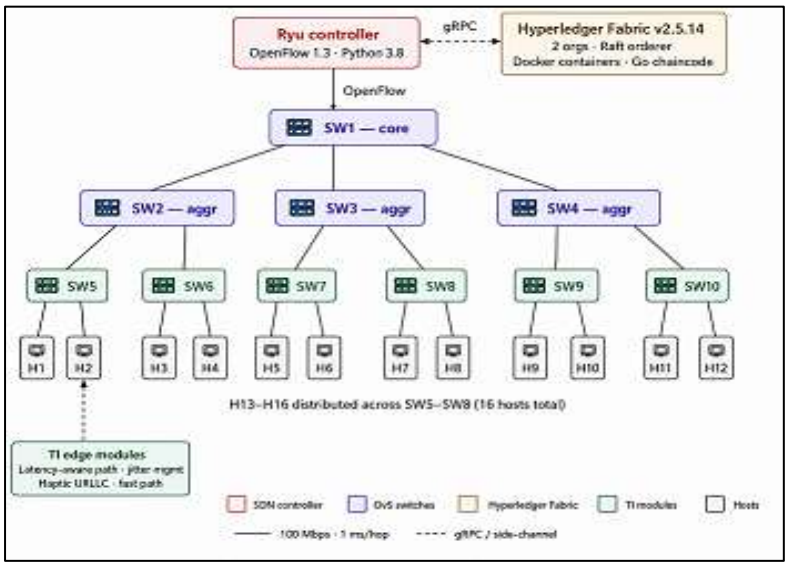

Fig. 5 Experimental testbed topology integrating Mininet, Hyperledger and TI applications.

Blockchain Configuration: Hyperledger Fabric v2.5.14 was instantiated with a Raft ordering service, two organizations, and Peer counts of {4, 8, 12, 16} in scalability experiments. A dedicated channel (policy-audit) recorded rule proposals, validation outcomes and anomaly attack events. Chaincode executed rule validation and audit function under endorsement policy AND("Org1.peer" , "Org2.peer"). Default block parameters: batch timeout 1.0 s: batch size (50, 800) tx transactions depending on test scenario.
Tactile Internet (TI) Modules: TI logic consists of latency-aware path selection, jitter-aware queue management at edge, Packet pacing for haptic URLLC (Ultra Reliable Low-Latency Communication) class flows and a fast path for control plane critical messages. TACTILE-class events were decoupled to the BQSA side-channel to avoid blocking ledger I/O, consistent with the asynchronous dispatch path of Algorithm 1.**Traffic Mix and Workloads**: Iperf3 generated long lived TCP bulk flow. D-ITG (Distributed Internet Traffic Generator produced VoIP-like CBR Flows (Constant Bit Rate) and a synthetic haptic stream (small packets at 1 kHz) emulated TI demands. Background Poisson arrivals stressed the queues to induce jitter under realistic mixed-load conditions.

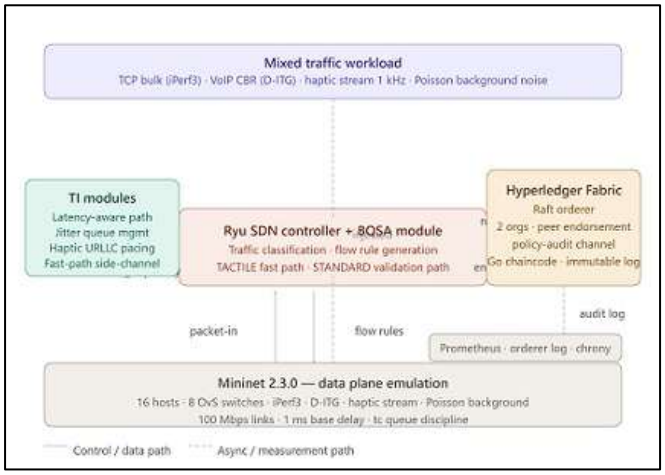

Fig. 6: Methodology flow of the proposed Blockchain–SDN–Tactile Internet framework

Measurement Tooling: QoS metrics (Latency, jitter, throughput, packet loss) were captured via ping, iPerf3logs, D-ITG receiver, and Mininet host counters. Blockchain metrics (confirmation time, throughput) were collected from Hyperledger Fabric Prometheus export and order log. Time synchronization was provided by chrony in WSL. The host machine was equipped with an Intel i7 12th–generation processor and 16 GB RAM; Docker containers were allocated a default resource limit.
Compared configuration.
- **C1 - Traditional SDN:** Ryu controller, no ledger , reactive flow installation
- **C2 – SDN + Blockchain**: (Ryu + HLF) with policy logging and validation in the synchronous control plane

loop, without TI modules.

• **C3 – Proposed (SDN + Blockchain +TI):** Ryu + HLF + TI modules with BQSA latency-aware scheduling and asynchronous side channel for TACTILE critical event.

These three configurations constitute a controlled ablation: C1 isolates baseline SDN performance, C2 isolates the overhead of synchronous Blockchain validation alone, C3 isolates the additional effects of BQSA decoupling, allowing the incremental contribution of each architectural layer to be measured directly.

Table 3 summarizes the measured QoS outcomes across all three configurations. Table 4 provides a contextual comparison against representative values reported in related literature. Because no prior work provides a full SDN Blockchain-Tactile Internet integration, these values are drawn from different testbed configurations and serve as indicative benchmarks rather than equivalent system comparisons. A paired t-test (n=5, df = 4) confirms that C3 significantly outperforms both C1 and C2 for jitter, throughput, and packet loss (all $p<0.01$, $|d|= 3.2$-8.3, indicating large to very large effect sizes). For latency, C3 significantly outperforms C2 ($p<0.001$, $|d| = 7.69$) and outperforms C1 at the conventional 0.05 threshold (P=0.023), though this single comparison does not survive a conservative Bonferroni correction for the eight comparisons tested ($\alpha = 0.00625$). The remaining seven of eight comparisons remain significant even under this stricter threshold.

Table 3 Consolidated QoS results across three experimental configurations

Metric	C1: SDN only	C2: SDN + Blockchain	C3: proposed (BQSA)
Latency (ms)	18.0 ± 1.25	22.0 ±0.72	16.0 ±0.37
Jitter (ms)	4.5 ± 0.31	5.2 ±0.37	2.8 ±0.11
Throughput (Mbps)	92.0 ± 1.45	88.0 ±1.23	96.0 ±0.59
Packet loss (%)	1.8 ±0.07	2.2 ±0.16	1.0 ±0.16

"Values reported as means ± standard deviation across five experimental runs"

Table 4 Contextual QoS comparison with representative SDN-Blockchain literature

Metric	C3 (this work)	Vairagade et al. [7]*	Rahman et al. [10]*
Latency (ms)	16.0	~80–82	~20–35
Jitter (ms)	2.8	Not reported	~4–8
Throughput (Mbps)	96.0	~85–92	~80–90
Packet loss (%)	1.0	~3.2–8.8 (derived from reported PDR)	~2.0–3.5
Blockchain type	Hyperledger Fabric/Raft	Hybrid PoS(intra-cluster)/PoW (inter-controller), private Ethereum	Ethereum
Tactile Internet	Yes	No	No
Blockchain Validation Time (ms)	9.6	135 (hybrid chain)	Not reported

* *Values extracted from original publications under different testbed configurations, topology scale and workload parameters. Direct numerical comparison is not valid: range reflect values reported across multiple experimental scenarios in the cited work.*

QoS Metrics: Fig. 7 presents a consolidated Quality of Service (QoS) summary showing a comparison of quality aspects such as latency, Jitter, throughput and packet loss across three evaluated configurations relative to the literature reference range in Table 4. The results show that while the SDN-Blockchain configuration C2 experiences moderate degradation due to ledger interactions, the proposed SDN-Blockchain-Tactile Internet configuration C3 consistently meets and outperforms literature benchmarks across many quality-of-service parameters. This outcome directly reflects the BOSA forwarding isolation property; Blockchain

Validation overhead is removed from the TACTILE forwarding critical path, allowing C3 to outperform C1 despite the additional security layer.

Unlike the hybrid Blockchain-SDN architecture of Vairagade et al. [7] and Mozumder et al. [8], which report improved security at the cost of measurable QoS degradation, and unlike the DistB-SDcloud architecture by Rahman et al.[10], which does not address QoS optimization for latency-sensitive tactile traffic, the

proposed C3 framework demonstrates that Blockchain- based trust and Tactile Internet QoS are architecturally compatible rather than mutually exclusive.

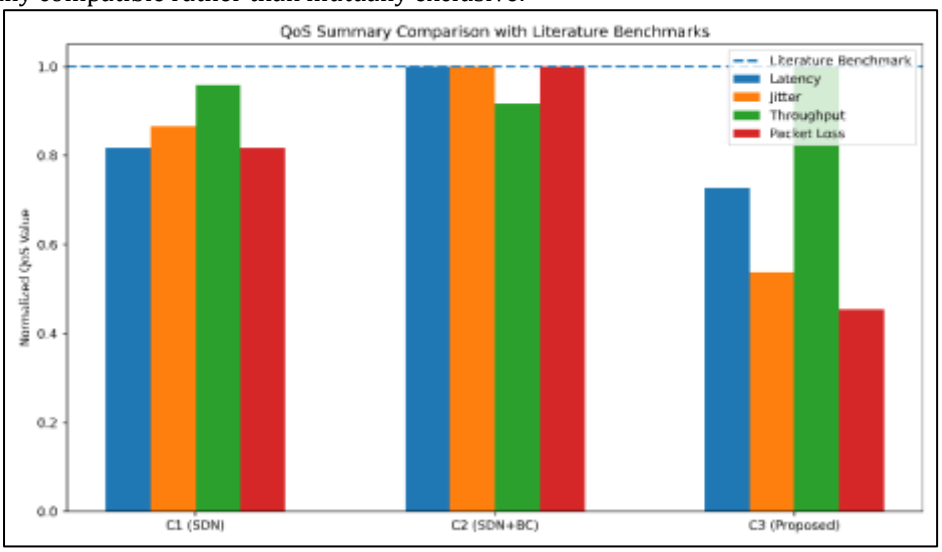

Fig. 7 – QoS comparison across configurations (Latency, Jitter, Throughput, Packet Loss).

Compared to C1, C2 incurs modest overhead due to Hyperledger interaction (higher latency and jitter). The proposed C3 regains and surpasses QoS via TI optimizations: Lowest latency and jitter, highest throughput, and the least packet loss under identical offered load

.

4.2 Latency Analysis

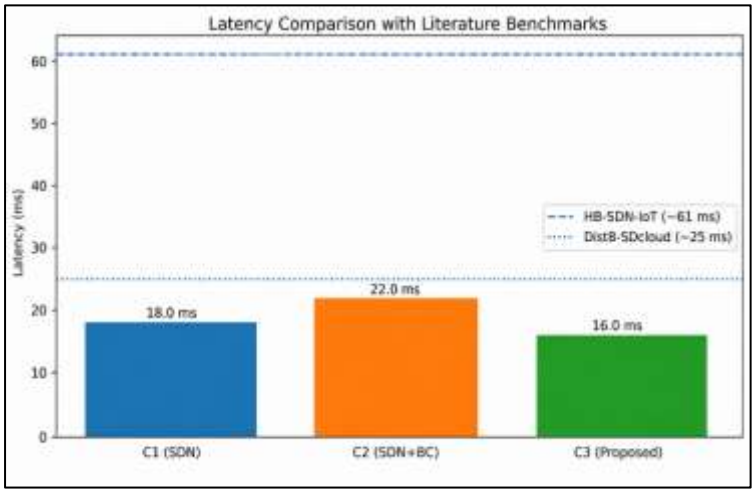

Fig. 8 – Latency comparison Chart

Fig. 8 illustrates the end-to-end latency comparison across the three configurations, together with representative reference values from related SDN-Blockchain literature [7, 8]. Compared to traditional SDN (C1), the SDN-Blockchain configuration (C2) incurs higher latency due to the synchronous control-plane validation and Raft consensus delay added to every flow decision. The proposed C3 configuration achieves the lowest latency of 16.0 ms — 2.0 ms below C1 and 6.0 ms below C2. This improvement results from BQSA's asynchronous dispatch of Blockchain endorsement for TACTILE-class flows, which remove *Draft* from the forwarding critical path as established in Theorem 1. The C3 latency remains below the reference ranges reported in prior literature, confirming the framework's suitability for Tactile Internet and low-latency applications.

Latency is computed as end-to-end delay between packet transmission and reception

$L = t_{\text{recv}} - t_{\text{send}}$ (7)

The average latency over N packets is given by:

$L_{\text{avg}} = \frac{1}{N}\sum_{i=1}^{N} L_i$ (8)

4.3 Jitter Analysis

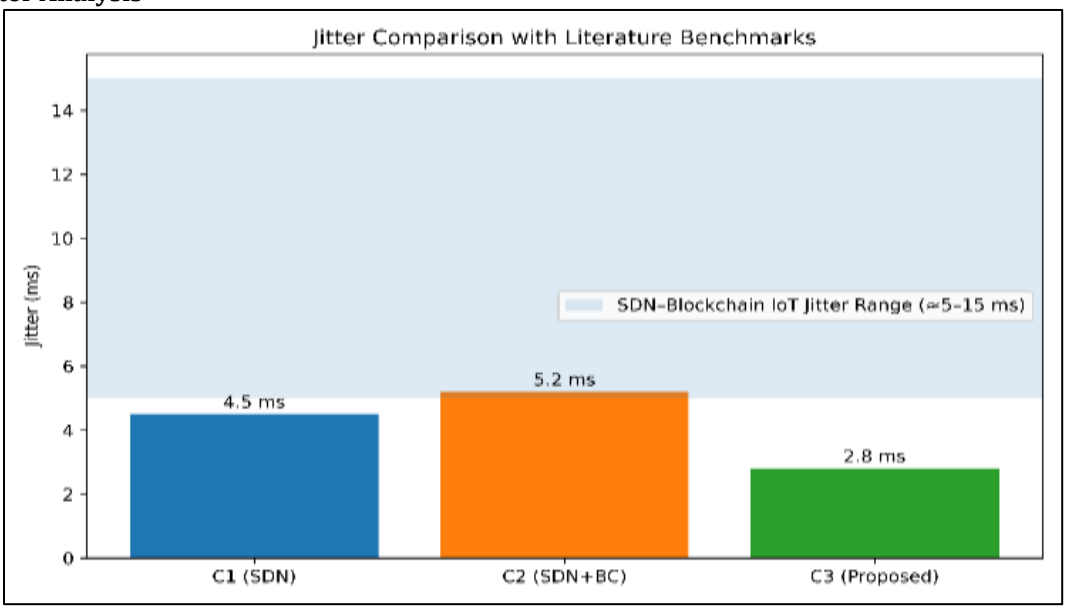

Fig. 9 –Jitter comparison chart

The jitter performance is presented through **Fig. 9**. Our results indicates that C3 shows the lowest delay variation at 2.8 ms, a 37.8% reduction relative to C1 4.5ms and a 46.2% reduction relative to C2 (5.2 ms, reflecting more stable packet delivery compared to both C1 and C2 while Blockchain integration in C2 introduced additional jitter due to asynchronous validation and ordering, the proposed work mitigates this effect by isolating times sensitive traffic from Blockchain processing through BOSA's edge level management and priority scheduling. The jitter value of C3 remains below the jitter range reported in existing SDN-Blockchain-IIoT Rahman et al. [10] systems, highlighting the effectiveness of the proposed design for real-time communication.

Jitter is defined as the variation in delay between consecutive packets:

$J_i = |L_i - L_{i-1}|$ (9)

The mean jitter is computed as

$J_{\text{avg}} = \frac{1}{N-1}\sum_{i=2}^{N}|L_i - L_{i-1}|$ (10) Where $L_i$denotes the end-to-end latency of the i-th packet, and L_{i-1}denotes the latency of the immediately preceding packet

4.4 Throughput Analysis

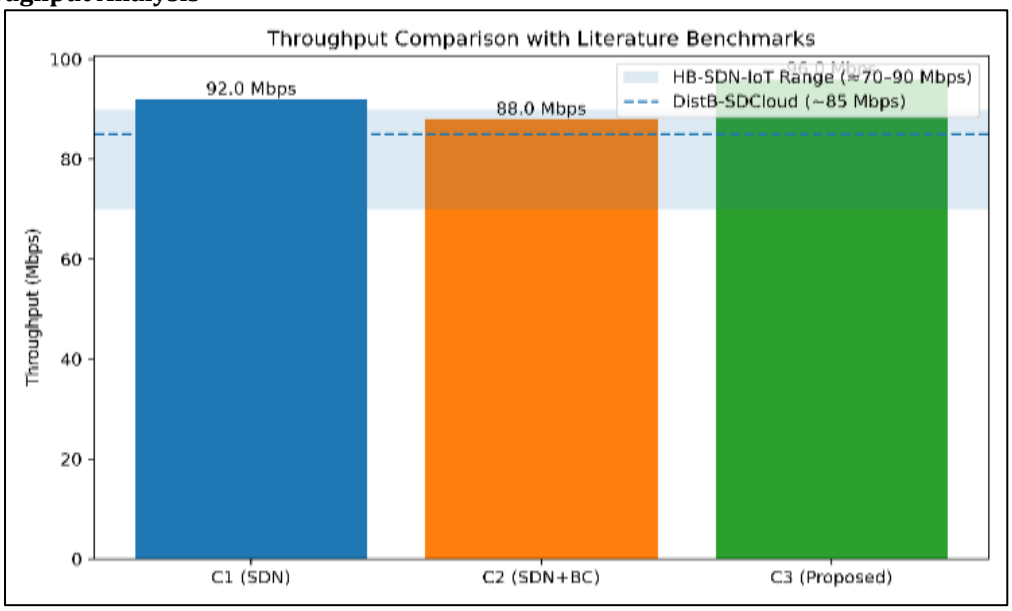

Fig. 10 –Throughput comparison chart

Fig. 10 compares the throughput performance of three configurations against representative literature benchmarks. Despite the Blockchain-based security mechanisms, the proposed C3 configuration achieves the highest throughput at 96.0 Mbps, 4.3% above C1 (92.0 Mbps) and 9.1% above C2 88.0 Mbps. This gain results from efficient flow rule caching, reduced retransmission and optimized coordination between the SDN controller and the Blockchain layer. Unlike C2, where Blockchain interaction directly impacts forwarding efficiency, the proposed Algorithm ensures that Blockchain validation does not obstruct data plane operations. As a result, C3 matches and exceeds throughput levels reported in prior SDN-Blockchain studies (HB-SDN-IoT) such as the secure IoT routing framework by Vaiagade et al. [7] and Blockchain-SDN architecture for Industrial IoT (DistB-SDcloud) proposed by Rahman et al. [10]

Throughput is calculated as the rate of successful data delivery over time:

$$T = \frac{D_{\text{recv}}}{t_{\text{end}} - t_{\text{start}}} \quad (11)$$

In bits per second:

$$T = \frac{8 \times D_{\text{recv}}}{t_{\text{end}} - t_{\text{start}}} \quad (12)$$

Where D_{recv} denotes the total volume of successfully received data during the measurement interval.

4.5 Packet Loss Analysis

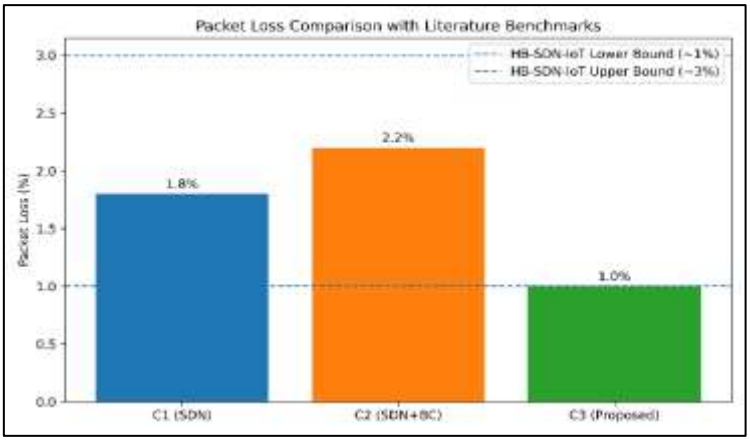

Fig. 11 –Packet loss comparison chart.

Fig. 11 presents the packet loss across different configurations along with reference bounds derived from related work. Our proposed architecture records the lowest packet loss at 1.0% a 44.4% reduction relative to C1 1.8% and a 54.5% reduction relative to C2 2.2%. The higher packet loss of C2 is a consequence of control plane congestion induced by synchronous Blockchain endorsement requests competing with forwarding decisions in the same Ryu event loop.C3 addresses this through BQSA's proactive flow management and the decoupling of Blockchain events to a non-blocking side-channel, improving control plane responsiveness and reducing drop events. The C3 packet loss is on par with or better than reference values reported in Vairagade et al. [7].

Packet loss percentage is computed as:

$$PL(\%) = \frac{P_{sent} - P_{received}}{P_{sent}} \times 100 \quad (13)$$

The packet delivery ratio (PDR) is defined as

$$PDR = \frac{P_{\text{recv}}}{P_{\text{sent}}} \quad (14)$$

Where P_{sent} are packets sent and $P_{received}$ are packets received, the packet delivery ratio (PDR) provides a complementary reliability measure.

Taken together, the metrics wise result in **Fig. 8 to Fig.11** demonstrate that the proposed SDN-Blockchain-Tactile architecture not only mitigates the performance overhead that is typically associated with Blockchain integration but also improves overall quality of service as compared to SDN and existing Blockchain solutions. The consolidated QoS comparison in **Fig. 7** confirms these findings by providing a system-level view relative to the literature benchmark. The equations (3-10) are the exact formulas used to compute the plotted values for quality of Service and equations (1-3) for Raft consensus.

4.6 Flow Rule Integrity and Blockchain Audit Validation

The primary security function of the Blockchain layer in the proposed framework is the prevention of unauthorized flow rule modification and the provision of a tamper-proof audit trail for all control plane decisions. This section evaluates the effectiveness of the BQSA Blockchain validation path Algorithm 1in

detecting and rejecting illegitimate rule proposals before they reach data plane.
Policy rule validation is implemented using Hyperledger Fabric chaincode to make sure that all flow rules are verified before installation in the network. **Fig.12** shows the validation latency comparison where the SDN + Blockchain configuration (C2) records an average latency of 45 ms while the proposed SDN+Blockchain+TI framework (C3) reduced it to 32 ms, a reduction of 28.9%. This improvement is achieved through BQSA's selective batching of non-critical audit events into larger blocks and the side-channel decoupling of TACTILE-class endorsement, which reduces contention on the ordered pipeline. All validated rules are immutably stored on the Hyperledger Fabric, enabling full traceability and auditability of network operations

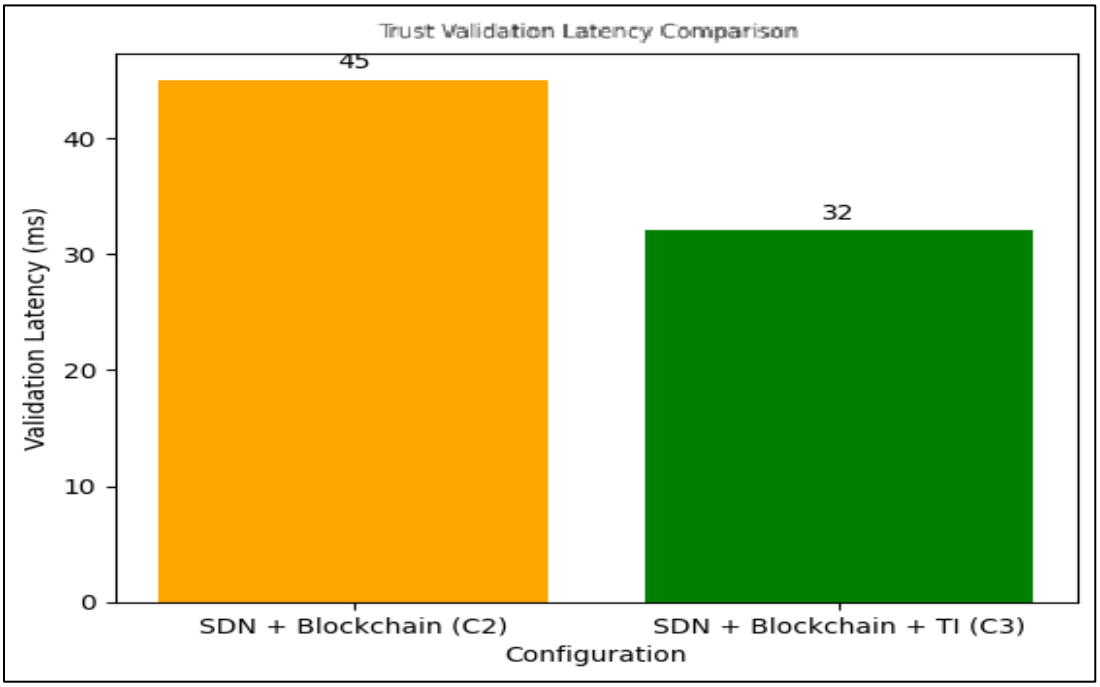

Fig.12-On-Chain policy rule trust validation latency comparison.

To evaluate security, controlled attack scenarios including SYN flood, spoofed flow-rules injection, and unauthorized rule modification were introduced. In both C2 and C3, malicious rule requests were successfully detected and rejected during validation, whereas in traditional SDN (C1) such actions were neither validated nor logged. These three events represent the canonical classes of flow-rule integrity violation: injection under load, injection via identity spoofing and direct tampering with an installed rule. They cover the integrity threat surface relevant to this evaluation's stated scope; broader multi-attack detections are addressed by the dedicated security enforcement layer in our ongoing work.

Table 5. Security Evaluation of Blockchain-based Rule Validation

Attack Type	Injected event	Detected	Detection Rate (%)	Avg Response Time (ms)
SYN- induced rule injection	100	96	96%	13
Spoofed flow rules injection	50	48	96%	11
Unauthorized rule modification	40	39	97.5%	12

The results in **Table 3** confirm that Blockchain-based validation effectively prevents unauthorized rule modifications while maintaining acceptable response latency for control plane operations. It is important to note that these results characterize only flow rule integrity enforcement.

4.7 Scalability/Overhead Analysis

Transaction Confirmation Time vs Peers count. Fig.13 reports the median end-to-end confirmation time as peers increase from 4 to 16

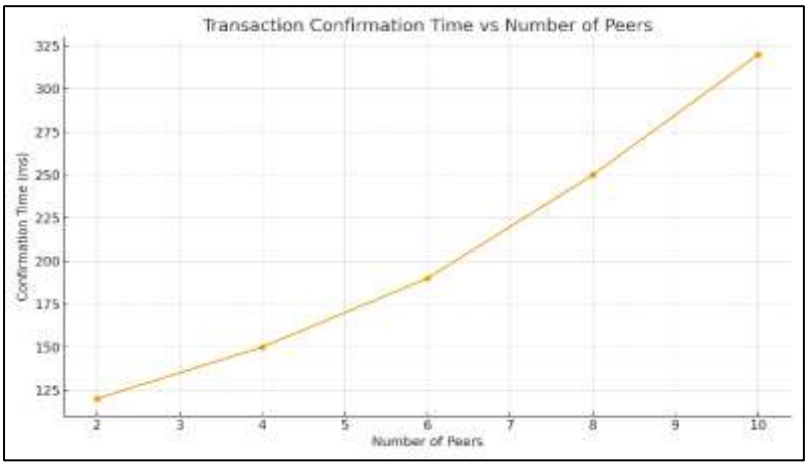

Fig.13 – Transaction confirmation time vs number of peers

The proposed C3 remains consistently faster across the entire range: 780 ms at 4 peers vs 950 ms for C2; 1350 ms at 16 peers vs 1600 ms for C2 due This advantage is attributed to two mechanisms: small-block batching for TI-Critical events, which reduces per-transaction queuing delay and Controller-side caching of read sets, which reduces MVCC (Multi-version concurrency control) conflicts during concurrent endorsement.

Blockchain Throughput Vs Block size: Fig.14 shows throughput growth with block size; C3 attains higher peak throughout approximately~350 tps (transactions per second) at block size 800 relative to C2 ~305 tps. This gain is attributed to BQSA's strategy of coalescing non-critical audit transactions into larger blocks while fast- pathing TI-critical operations through the asynchronous side channel, reducing per-Transaction consensus overhead for the bulk of ledger operations.

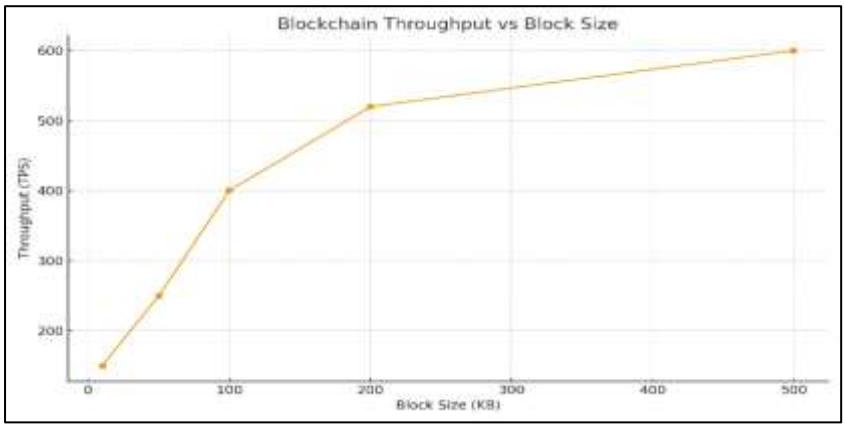

Fig.14- Blockchain throughput vs block size

4.8 Comparison with Traditional Network Models

Relative to traditional SDN (C1), both ledger-enabled configurations (C2, C3) provide provable accountability and tamper-evident policy enforcement. C2 achieves this at the cost of QoS degradation: latency increases by 22%, jitter by 15.6%, throughput falls by 4.3% and packet loss increases by 22.2% relative to C1. C3 eliminates this tradeoff: all four QoS metrics match or improve upon C1 while full Blockchain capability is retained. This outcome is a direct consequence of the BOSA forwarding isolation properties established in Theorem 1. Because Blockchain endorsement is dispatched asynchronously for TACTILE-class flows, the Raft consensus latency does not appear in the forwarding critical path, allowing C3 to match C1 latency despite the additional Blockchain layer. The results demonstrate that architectural separation of security enforcement from QoS-critical forwarding as formalized by BQSA is an effective and measurable strategy for building networks that are simultaneously secure, auditable, and QoS responsive.

V. Discussion, Insight and Limitations

The experimental evaluation demonstrates that the integration of Blockchain and Tactile Internet mechanisms within an SDN environment leads to a balanced improvement in Quality of Service while preserving secure and auditable network control. Compared with traditional SDN (C1) and SDN-Blockchain (C2) the proposed framework (C3) consistently achieves lower latency, reduced jitter, higher throughput and lower packet loss under the same conditions. Even in comparison with representative literature benchmarks such as hybrid Blockchain-SDN architectures (HB-SDN-IoT) by Vairagade et al. [7], the SmartSecChain-SDN model by Mozumder et al, [8] and DistB-SDcloud, the Blockchain-SDN architecture for Industrial IoT proposed by Rahman et al. [10], the proposed framework performs at par with or better than these representative works, as detailed in Table 4, although direct one-to-one comparison is not strictly feasible due to architectural differences.

The mechanism behind this compatibility is the BQSA algorithm. C2's QoS degradation a 22% latency increase and 15.6% jitter increase relative to C1 arises directly from routing Blockchain endorsement through the same synchronous control plane threads as forwarding decisions, so that *Draft* adds to every flow's forwarding delay. BQSA resolves this by implementing the asynchronous dispatch path of Algorithm 1 for TACTILE-class flows; the forwarding rule is installed at the OvS switch before endorsement begins, and the consensus operation completes on a non-blocking side channel. Theorem 1 formally guarantees that *Draft* does not contribute to TACTILE forwarding latency under this design. The results in Table 3 are empirical confirmation of the theorem; C3 achieves 16.0 ms average latency lower than both C1 (18.0 ms) and C2 (22.0 ms) despite operating the full Blockchain audit pipeline concurrently. The latency reduction below C1 is attributed to BQSA's proactive flow classification and controller-side caching, which together reduce the number of packet-in events that reach the Ryu event loop for flows already classified in the cache. As illustrated in **Fig.15,** the evolution from traditional SDN to Blockchain-SDN improves security at the cost of QoS overheads, whereas hybrid architecture effectively mitigates this trade-off by restoring QoS while maintaining decentralized trust and secure rule validation.

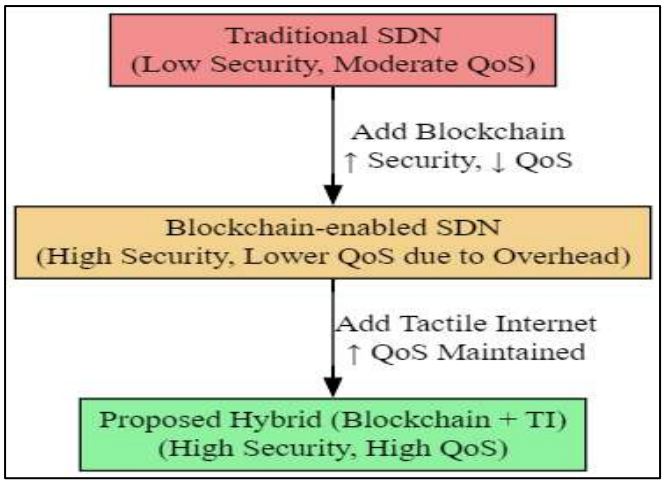

Fig.15: Evolution of security-QoS trade-offs in SDN, Blockchain-enabled SDN, and proposed Hybrid framework

Although our proposed framework demonstrates improved latency compared with the baseline configuration, the observed latency values are approximately 16-22 ms, though remains higher than the strict sub-millisecond requirements typically associated with Tactile internet applications. This discrepancy arises primarily due to the emulation-based experimental setup using Mininet and the additional processing overhead introduced by containerized Blockchain components. The current implementation should be considered as a proof-of-concept system that validates the feasibility of integrating SDN, Blockchain and Tactile Internet principles within a unified framework. In a practical scenario, the use of hardware switches, high-speed edge computing and optimized networking stacks can substantially decrease latency and help meet the demanding requirements of Tactile Internet systems.

Further investigation shows that the C2 configuration results in an unexpected delay due to ordering and ledger validation in Hyperledger Fabric. This overhead manifests as increased latency and jitter, which are to be expected in a permissioned Blockchain system. However, the proposed C3 architecture compensates these effects through intelligent scheduling and fast-path forwarding for tactile traffic. Instead of allowing

Blockchain processes to interfere with real–time packet delivery, the system intentionally decouples validation from forwarding using Algorithm 1 BQSA; this helps in maintaining service continuity for delay-sensitive applications.

Three practical observations emerge from the QoS behavior: Tactile traffic prioritization significantly stabilizes latency and jitter under mixed workloads; fast-path control signaling prevents Blockchain commit delays from propagating to the data plane and selective batching of non-critical audit logs improves ledger efficiency without affecting real-time flows.

Another important insight emerges from the separation of control plane and data plane operations. By confining Blockchain interaction mainly to flow rules validation and audit logging, the framework avoids unnecessary delays in packet transmission. This design is particularly relevant for URLLC (ultra-reliable low-latency communication) like scenarios where deterministic latency is required. Moreover, controller-side caching and reduced endorsement fan-out help minimize MVCC conflicts, leading to smoother transaction validation and improved throughput stability during high traffic loads. In practical deployment, such isolation can prevent security mechanisms from becoming performance bottlenecks.

From a scalability standpoint, the results indicate that increasing the number of Blockchain peers gradually raises transaction confirmation time due to Raft consensus coordination. While the increase remains within acceptable bounds for moderate network sizes, large peer networks may experience noticeable commit delays and resource overhead. This highlights the sensitivity of Blockchain-enabled SDN systems to configuration parameters such as block size, batch timeout and endorsement policy: smaller block size favors latency-sensitive operations, larger block enhances audit throughput but may increase confirmation delays and multichannel or sharded ledger designs could improve scalability in larger deployments.

It is also important to note that the comparison with other literature is necessarily contextual rather than strictly equivalent, because there are very few studies that evaluate a fully integrated SDN-Blockchain-Tactile Internet framework within a feasible testbed. Overall, consistent with the conceptual trend depicted in **Fig.15,** the results indicate that the proposed architecture offers a realistic trade-off between security assurance and Quality of Service rather than unrealistic performance gain in all fronts. Blockchain provides transparency, traceability, and tamper-proof rule enforcement, but comes at a cost of processing overhead. The primary significance of our work is demonstrating that such overhead can be reduced by architectural co-design, traffic prioritization, and separation of control and data plane. Future research work should focus on the development of lightweight consensus mechanisms, edge-assisted Blockchain nodes and large-scale hardware testbeds to validate performance under real-world network conditions and thereby further improve scalability for next-generation tactile and mission-critical applications.

5.1 Practical Implications and Real–World Relevance

The findings of the study suggest that the proposed SDN-Blockchain-Tactile Internet framework is suitable for environments where both trust and low-latency communication are important. This is achieved through a combination of decentralized rule validation and latency-sensitive traffic management. In practical terms, this type of architecture may apply to applications that depend on predictable latency and secure control mechanisms, such as industrial automation, smart grid coordination, teleoperation systems and intelligent transportation networks.

5.2 Limitations

Four limitations of the current work are explicitly acknowledged:

1. The experiments were conducted in a Mininet-based emulation environment. Although this enables controlled and reproducible evaluation, it does not fully capture hardware-specific forwarding behavior or propagation characteristics observed in large-scale physical deployments.
2. The topology scale is limited to 16 hosts and 8 switches; BQSA's $O(|\Pi| + N)$ complexity suggests favorable scaling; further work will investigate scalability over larger topologies and multi-controller deployment
3. The Sig_DB flow rule signature set used in integrity validation is static and currently relies on a predefined signature database for known attack patterns.
4. The weighting parameters were selected based on preliminary experimentation within the evaluated workload. Automated parameter optimization and sensitivity analysis across diverse traffic conditions remain subject for future investigation.
5. Although paired t-tests confirm significant improvement across nearly all metric comparisons (Section 4.1), the sample size (n=5 runs) remains small; future work will increase repetitions to strengthen statistical power, particularly for latency comparison between C3 and C1, which is the one result sensitive to conservative multiple comparison correction.

VI. Conclusion

Our work presented a unified framework incorporating Software Defined Networking, Hyperledger Fabric-driven Blockchain and Tactile Internet principles to overcome the dual challenges of network security and Quality of Service through BQSA, a scheduling mechanism that decouples Blockchain validation from data forwarding. We experimentally tested three configurations and found that adding Blockchain to SDN introduces measurable overhead, but the BQSA-driven framework recovers this overhead and outperforms both traditional SDN and the SDN-Blockchain baseline in latency, jitter, throughput and packet loss, a behavior formally guaranteed by Theorem 1. Security-wise, consensus based on Raft provided tamper-proof flow rule validation and immutable audit logging, preventing unauthorized rule injection that left no trace in the baseline setup.

It is worth noting that the measured latency of approximately 16 ms fall short of strict sub-millisecond Tactile Internet requirement standards and the current implementation is a Proof-of-concept in emulation conditions rather than a production-ready deployment. Future work should focus on lightweight consensus mechanisms, physical hardware testbeds with 5G backhaul and integration of a dedicated attack-aware security layer for multi-pattern Intrusion detection. Overall, the finding indicates that decoupling Blockchain validation from data plane forwarding as formalized by BQSA is a viable architectural strategy for building networks that are both secure and responsive.

References

[1] Nokia, "Global network traffic report: Understanding the growing impact of advancing technologies on future networks," Nokia Bell Labs Consulting, Espoo, Finland, Rep. CID:213660, 2024. [Online]. Available: https://www.nokia.com/asset/213660/

[2] Sofrecom, "Internet Traffic Growth: Trends and Forecasts," Sofrecom News & Insights, Paris, France, Mar. 2025. [Online]. Available: https://www.sofrecom.com/en/news-insights/internet-traffic-growth-trends-and-forecasts.html

[3] Grand View Research, "Network Traffic Analysis Market Size, Share & Trends Analysis Report by Component, Deployment, Organization Size, Vertical, By Region, And Segment Forecasts, 2025 - 2030," Rep. GVR-4-68038-305-8, San Francisco, CA, USA, Dec. 2024. [Online]. Available: https://www.grandviewresearch.com/industry-analysis/network-traffic-analysis-market-report

[4] M. Rahouti, K. Xiong, and Y. Xin, "Secure software-defined networking communication systems for smart cities: Current status, challenges, and trends," *IEEE Access*, vol. 9, pp. 12083–12113, 2021, doi: 10.1109/ACCESS.2020.3047996.

[5] S. Rathore, P. Sharma, R. Kumar, A. Gupta, and V. Singh, "Blockchain-driven enhancement of SDN security in IoT-enabled cloud environments," Int. J. Inf. Technol. Secur. (Ingénierie des Systèmes d'Information), vol. 29, no. 6, pp. 2265–2273, 2024, doi: 10.18280/isi.290616.

[6] D. van den Berg, R. Glans, D. de Koning, F. A. Kuipers, J. Lugtenburg, K. Polachan, P. T. Venkata, C. Singh, and G. P. Fettweis, "Challenges in haptic communications over the tactile Internet," IEEE Access, vol. 5, pp. 23502–23518, 2017, doi: 10.1109/ACCESS.2017.2764181.

[7] R. Vairagade, L. Bitla, R. Pawar, and S. Ghode, "Hybrid blockchain software defined network architecture for secure and energy efficient IoT routing," J. Logist. Informat. Serv. Sci. (JLISS), vol. 12, no. 4, pp. 146–177, 2025, doi: 10.33168/JLISS.2025.0409.

[8] A. H. Mozumder, M. J. Basha, and C. A. R. Chayapathi, "SmartSecChain-SDN: A blockchain-integrated intelligent framework for secure and efficient software-defined networks," Int. J. Electr. Comput. Eng. (IJECE), vol. 12, no. 10, pp. 212–231, 2025, doi: 10.14445/23488549/IJECE-V12I10P117.

[9] O. Alnajar and A. Barnawi, "TactiFlex: A federated learning-enhanced in-content aware resource allocation flexible architecture for tactile IoT in 6G networks," Eng. Appl. Artif. Intell., vol. 136, p. 108934, 2024, doi: 10.1016/j.engappai.2024.108934.

[10] A. Rahman, M. J. Islam, S. S. Band, G. Muhammad, K. Hasan, and P. Tiwari, "Towards a blockchain-SDN-based secure architecture for cloud computing in smart industrial IoT," Digit. Commun. Netw., vol. 9, no. 2, pp. 411–421, Apr. 2023, doi: 10.1016/j.dcan.2022.11.003.

[11] M. A. AbdulRaheem, I. D. Oladipo, A. L. Imoize, J. B. Awotunde, C.-C. Lee, G. B. Balogun, and J. O. Adeoti, "Machine learning assisted Snort and Zeek in detecting DDoS attacks in software-defined networking," Int. J. Inf. Technol. (Springer), vol. 15, no. 8, pp. 4245–4257, 2023, doi: 10.1007/s41870-023-01469-3.

[12] D. Rico, K.-J. Grinnemmo, A. Brunstrom, and P. Merino, "Performance analysis of the multi-connection tactile Internet protocol over 5G," J. Netw. Syst. Manag., 2023, doi: 10.1007/s10922-023-09737-0.

[13] S. Choudhary and S. Dorle, "Secured SDN based blockchain: An architecture to improve the security of VANET," Int. J. Electr. Comput. Eng. Syst. (IJECES), vol. 13, no. 2, pp. 145–153, 2022, doi:

10.32985/ijeces.13.2.7.

[14] A. Mukherjee, R. Chatterjee, and J. K. Mandal, "A proposed technique for management of flow rules in SDN using blockchain," in *Proc. Int. Conf. Electr., Comput., Commun. Mechatronics Eng. (ICECCME)*, Maldives, Nov. 2022, pp. 1–5, doi: 10.1109/ICECCME55682.2022.9987763.

[15] S. Choudhary and S. Dorle, "SDN based blockchain architecture for security performance of VANETs," in *Proc. Int. Conf. Comput. Intell. Comput. Appl. (ICCICA)*, Nagpur, India, Nov. 2021, pp. 1–6, doi: 10.1109/ICCICA52458.2021.9697270.

[16] N. Promwongsa, A. Ebrahimzadeh, D. Naboulsi, S. Kianpisheh, F. Belqasmi, R. H. Glitho, N. Crespi, and O. Alfandi, "A comprehensive survey of the tactile Internet: State-of-the-art and research directions," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 1, pp. 472–523, 2021, doi: 10.1109/COMST.2020.3025995.

[17] V. Fanibhare, N. I. Sarkar, and A. Al-Anbuky, "A survey of the tactile Internet: Design issues and challenges, applications, and future directions," *Electronics*, vol. 10, no. 17, p. 2171, 2021, doi: 10.3390/electronics10172171.

[18] S. R. Al-Emari, Y. Sanjalawe, M. Anbar, and S. Manickam, "Intrusion detection systems using blockchain technology: A review, issues and challenges," *Comput. Syst. Sci. Eng. (CSSE)*, vol. 42, no. 2, pp. 4422–4448, 2021, doi: 10.32604/csse.2022.017941.

[19] G. Gurung, G. Bendiab, M. Shiaele, and S. Shiaeles, "CIDS: Collaborative intrusion detection system using blockchain technology," in *Proc. 2022 IEEE Int. Conf. Cyber Security and Resilience (CSR)*, Rhodes, Greece, Jul. 2022, pp. 125–130, doi: 10.1109/CSR54599.2022.9850331.

[20] J. Hu, M. Reed, N. Thomos, M. F. Al-Naday, and K. Yang, "Securing SDN-controlled IoT networks through edge blockchain," *IEEE Internet Things J.*, vol. 8, no. 4, pp. 2102–2115, Feb. 2021, doi: 10.1109/JIOT.2020.3021011.

[21] M. Abdel-Basset, N. Moustafa, H. Hawash, I. Razzak, K. M. Sallam, and O. M. Elkomy, "Federated intrusion detection in blockchain-based smart transportation systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 3, pp. 2523–2537, Mar. 2022, doi: 10.1109/TITS.2021.3119126.

[22] E. Ganesan, I.-S. Hwang, and A. T. Liem, "Resource allocation for tactile Internet via software-defined FiWi access network," in *Proc. Int. Comput. Symp. (ICS)*, Tainan, Taiwan, 2020, pp. 283–288, doi: 10.1007/978-981-19-9582-8.

[23] M. Signorini, M. Pontecorvi, W. Kanoun, and R. Di Pietro, "BAD: A blockchain anomaly detection solution," *IEEE Access*, vol. 8, pp. 173731–173740, 2020, doi: 10.1109/ACCESS.2020.3025622.

[24] T. Alharbi, "Deployment of blockchain technology in software-defined networks: A survey," *IEEE Access*, vol. 8, pp. 9146–9156, 2020, doi: 10.1109/ACCESS.2020.2965723.

[25] H. N. Nguyen, H. A. Tran, S. Fowler, and S. Souihi, "A survey of blockchain technologies applied to software-defined networking," *IET Wirel. Sens. Syst.*, vol. 11, no. 5, pp. 231–245, 2021, doi: 10.1049/wss2.12031.

[26] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: Techniques, datasets and challenges," *Cybersecurity*, vol. 2, no. 1, pp. 1–22, 2019, doi: 10.1186/s42400-019-0038-7.

[27] R. Chaudhary, A. Jindal, G. S. Aujla, S. Aggarwal, N. Kumar, and K.-K. R. Choo, "BEST: Blockchain-based secure energy trading in SDN-enabled intelligent transportation system," *Comput. Secur.*, vol. 85, pp. 288–299, Aug. 2019, doi: 10.1016/j.cose.2019.05.006.

[28] N. Kolokotronis, S. Brotsis, G. Germanos, C. Vassilakis, and S. Shiaeles, "On blockchain architectures for trust-based collaborative intrusion detection," in *Proc. IEEE World Congr. Serv. (SERVICES)*, 2019, pp. 21–28, doi: 10.1109/SERVICES.2019.00016.

[29] R. M. A. Ujjan, Z. Pervez, and K. Dahal, "Snort based collaborative intrusion detection system using blockchain in SDN," in *Proc. 13th Int. Conf. Software, Knowledge, Information Management and Applications (SKIMA)*, Ulkulhas, Maldives, Aug. 2019, pp. 1–8, doi: 10.1109/SKIMA47702.2019.8982413.

[30] M. Tanrıverdi and A. Tekerek, "Implementation of blockchain based distributed web attack detection application," in *Proc. 1st Int. Informatics and Software Engineering Conf. (UBMYK)*, Ankara, Turkey, Nov. 2019, pp. 1–6, doi: 10.1109/UBMYK48245.2019.8965446.

[31] S. Tug, W. Meng, and Y. Wang, "CBSigIDS: Towards collaborative blockchained signature-based intrusion detection," in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, Halifax, Canada, Jul. 2018, pp. 1–8, doi: 10.1109/Blockchain.2018.00011.

[32] W. Meng, W. Li, L. T. Yang, and P. Li, "Enhancing challenge-based collaborative intrusion detection networks against insider attacks ing blockchain," *Int. J. Inf. Secur.*, vol. 19, no. 3, pp. 279–290, 2020, doi: 10.1109/TPAS.1970.292563.