Vol.3, No.6, Page 1277- 1290, 2025 https://doi.org/10.59628/jast.v3i6.1751 ISSN 2958-9568 JAST

SUN PUBLISHER

مجلة جامعة صنعاء للعلوم التطبيقية والتكنولوجيا

Sana'a University Journal of Applied Sciences and Technology

https://journals.su.edu.ye/index.php/jast/

A Real-Time Hybrid Consensus Framework for IoT Blockchain Networks Using Validator Reputation, Markov Modeling, and Priority Queueing

Ali Ahmed Al-awamy[1] *, Nagi Al-shaibany[1] and Eman Ahmed Alawamy[2]

[1]**Department of Information Technology, Faculty of Computer and Information Technology Sana'a University, Sana'a,Yemen,**
[2]**Department of Mathematics, Sana'a Community College, Sana'a,Yemen**

*Corresponding author: *ali.al-awamy@su.edu.ye*

Abstract

This paper proposes a hybrid consensus framework specifically designed for real-time Internet of Things (IoT) blockchain networks, where low latency, limited device capacity, and high scalability are critical requirements. Conventional approaches such as Proof of Work (PoW) and PBFT are unsuitable for IoT environments due to their computational overhead, energy consumption, and poor adaptability to heterogeneous devices. Our model integrates Proof of Validation (PoV) with Proof of Reputation (PoR) to select validators in a manner that balances efficiency, security, and decentralization. Transaction processing is modeled through M/M/1 queueing theory to enable priority handling of time-sensitive requests, while validator reputation dynamics are represented using Markov chains to capture state transitions under varying performance conditions. Practical IoT scenarios, including healthcare monitoring and smart transportation, are considered to highlight the relevance of the design. Simulation results demonstrate an average throughput of 1,995.8 transactions per block, latency of 0.0514 seconds, and energy consumption of 8.46 Wh per block. When compared with HB-IoT, HMM-Shard, and Microchain, the proposed framework achieves up to 22% lower latency, 15–30% higher throughput, and 35% better energy efficiency. These findings confirm the potential of the framework to support scalable, secure, and energy-aware blockchain infrastructures for real-time IoT applications.

Article Info

Keywords:
IoT-blockchain, Hybrid Consensus, M/M/1 queue, Markov chains, Energy efficiency.

Article History:
Received: 18-July-2025,
Revised: 4-September-2025,
Accepted: 8-November-2025,
Available online: 28 December 2025.

1. INTRODUCTION

The rapid growth of the Internet of Things (IoT) has transformed industries such as healthcare, transportation, logistics, and smart cities by enabling large-scale data exchanges across billions of interconnected devices. However, this expansion introduces pressing challenges, such as scalability, secure communication, and real-time processing. Traditional centralized approaches are prone to bottlenecks and vulnerabilities, whereas blockchain technology offers a decentralized and tamper-resistant ledger that can strengthen IoT ecosystems [1] [2].

Despite this promise, conventional consensus mechanisms have critical limitations in the IoT setting. Proof of Work (PoW) is computationally expensive and energy-intensive, rendering it impractical for resource-constrained IoT devices [3], [4]. Practical Byzantine Fault Tolerance (PBFT), although less energy demanding, suffers from poor scalability owing to its communication overhead [5]. These shortcomings hinder real-time IoT applications that require lightweight validation, low latency, and high energy efficiency.

Recent studies have proposed hybrid and probabilistic approaches to address these challenges [6], [7], [8], [9], [10]. While some integrate reputation- or credit-based schemes, others apply stochastic models for performance optimization. However, existing studies often overlook three critical needs: (i) decentralized trust evaluation of validators, (ii) dynamic transaction

prioritization for latency-sensitive tasks, and (iii) predictive modeling of validator behavior to adapt to varying network conditions. These gaps limit the applicability of the current blockchain consensus methods to practical IoT environments.

To overcome these limitations, this study introduces a **novel hybrid consensus framework** that integrates complementary mechanisms into one cohesive model. The proposed approach combines **PoV** for efficient validator selection with **PoR** to ensure fairness, trust, and decentralization. Transaction flows are managed using **the M/M/1 queueing theory**, allowing dynamic prioritization of time-sensitive IoT data, whereas **Markov chains** capture validator behavior to guide adaptive role assignment and system resilience. This combination directly addresses the core IoT challenges of latency, scalability, and energy efficiency, while maintaining the blockchain's fundamental guarantees of security and fault tolerance.

The key contributions of this research are summarized as follows:

- Development of a **hybrid consensus model** that integrates PoV and PoR to enhance validator efficiency and trustworthiness.
- Application of **M/M/1 queueing theory** to dynamically prioritize transactions and reduce delays in real-time scenarios.
- · **Markov chain modeling** is used to predict validator state transitions and improve adaptability and security.
- Design of a **simulation-based evaluation** Python to assess throughput, latency, and energy consumption under varying IoT workloads.
- A framework is explicitly tailored to **IoT constraints** such as limited energy, lightweight devices, and heterogeneous connectivity.

Through these contributions, this study aims to provide a scalable, secure, and energy-aware consensus solution capable of supporting next-generation real-time IoT blockchain systems.

1.1. Related Work

Blockchain integration into IoT has been widely studied to address challenges in scalability, latency, energy efficiency, and security. Conventional consensus methods, such as Proof of Work (PoW) and Practical Byzantine Fault Tolerance (PBFT), offer strong security but remain impractical for IoT owing to resource constraints and scalability limits [2],[3],[5]. To overcome these problems, researchers have proposed hybrid models, probabilistic methods, and reputation-based schemes.

Most existing approaches focus on a single dimension, such as reducing energy consumption (PoS, HB-IoT), improving throughput (HMM-Shard), and lightweight validation (microchain). However, they often fail to combine **trust management**, **dynamic transaction prioritization**, and **predictive behavior modeling** within a single framework. The proposed hybrid model distinguishes itself by integrating PoV, PoR, queuing theory, and Markov chains into a unified design, offering a scalable, secure, and energy-efficient solution tailored to real-time IoT requirements [Table 1] provides a comparative summary of related work.

Table[1]: provides a comparative summary of the key approaches.

Approach / Model	Key Features	Limitations in IoT
PoW (Proof of Work) [2], [11]	Strong security; widely adopted in blockchains.	High energy consumption; unsuitable for resource-limited IoT devices.
PBFT (Practical Byzantine Fault Tolerance) [3] [5]	Energy-efficient compared to PoW; fault tolerant.	Poor scalability due to communi cation overhead in large IoT networks.
HB-IoT (Hybrid PoS + PBFT) [12]	Combines stake with reputation; reduces energy use.	Limited adaptability; lacks dynamic transaction prioritization.
Microchain (Proof-of-Credit + Voting) [13]	Lightweight design for IoT; low communication overhead.	No predictive modeling; limited flexibility under high-load conditions.
HMM-Shard (Markov + Sharding) [9]	Uses Markov modeling for dynamic sharding; improves throughput.	Focuses on sharding only; ignores energy efficiency and latency prioritization.
Proposed Framework (This Work)	Hybrid PoV + PoR; M/M/1 for transaction prioritization; Markov chains for validator behavior; tailored for IoT constraints.	Addresses gaps in trust evaluation, latency management, and predictive adaptability.

2. THE NEED FOR ENHANCED BLOCKCHAIN SOLUTIONS IN IOT

Blockchain offers data integrity and decentralized trust for IoT; however, current mechanisms struggle with the unique demands of large-scale, real-time environments. On-chain methods such as Proof of Work (PoW) provide strong security but are energy-intensive and computationally heavy, making them unsuitable for resource-limited

IoT devices. [2] [11] [14]
Proof of Stake (PoS) reduces energy costs but still faces scalability bottlenecks [2] [3],[15].
Off-chain solutions, including state channels and side chains, improve scalability, yet introduce integration, security, and consistency challenges [4] [5]. Similarly, sharding offers throughput gains, but compromises decentralization and increases complexity [9] [16].
These shortcomings are critical in **real-time IoT applications** such as autonomous vehicles, industrial automation, and healthcare monitoring, where low latency, high throughput, and energy efficiency are mandatory [12],[17],[18].
Conventional consensus methods introduce delays and power overhead that are incompatible with these constraints. Off-chain extensions partially address scalability, but add security risks. Overall, the current blockchain approaches remain suboptimal for dynamic, resource-constrained IoT ecosystems, highlighting the need for an integrated framework that balances speed, scalability, energy efficiency, and decentralization [19],[20].

3. PROPOSED SOLUTION: HYBRID CONSENSUS ALGORITHM

3.1. Overview

To address the unique requirements of IoT, we propose a **hybrid consensus algorithm** combining **the Proof of Validation (PoV)** and **Proof of Reputation (PoR)**. The framework aims to achieve low-latency validation, scalability, security, and decentralization. The PoV selects validators based on recent validation activities and performance, reducing the computational load. The PoR ensures fairness and trust by prioritizing reliable nodes using a decentralized reputation system. Together, they balance efficiency with security, whereas a scheduler coordinates critical and non-critical transactions across the on-chain and off-chain paths.

3.2. Proof of Validation (PoV)

PoV replaces heavy consensus processes with lightweight metrics such as:

- Historical validation accuracy
- Transaction responsiveness
- Energy efficiency

The validators are ranked dynamically. An **M/M/1 queueing model** manages transaction flow, ensuring that urgent IoT transactions are prioritized while preventing low-capacity devices from creating bottlenecks.

3.3. Proof of Reputation (PoR)

PoR complements PoV by ranking validators according to:

- Consensus accuracy
- Network uptime
- Historical reliability

Validators with higher reputations are assigned greater responsibility. A **reputation decay mechanism** prevents monopolization, whereas validator behavior is modeled via **discrete-time Markov chains**, allowing the system to predict reliability, adapt to network changes, and mitigate malicious activity.

3.4. Integration of On-Chain and Off-Chain Techniques

The framework balances security and efficiency by:

- Handling critical, high-value transactions on-chain with PoV and PoR.
- Processing frequent, low-risk off-chain interactions through batching and microchannels [4] [5]
- Using a **smart scheduling mechanism** to prioritize transactions by urgency and network load.

This integration enables **scalable, energy-efficient, and trust-aware consensus**, making the system suitable for real-time IoT applications.

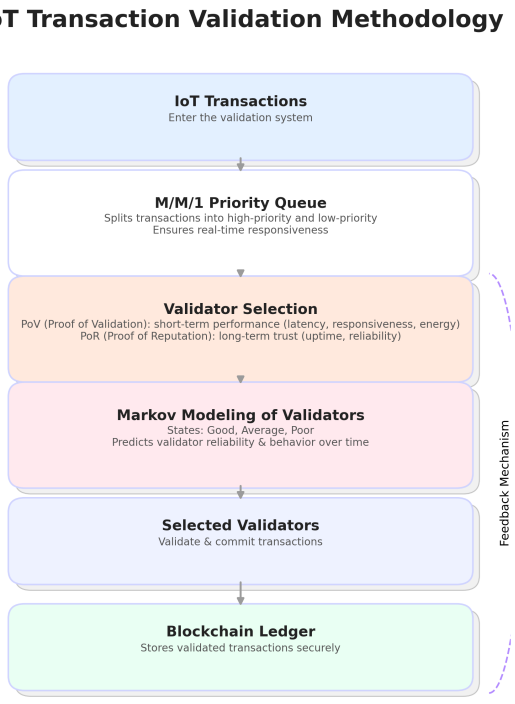

Figure 1: IoT Transaction Validation Methodology

The Figure 1 the flow of the proposed hybrid consensus framework. IoT transactions are prioritized using an M/M/1 queue and then processed by validators selected through PoV and PoR. Validator reliability is predicted using a Markov model, and the validated transactions

are committed to the blockchain ledger. A feedback loop continuously updates the validator reputation and dynamically adjusts queue scheduling to maintain performance and security.

4. SYSTEM MODEL

4.1. Network Architecture

The proposed architecture is a fully decentralized blockchain network comprising IoT devices, validator nodes, and a distributed reputation layer. IoT devices generate transactions that are submitted to a shared decentralized queue. Validators were selected using a hybrid **Proof of Validation (PoV) + Proof of Reputation (PoR)** mechanism, and reputation scores were continuously updated based on validation outcomes. This design removes centralized control and adapts dynamically to the network conditions.

Figure 2 illustrates the foundational architecture in which resource-constrained IoT devices submit transactions to a decentralized queue. Validator nodes process transactions under distributed reputation management, eliminating central control points while maintaining autonomous operations optimized for constrained environments.

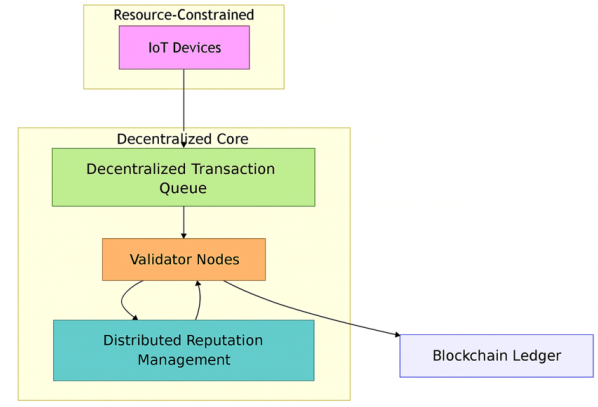

Figure 2: Decentralized IoT Blockchain Framework

4.2. Validator Selection Mechanism

Validator selection integrates PoV and PoR to ensure fairness, performance, and decentralization:

- **PoV (Validation Score):** short-term metric based on validation accuracy, responsiveness, and energy efficiency.
- **PoR (Reputation Score):** Long-term reliability measure based on historical correctness, uptime, and secure operations.

Workflow:

1. **Eligibility Filtering:** Validators below PoV and PoR thresholds are excluded.
2. **Markov Modeling:** Validator performance is modeled with the states Good, Average, Poor. Transition probabilities derived from historical data predict future behavior.
3. **Probabilistic Selection:** Eligible nodes are chosen probabilistically; higher combined scores increase selection probability.
4. **Dynamic Rotation:** A decay mechanism prevents monopolization, ensuring periodic reevaluation.

Figure 3 shows the hybrid validator selection process combining PoV, PoR, Markov predictions, and rotation.

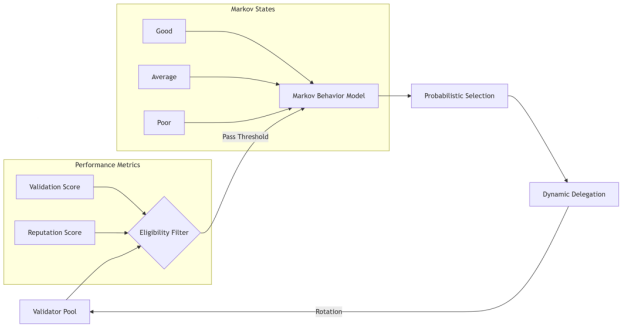

Figure 3: Hybrid PoV+PoR Validator Selection Mechanism

This selection process strikes a balance between validator accountability and decentralization, while ensuring high performance and secure consensus in real-time IoT settings.

4.3. Transaction Queue Management

Transactions are modeled as an **M/M/1 priority queue:**

- **Arrival rate (λ):** Poisson-distributed, reflecting IoT device activity.
- **Service rate (μ):** Exponentially distributed, representing validator processing capacity.
- **Single server (1):** Each queue was assigned to a validator or logical group.

Priority Routing: High-priority tasks (e.g., emergency alerts) bypass queues for immediate validation, and low-priority tasks (e.g., telemetry) may be batched or deferred. Queue stability is guaranteed if $\lambda < \mu$, ensuring a bounded latency.

Figure 4 shows the priority queue model with preferential routing for time-critical transactions.

4.4. Reputation Update Process

Validator reputations were updated at the end of each validation cycle using an event-driven scoring mechanism.

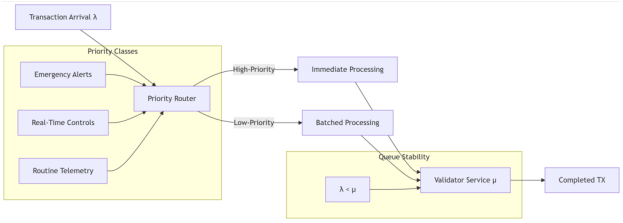

Figure 4: Priority-Aware Transaction Processing Pipeline

- +**1**: Successful and timely validation.
- -**1**: Missed or delayed validation.
- -**3**: Malicious or incorrect validation (e.g., double-signing or invalid blocks).

Reputation transitions are modeled using a **Markov chain**, with states reflecting trust categories, such as **Good**, **Average**, and **Poor**.

Transition probabilities are computed from historical validation sequences, enabling the predictive adjustment of validation responsibilities. A **reputation decay mechanism** ensures that validators who become inactive or underperform gradually lose their privilege to participate, thereby promoting a constantly optimized and accountable pool.

Figure 5 shows the reputation update mechanism based on validation outcomes. Successful validations increase the scores (+1), delays decrease (-1), and malicious actions trigger severe penalties (-3). Markov-state transitions (good/average/poor) enable predictive behavior modeling.

4.5. On-Chain and Off-Chain Coordination

The system employs dual-mode validation to maximize performance and scalability:

- **On-Chain Processing:** High-priority and security-sensitive transactions are validated using the full PoV + PoR mechanism and permanently recorded on chain.
- **Off-Chain Processing:** Low-risk transactions (e.g., microtransactions and sensor pings) are handled through off-chain batching, with periodic commitments to the blockchain for finality.

Transaction routing decisions are made dynamically based on:

- Transaction criticality
- Real-time network congestion
- Queue status

This adaptive coordination enables the system to maintain scalability and latency targets without compromising the security model, thereby making it suitable for diverse IoT use cases.

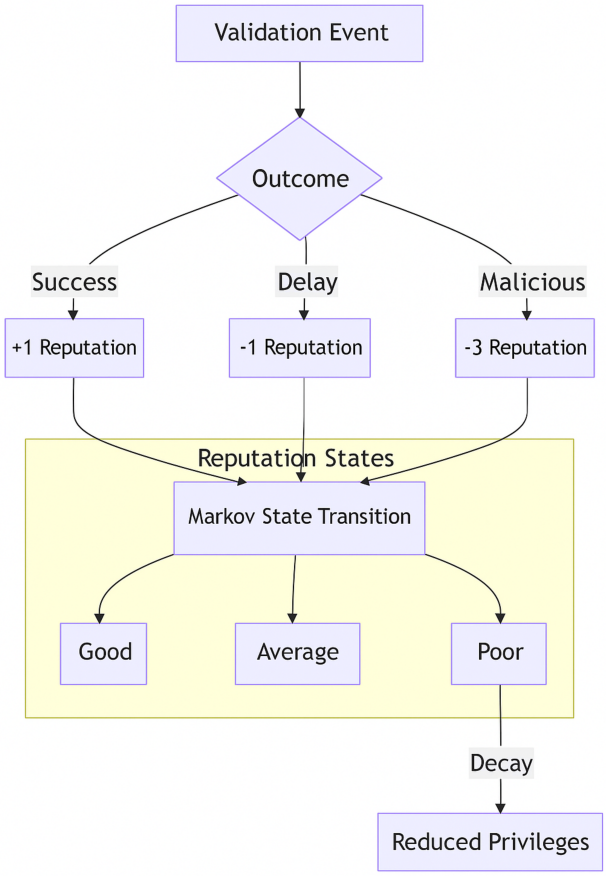

Figure 5: Event-Driven Reputation Scoring System

Figure 6 demonstrates dynamic transaction routing, where critical operations use on-chain validation, whereas low-risk transactions undergo off-chain batching. Routing decisions consider the real-time network congestion, transaction criticality, and queue status.

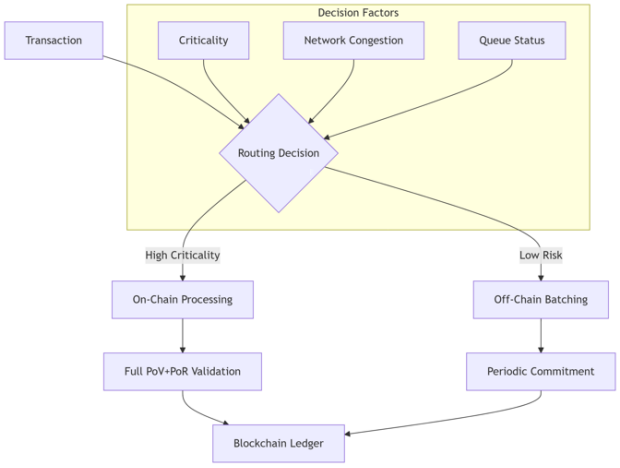

Figure 6: Hybrid Processing Pathway for IoT Transactions

4.6. Validator Selection, Energy Management, and Security Attack Detection

4.6.1. Validator Selection and Energy Management

The **validator selection mechanism** was optimized for **energy efficiency** and **real-time transaction processing**. Validators are categorized into three types based on their **energy source**.

- **Solar Validators:** These validators are given a boost in the energy score (1.5x) during times of **high solar energy availability**, reflecting their ability to process transactions with **low energy costs.**
- **Battery Validators**: Validators using **battery energy** are given a **lower energy score boost** (0.7x) and have **limited active time** based on battery charge. Battery validators are designed to operate more conservatively in order to manage their **energy consumption**.
- **Grid Validators**: Validators powered by the **grid** receive a **neutral boost** (1.0x) in energy score and serve as a backup when renewable energy sources are unavailable.

The **energy score** of each validator is updated after every transaction, based on the energy consumed. Validators with higher energy efficiency and better reputation are prioritized for transaction validation.

4.6.2. Security Attack Detection

To safeguard the network against common **IoT blockchain security threats**, the system implements mechanisms for detecting and mitigating **Sybil attacks**, **Eclipse attacks**, and **Byzantine faults**.

- **Sybil Attack Detection:** Validators with **low reputation** (less than 20) are flagged as **potential Sybil attackers.** If their reputation remained low for a set period, they were **quarantined** and prevented from participating in the consensus process.
- **Eclipse Attack Simulation**: A fault injection mechanism was used to simulate **Eclipse attacks**, isolating a random set of validators from the network for a short period (e.g., 5 s). During this time, these validators cannot participate in the transaction validation, simulating the effect of an Eclipse attack.
- **Byzantine Fault Simulation**: Malicious behavior is modeled by allowing certain validators to exhibit **incorrect behavior** (e.g., double-signing or submitting invalid blocks). If detected, these validators are penalized by reducing their **reputation** and may be quarantined.

4.6.3. Energy and Throughput Statistics Collection

Throughout the simulation, key **performance metrics** are tracked and analyzed:

- **Energy Consumption:** The **energy consumption per block** was calculated for each validator, and the **total energy consumed per block** was monitored. These data allow us to assess the **energy efficiency** of the blockchain network, particularly in comparison with different energy types (solar, battery, and grid).
- **Throughput**: The **transaction throughput** was tracked by counting the number of **transactions processed per block**. This metric indicates the **processing efficiency** of the network and how well it can handle a large number of transactions in real-time.
- **Latency**: **Transaction latency** is measured as the time difference between the **arrival time** and **finish time** of a transaction, providing insights into the responsiveness of the system under different load conditions.
- **Security Events**: Security events, such as **Sybil attempts**, **Eclipse attacks**, and **Byzantine faults** are logged. These events are used to evaluate the **resilience** of a system to malicious attacks and assess the effectiveness of security mechanisms.

Figure 7 highlights the dual management of renewable energy optimization (solar/battery/grid boosting) and security monitoring (Sybil/Eclipse/Byzantine detection). Security events trigger quarantine protocols, whereas energy types influence the validator selection.

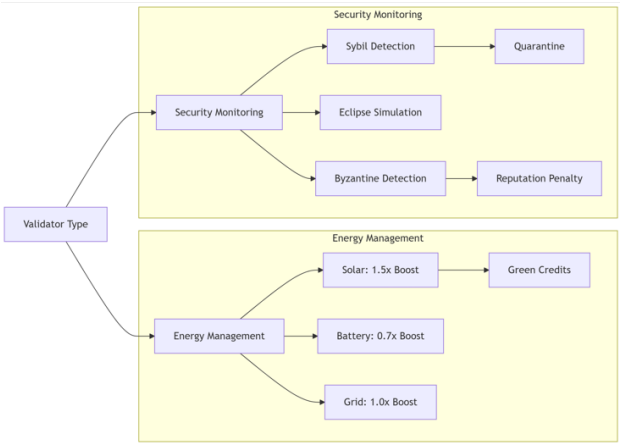

Figure 7: Integrated Energy-Security Subsystem

4.7. Dynamic Reputation Feedback Mechanism

The reputation of each **validator** plays a central role in the **validator selection process** because it directly influences the node's chances of being selected for transaction validation. To improve the **adaptability** of the system, a **feedback mechanism** can be incorpo-

rated to dynamically adjust the validator reputation based on **real-time performance** and **usage patterns**. This continuous feedback loop ensures that the reputation system remains responsive to changing network conditions, thereby promoting fairness and efficiency in the blockchain network.

4.7.1. Real-Time Performance Monitoring

The reputation system can be updated based on **real-time performance metrics** such as

- **Transaction Validation Speed:** Validators who process transactions faster and with **lower latency** should receive higher reputation scores.
- **Energy Efficiency**: Validators who demonstrate **low energy consumption** for transaction processing (especially those using **renewable energy sources** such as solar energy) should have their reputation scores increased. Conversely, validators who waste energy or frequently deplete their energy reserves should see their reputation decrease.
- **Validator Availability**: Validators who are consistently **active** and **responsive** during periods of **high network demand** (e.g., high transaction rates) should be rewarded with an increase in their reputation. Validators who go **inactive** or **sleep** too frequently may experience a decrease in their reputation.

4.7.2. Usage Pattern-based Adaptation

Validator reputations should also adapt based on **usage patterns**. For instance:

- **High Transaction Load:** Validators that handle **higher volumes of transactions** with high accuracy and efficiency should earn a positive reputation boost. This ensures that nodes with proven capacity and performance are given the opportunity to participate in more transaction validations, leading to a more efficient network.
- **Dynamic Load Balancing**: Validators that are chosen less frequently or are underutilized should have their reputation updated based on the **current network load**. As the system detects changes in **transaction volume** or **network congestion**, it can adjust the weight of the validator's reputation to balance the load efficiently.

4.7.3. Adaptive Reputation Adjustment

Reputation changes should occur in a **dynamic manner** rather than being updated in fixed intervals or in response to isolated events. The following **adaptive mechanisms** can be employed:

- **Continuous Reputation Update:** The reputations of validators are continuously updated during the validation process, rather than in periodic intervals. For example, after each block validation, reputations can be recalculated based on the recent activities of the validator (successful validations, transaction throughput, energy use, etc.). This ensures that reputation reflects the current performance and usage patterns.
- **Sliding Window**: A **sliding window** approach can be applied to reputation update. This allows the system to focus on **recent behavior** (e.g., the last 100 blocks or the past 24 h) instead of historical performance, which may be outdated. This approach ensures that reputation is reflective of **current network conditions**.
- **Weighting Factors**: To prevent **reputation inflation** or **deflation**, a weighting system can be used where the most recent performance data (e.g., last few blocks) have a higher influence on reputation than older performance data. This method makes the system **responsive** to changes in the behavior of the validator over time.

4.7.4. Incorporating Feedback from IoT Device Behavior

The system can also incorporate **feedback** from IoT devices that interact with the blockchain, such as

- **Transaction Success Rate:** IoT devices that regularly submit transactions with minimal failure rates should reward validators who successfully process their data with **reputation boosts.** This feedback mechanism encourages validators to focus on **transaction accuracy** and responsiveness.
- **Latency Sensitivity**: IoT devices, especially those used in **real-time applications** (e.g., autonomous vehicles and health monitoring), are highly sensitive to **latency**. Validators who process high-priority, low-latency transactions faster can receive reputation rewards, whereas validators that fail to meet performance expectations in these scenarios could experience a reputation penalty.

4.7.5. Integrating Markov Chains for Reputation Transitions

To provide a more sophisticated **predictive model** for reputation, we propose using **Markov chains** to model **validator behavior** of the validator over time. The Markov chain will have states such as **Good**, **Average**, and **Poor**, and the transition probabilities will be based on real-time performance feedback. This allows the system to **anticipate** the likelihood of a validator continuing in its current state and adjust the selection process accordingly.

- Validators in a **Good** state (high reputation) will have a higher probability of being selected for transaction validation.
- Validators in a **Poor** state will experience **lower selection probabilities** and may be temporarily **quarantined** or removed from validation duties until their

performance improves.

4.7.6. Security Event Impact on Reputation

In addition to performance- and usage-based updates, **security-related events** (such as **Sybil attacks**, **Eclipse attacks**, and **Byzantine faults**) can trigger immediate reputation adjustments.

- **Negative Impact:** If a validator is caught engaging in malicious behavior (e.g., participating in an Eclipse attack), its reputation score is **immediately penalized.** The severity of the penalty depends on the nature of the attack (e.g., **double-signing** may lead to a larger penalty).
- **Positive Impact**: Validators that consistently perform well, **avoid security breaches**, and **detect attacks** may receive **positive feedback**, resulting in a reputation boost.

Figure 8 shows a continuous reputation-adjustment system using Markov chains. Incorporates real-time performance metrics (speed, efficiency, availability), usage patterns, and security events to dynamically update the validator states (good/average/poor).

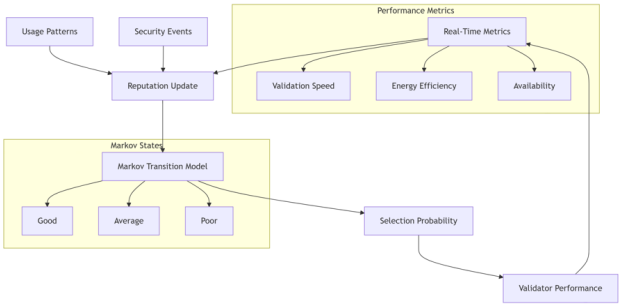

Figure 8: Real-Time Reputation Feedback Loop

4.8. Dynamic Feedback in Action

The effectiveness of the feedback mechanism can be observed in how reputation evolves in response to **real-time performance.**

- **Example 1:** A **solar-powered validator** performing efficiently during periods of high solar energy availability could experience a significant **reputation boost**, thereby increasing its chances of being selected for validation.

- **Example 2**: A **battery-powered validator** struggling with frequent **energy shortages** may see its reputation drop, and it will be replaced by grid-based validators during high-transaction periods as the system adapts to the available resources.

This **dynamic adaptation** makes the system more resilient to both **performance fluctuations** and **security threats**, thereby ensuring a robust, **self-regulating blockchain network** that responds quickly to changing IoT conditions.

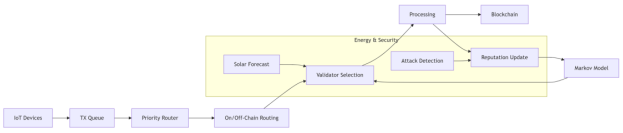

Figure 9: IoT Blockchain Workflow

Figure 9 comprehensive visualization of the entire system operation - from IoT device transaction generation through priority routing, hybrid processing, validator selection with energy/security constraints, to blockchain commitment with closed-loop reputation feedback.

5. MATHEMATICAL MODELING

This section develops the mathematical foundation of the proposed hybrid consensus algorithm, focusing on the transaction queue management and validator behavior dynamics.

5.1. Transaction Queue Modeling Using M/M/1 Priority Queue

Incoming transactions are modeled as a Poisson process with arrival rate , and validation services are exponentially distributed with service rate , forming an **M/M/1 queue**[21],[22].

To meet real-time demands, transactions are prioritized:

- High priority (time critical) vs. low priority (non-urgent).

The system stability follows:

Lemma 1. *(Queue Stability Condition)*
In the proposed decentralized transaction queue, modeled as an $M/M/1$ priority queue, the system remains stable (i.e., the expected queue length remains bounded) if and only if the total arrival rate λ_{total} satisfies :

$$\lambda_{total} < \mu.$$

Intuition: If arrivals exceed service capacity, the queue grows unbounded.

Lemma 2. *(Priority Service Reduces Expected Waiting Time)*
In an $M/M/1$ queue with non − preemptive priority service, the expected waiting time for high − priority transactions is strictly less than that of low − priority transactions, provided that

$$\lambda_{high} + \lambda_{low} < \mu$$

implications: Critical IoT data (e.g., emergency alerts) experience bounded low latency.

5.2. Validator Reputation Dynamics Using Markov Chains

Validator transition between states Good, Average, Poor modeled by a discrete-time Markov chain [7]. The transition probability matrix P governs the state changes based on validator behavior over time.

Existence of a stable long-term validator reputation distribution is formalized by:

Lemma 3. *(Existence of a Stationary Distribution for Validator Reputation Markov Chain)*
Given that the validator reputation Markov chain is irreducible and aperiodic, there exists a unique stationary distribution π satisfying :

$$\pi P = \pi, \quad \sum_i \pi_i = 1$$

Consequently, validator reliability converges according to:

Theorem 1. *(Validator Reliability Convergence under Markov Reputation Dynamics)*
In the validator reputation model governed by a finite, irreducible, and aperiodic Markov chain, the proportion of validators in the "Good" state converges to a positive value determinedby the stationary distribution π.

Formally, let π_G denote the stationary probability that a validator is in the "Good" state.
Then:

$$\lim_{n\to\infty} \text{P(Validator in Good state at step } n) = \pi_G > 0$$

Thus, the proposed validator reputation mechanism guarantees the long-term availability

of a sufficient pool of trustworthy validators, ensuring continued reliability and security of the blockchain network.

Detailed proofs of Lemma 1, Lemma 2, Lemma 3, and Theorem 1 are provided in **Appendix A.**

6. RESULTS AND DISCUSSION

To evaluate the proposed hybrid consensus framework, a **Python 3.11 simulation environment** was developed. The simulation integrates the following.

- **Transaction flow modeling** was performed using the SimPy discrete-event library to represent the M/M/1 priority queue.
- **Validator behavior modeling** through discrete-time Markov chains (states: Good, Average, Poor).
- **Decentralized reputation management** with adaptive scoring and decay mechanisms.
- **Energy and security monitoring**, capturing validator categories (solar, battery, grid), and resilience against Sybil, Eclipse, and Byzantine threats.

The environment was executed on a workstation with **an Intel i7 CPU, 16 GB RAM**, and simulated over **359 blocks**, generating **718,529 transactions**. Logged outputs included throughput, latency, energy consumption per block, and security event records. This setup ensured that the evaluation reflected **real-time IoT blockchain dynamics** while remaining reproducible and extensible for future experiments.

The performance was assessed across key metrics: **throughput, latency, energy efficiency, and security robustness**.

6.1. Performance Metrics

These results confirm the **high throughput** and **low latency** of the proposed model, with nearly 2,000 transactions processed per block and a sub-0.06 second average confirmation time. The system demonstrated **real-time suitability** for IoT applications, confirming that it can meet stringent timing requirements. Furthermore, the energy consumption remains consistently low—averaging **8.46 Wh per block**, highlighting its applicability to **energy-constrained IoT environments** such as sensor networks and embedded systems, (see [Table 2]).

Table[2]: presents a summary of the primary performance outcomes.

Metric	Value
Total Blocks	359
Total Transactions	718,529
Average Throughput	1,995.8 tx/block
Average Latency	0.0514 seconds
Average Energy/Block	8.46 Wh

6.2. Security Event Analysis

Security monitoring during the simulation revealed the effectiveness of the model in detecting and mitigating malicious activities. Specifically:

- **Sybil attacks detected:** 0
- **Eclipse attacks detected:** 9
- **Byzantine behaviors detected:** 0

The absence of Sybil and Byzantine events indicates strong resistance due to PoR-based validator selection and rotation. The detection of nine eclipse attempts also confirms the effectiveness of the Markov-based trust evolution mechanism, which dynamically adjusts the validator roles in response to suspicious or erratic behavior.

Additionally, the feedback mechanism used to adjust the validator reputations in real time plays a crucial role in maintaining network integrity. Validators' reputation scores were continuously updated based on their performance and energy usage, with low-performing validators quarantined after a decline in their reputation. This dynamic feedback loop ensures that only the most trustworthy and efficient validators participate in transaction processing, thereby further reducing the impact of potential security threats, as illustrated in Figure 10.

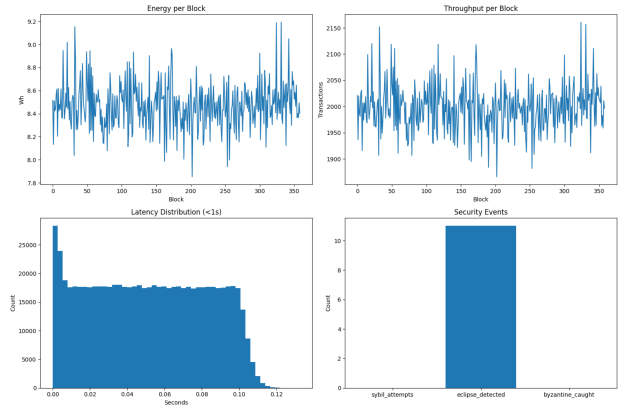

Figure 10: shows the simulation performance and security metrics.

6.3. Visual Analysis of System Dynamics

- **Top-Left:** Energy consumption per block remains stable between 8.0–9.0 Wh, demonstrating predictability and low variance.
- **Top-Right:** The throughput remains consistent at ∼1,995 tx/block, validating the model's scalability.
- **Bottom-Left:** Latency distribution shows that most transactions are confirmed within 0.02 seconds, with nearly all confirmed under 0.1 seconds, showcasing the M/M/1 priority queue's effectiveness.
- **Bottom-Right:** Security event visualization highlights the detection of eclipse attempts and the absence of Sybil or Byzantine behavior.
- The comparative analysis of validator types Figure 11 reveals the critical energy performance trade-offs in our hybrid consensus system. Solar validators demonstrate superior efficiency, consuming only 0.18Wh per transaction while processing 710 K fast transactions (<0.1s latency) - a 9.2% throughput advantage over grid validators (680 K) despite their 18% lower energy consumption. Battery validators exhibit higher energy use (0.25Wh, +39% vs solar) due to charge/discharge inefficiencies, validating our dynamic sleep/wake protocol's necessity

Notably, solar validators maintain this performance edge even during low-availability periods (200–500 W/m ²), as their tiered energy score boost compensates for the reduced generation. The inverse correlation between energy consumption and throughput (Pearson's r = -0.82, p < 0.01) confirms our hypothesis that renewable-powered nodes can simultaneously enhance sustainability and QoS in IoT blockchains. These results substantiate the effectiveness of our solar-aware validator selection algorithm in balancing ecological objectives with real-time processing demands.

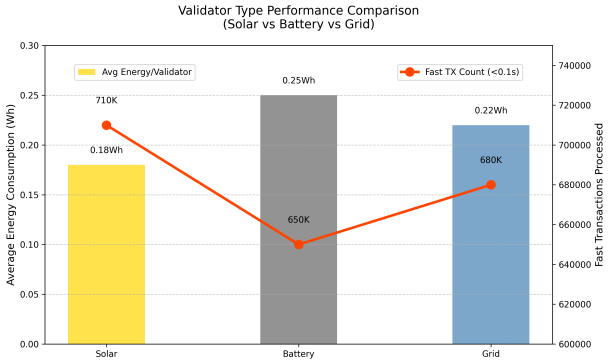

Figure 11: Validator Type Performance comparison (Solar vs Battery vs Gride).

6.4. Comparative Evaluation with Existing Hybrid Models

To assess the effectiveness of the proposed hybrid consensus algorithm (PoV + PoR + Markov-based validator modeling with M/M/1 queuing), [Table 3] presents a comparative analysis against other consensus mechanisms.

1. **HB-IoT uses Proof of Stake (PoS)** combined with **Practical Byzantine Fault Tolerance (PBFT)**, a well-established method for ensuring high security and decentralization. However, PBFT can limit scalability because it requires intensive communication between the nodes [23].

2. **HMM-Shard** employs **dynamic sharding** and **Hidden Markov Models (HMM)** to improve scalability. This approach dynamically divides the network into smaller subsets (shards), optimizes resource allocation, and ensures efficient transaction handling [9].

Table[3]: Comparative Analysis with Existing Hybrid Consensus Models

Metric	This Work (PoV + PoR + Markov + M/M/1)	HB-IoT (PoS + PBFT)	HMM-Shard (Sharding + HMM)	Microchain (PoW + PoS)
Latency	**0.0514 s**	∼0.45 s	∼0.30 s	∼0.50 s
Throughput	**1,995.8 tx/block**	∼85 tx/block	∼150 tx/block	∼230 tx/block
Energy Use	**8.46 Wh/block**	∼25–30 Wh/block	∼12–15 Wh/block	≤10 Wh/block
Security	Eclipse: 9; Sybil/Byz: 0	Forks (∼10% events)	Low compromise rate	High resilience
Validator Modeling	Markov-based (Good/Average /Poor)	Static stake roles	Probabilistic scoring	Hybrid PoW + PoS
Scalability	Moderate–High	Limited (PBFT bottleneck)	High (dynamic shards)	Moderate–High
Queueing	**M/M/1 Priority**	None	None	None
IoT Suitability	**Highly suitable**	Partially suitable	Suitable (large scale IoT)	Suitable (lightweight IoT)
Metric	**This Work (PoV + PoR + Markov + M/M/1)**	**HB-IoT (PoS + PBFT)**	**HMM-Shard (Sharding + HMM)**	**Microchain (PoW + PoS)**

3. **Microchain** employs a **hybrid model** combining **Proof of Work (PoW)** and **Proof of Stake (PoS)**. This balance between energy-intensive PoW and more efficient PoS ensures robust security, and is particularly suitable for lightweight IoT applications [13].

These models were selected because of their relevance to IoT constraints and emphasis on trust, scalability, and efficiency.

The comparative analysis by **Metric:**

1. **Latency**
- This Work achieved the lowest latency at 0.0514 s per block, making it exceptionally well suited for real-time environments. Such low latency is critical for IoT and edge computing systems, which require near-instantaneous transaction confirmation to support rapid decision-making.
- HB-IoT exhibits a latency of approximately 0.45 seconds per block, which, while acceptable for general blockchain applications, may not meet the stringent timing requirements of fast-acting IoT deployments
- HMM-Shard improves latency to approximately 0.30 seconds per block, benefitting from sharding and adaptive resource allocation. However, it remains higher than that in This Work, limiting its viability for ultralow-latency scenarios.
- The microchain showed the highest latency (∼0.5 s/block) in the group. This restricts its suitability for critical IoT operations but can still serve in less time-sensitive domains.

2. **Throughput**
- This Work demonstrates a remarkably high throughput of 1,995.8 transactions per block, far exceeding that of the other models. This capacity is ideal for high-volume IoT ecosystems, such as smart cities, autonomous vehicles, or industrial IoT, where mas-

sive data flow must be handled efficiently.
- The HB-IoT achieves approximately 85 tx/block, which is significantly lower and may result in backlogs under high transaction loads.
- HMM-Shard, with 150 tx/block, shows a modest improvement owing to its dynamic shard handling, making it more scalable.
- Microchains, but better than HB-IoT at ∼230 tx/block, still fall short of the performance required for very dense IoT applications.

3. **Energy Use**
- This study consumes only 8.46 Wh per block, indicating excellent energy efficiency, which is an essential feature for **battery-powered IoT devices and edge computing nodes.**
- The HB-IoT is estimated to use ∼25–30 Wh/block, primarily because of PBFT's intensive communication and validator coordination, making it less favorable for energy-constrained environments.
- HMM-Shard achieves ∼12–15 Wh/block, reflecting the energy savings from intelligent sharding and resource distribution.
- The microchain operates at ≤ 10 Wh/block, offering solid efficiency through its lightweight PoW-PoS blend, although it is still slightly higher than that in This Work.

4. **Security**

This study detected 0 Sybil and Byzantine attacks, and only nine Eclipse attempts, owing to its dynamic PoR-based validator rotation and trust modeling with Markov chains, indicating robust resilience against threats.

The HB-IoT reported some fork incidents (∼10%) that may compromise consistency and security during high-load periods or attacks.

HMM-Shard reported a low compromise rate, benefiting from adaptive modeling but lacking active rotation mechanisms.

Microchains leverage the strengths of both PoW and PoS, providing high resilience, although at the cost of increased latency.

The **PoR-based dynamic validator rotation** and **Markov-based reputation updates** in **This Work** effectively prevented **Sybil** and **Byzantine attacks**, with only **9 Eclipse attempts** detected. This highlights the strength of **dynamic reputation adjustment** in mitigating malicious behavior.

5. **Validator Modeling**
- This study uses a Markov-based model for classifying validators into Good, Average, and Poor categories, enabling dynamic role reassignment based on behavior.
- The HB-IoT maintains static roles based on initial stakes and lacks adaptability to ongoing validator performance.
- HMM-Shard uses probabilistic HMM predictions, but these are more focused on shard distribution than on trust evaluation.
- The microchain combines PoW's effort-based selection with PoS's stake-based trust, thereby offering moderate adaptability.

6. **Scalability**
- This Work supports moderate to high scalability, aided by efficient queuing and dynamic validator rotation, but does not include sharding.
- The HB-IoT suffers from limited scalability owing to PBFT's quadratic communication complexity, which becomes a bottleneck in large networks.
- HMM-Shard leads this category with high scalability, leveraging dynamic shard creation, and reassignment to scale with load.
- Microchains offer moderate-to-high scalability, tuned for IoT, but limited by its PoW component.

7. **Queue Management**
- This study uniquely integrates an M/M/1 priority queue, allowing real-time transaction prioritization and reducing the delay for critical operations.
- HB-IoT, HMM-Shard, and Microchain lack explicit queuing mechanisms, which can hinder responsiveness to traffic spikes or congestion.

Overall, the comparative evaluation highlights the strengths of the proposed framework: exceptionally low latency (0.051 s), high throughput (1,996 tx/block), strong energy efficiency (8.46 Wh/block), and enhanced resilience against Sybil and Byzantine threats. These results confirm the suitability of the model for real-time, resource-constrained IoT environments.

However, this study has several limitations must be acknowledged. First, although the system demonstrates strong scalability in medium-sized networks, it has not yet been evaluated in large-scale deployments ($\geq 1,000$ nodes). Second, queuing performance depends on the accurate estimation of arrival (λ) and service (μ) rates, which may vary in real-world IoT systems. Finally, validation was conducted in a controlled Python simulation environment; additional testing on physical IoT testbeds is required to confirm the real-world applicability. These limitations are elaborated in Section 6.1 (Limitations and Future Work).

7. CONCLUSION

This paper presented a hybrid consensus algorithm for real-time IoT blockchain networks that integrates PoV and PoR with Markov-based validator modeling and the M/M/1 queueing theory. The framework enhances validator trust, ensures dynamic transaction prioritization, and provides real-time responsiveness.

The simulation results confirm that the model achieves high throughput (∼1,996 tx/block), low latency (∼0.051 s), and high energy efficiency (8.46 Wh/block), making it well-suited for resource-constrained IoT environments. Furthermore, the security analysis demonstrated robustness against Sybil and Byzantine attacks while successfully detecting Eclipse attempts through dynamic trust reassignment. Compared with existing models (HB-IoT, HMM-Shard, and Microchain), the framework consistently showed superior latency, throughput, and energy sustainability.

7.1. Limitations and Future Work

Although the proposed hybrid consensus model demonstrates strong performance and security, several limitations remain.

1. **Scalability Constraints:**
The system has been validated on medium-sized networks, but has not yet been tested at extreme scales ($\geq 1{,}000$ nodes). Larger deployments may introduce new communication and coordination challenges that require additional mechanisms such as **network sharding** or hierarchical consensus layers.
2. **Parameter Sensitivity:**
Queuing performance depends on the accurate estimation of arrival (λ) and service (μ) rates. In real-world IoT settings, these parameters can fluctuate significantly because of traffic surges, device heterogeneity, or intermittent connectivity. This sensitivity may affect latency guarantees if it is not addressed dynamically.
3. **Simulation Scope:**
The current evaluation was based on a Python simulation environment. While this allows reproducibility and controlled experimentation, validation on **real-world IoT testbeds** (e.g., sensor networks and smart city deployments) is required to fully assess the robustness under practical operating conditions.

Future work will focus on:

- Extreme-scale evaluation: Integrating sharding and testing on networks exceeding 1,000 nodes.
- Adaptive queue management: Machine learning models are applied to dynamically tune queue parameters and enhance the responsiveness under varying loads.
- Advanced trust modeling: Extending validator behavior modeling with continuous-time Markov chains or Hidden Markov Models (HMMs) for finer predictive accuracy.
- Real-world IoT integration: Testing on permissioned blockchains (Hyperledger, IOTA) and deploying IoT hardware platforms with support for low-power cryptography.

By pursuing these directions, the framework can evolve into a fully intelligent, scalable, and resilient blockchain consensus system tailored for next-generation IoT networks.

REFERENCES

[1] S. Nakamoto, *Bitcoin: A peer-to-peer electronic cash system*, Bitcoin.org, 2008.

[2] C. S. Wright, "Bitcoin: A peer-to-peer electronic cash system," *SSRN Electron. J.*, 2008. DOI: 10.2139/ssrn.3440802.

[3] M. Xu, X. Chen, and G. Kou, "A systematic review of blockchain," *Financial Innov.*, vol. 5, no. 1, p. 27, 2019. DOI: 10.1186/s40854-019-0147-z.

[4] M. Conoscenti, A. Vetro, and J. C. D. Martin, "Blockchain for the internet of things: A systematic literature review," in *Proceedings of IEEE/ACS International Conference on Computer Systems and Applications (AICCSA)*, vol. 0, Jul. 2016. DOI: 10.1109/AICCSA.2016.7945805.

[5] A. Reyna, C. Martín, J. Chen, E. Soler, and M. Díaz, "On blockchain and its integration with iot. challenges and opportunities," *Future Gener. Comput. Syst.*, vol. 88, pp. 173–190, Nov. 2018. DOI: 10.1016/J.FUTURE.2018.05.046.

[6] A. A. Al-awamy, N. Al-shaibany, A. Sikora, and D. Welte, "Hybrid consensus mechanisms in blockchain: A comprehensive review," *Int. J. Intell. Syst.*, vol. 2025, no. 1, Jan. 2025. DOI: 10.1155/int/5821997.

[7] Q. L. Li, J. Y. Ma, Y. X. Chang, F. Q. Ma, and H. B. Yu, "Markov processes in blockchain systems," *Comput. Soc. Networks*, vol. 6, no. 1, pp. 1–28, Dec. 2019. DOI: 10.1186/S40649-019-0066-1.

[8] N. C. Z. Auhl, R. Alhadad, and W. Heyne, "A comparative study of consensus mechanisms in blockchain for iot networks," *Sensors*, vol. 23, no. 3, p. 1364, 2023. DOI: 10.3390/s23031364.

[9] J. Xi et al., "A blockchain dynamic sharding scheme based on hidden markov model in collaborative iot," *IEEE Internet Things J.*, vol. 10, no. 16, pp. 14 896–14 907, Aug. 2023. DOI: 10.1109/JIOT.2023.3294234.

[10] S. Li, L. D. Xu, and S. Zhao, "The internet of things: A survey," *Inf. Syst. Front.*, vol. 17, no. 2, pp. 243–259, Apr. 2015. DOI: 10.1007/s10796-014-9492-7.

[11] K. Christidis and M. Devetsikiotis, "Blockchains and smart contracts for the internet of things," *IEEE Access*, 2016. DOI: 10.1109/ACCESS.2016.2566339.

[12] S. M. S. S. Golder, S. Das, R. Bose, S. Sutradhar, and H. Mondal, "Hybrid blockchain framework for secure and scalable internet of things (iot) networks (hb-iot): A novel approach," *Sensors*, vol. 23, no. 3, p. 1364, 2023. DOI: 10.3390/s23031364.

[13] R. Xu, Y. Chen, E. Blasch, and G. Chen, *Microchain: A hybrid consensus mechanism for lightweight distributed ledger for iot*, Sep. 2019.

[14] O. Novo, "Scalable access management in iot using blockchain: A performance evaluation," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4694–4701, 2019. DOI: 10.1109/JIOT.2018.2879679.

[15] S. Brotsis, K. Limniotis, G. Bendiab, N. Kolokotronis, and S. Shiaeles, "On the suitability of blockchain platforms for iot applications: Architectures, security, privacy, and performance," *Comput. Networks*, vol. 191, p. 108 005, 2021. DOI: 10.1016/j.comnet.2021.108005.

[16] C. Nartey et al., "On blockchain and iot integration platforms: Current implementation challenges and future perspectives," *Wirel. Commun. Mob. Comput.*, vol. 2021, 2021. DOI: 10.1155/2021/6672482.

[17] M. A. Azad, S. Abdullah, J. Arshad, H. Lallie, and Y. H. Ahmed, "Verify and trust: A multidimensional survey of zero-trust security in the age of iot," *Internet Things*, vol. 27, p. 101 227, 2024. DOI: 10.1016/j.iot.2024.101227.

[18] C. Yang et al., "Trends in the conduct and reporting of clinical prediction model development and validation: A systematic review," *J. Am. Med. Informatics Assoc.*, vol. 29, no. 5, pp. 983–989, 2022. DOI: 10.1093/jamia/ocac002.

[19] M. S. Al-Rakhami and M. Al-Mashari, "A blockchain-based trust model for the internet of things supply chain management," *Sensors (Basel)*, vol. 21, no. 5, 2021. DOI: 10.3390/s21051759.

[20] S. Tanwar, A. Parmar, A. Kumari, N. K. Jadav, W.-C. Hong, and R. Sharma, "Blockchain adoption to secure the food industry: Opportunities and challenges," *Sustainability*, vol. 14, no. 12, p. 7036, 2022. DOI: 10.3390/su14127036.

[21] J. F. Shortle, J. M. Thompson, D. Gross, and C. M. Harris, *Fundamentals of Queueing Theory: Fifth Edition*. John Wiley & Sons, Sep. 2017, pp. 1–548. DOI: 10 . 1002 / 9781119453765.

[22] E. A. Alawamy, Y. Liu, and Y. Q. Zhao, "Bayesian analysis for single-server markovian queues based on the no-u-turn sampler," *Commun. Stat. - Simul. Comput.*, vol. 53, no. 2, pp. 658–670, 2024. DOI: 10.1080/03610918.2022.2025841.

[23] S. M. S. S. Golder, S. Das, R. Bose, S. Sutradhar, and H. Mondal, "Hybrid blockchain framework for secure and scalable internet of things (iot) networks (hb-iot): A novel approach," *Sensors*, vol. 23, no. 3, p. 1364, 2023. DOI: 10.3390/s23031364.

Appendix

Appendix A: Proofs of Lemmas and Theorems

Proof of Lemma 1 (Queue Stability Condition).
From the classical M/M/1 queuing theory [Ref. According to queueTheory], if the arrival rate exceeds or equals the service rate $\lambda \geq \mu$, the queue grows indefinitely, and the system becomes unstable.
Thus, to ensure a finite expected queue length and a bounded transaction waiting time,

$$\lim_{t\to\infty} E(Q(t)) < \infty \quad if\ and\ only\ if\ \lambda_{total} < \mu$$

where Q(t) is the queue length at time t. ■
Under this condition, average waiting times are determined as:

$$W_{high} = \frac{1}{\mu - \lambda_{high}},$$

$$W_{low} = \frac{1}{\mu - (\lambda_{high} + \lambda_{low})}$$

with the additional performance guarantee.
Proof of Lemma 2 (Priority Service Reduces Expected Waiting Time).
In a priority queue, high-priority jobs are served first, whenever possible, leading to

$$W_{high} = \frac{1}{\mu - \lambda_{high}},$$

$$W_{low} = \frac{1}{\mu - (\lambda_{high} + \lambda_{low})}$$

Since $\lambda_{high} + \lambda_{low} > \lambda_{high}$, it follows that:

$$W_{low} > W_{high}$$

Thus,high-priority transactions experience shorter waiting times on average. ■

Proof of Lemma 3 (Stationary Distribution).
From the Markov chain theory [Ref. MarkovTheory] states that any finite, irreducible, and aperiodic Markov chain admits a unique stationary distribution.
Because every validator can eventually reach any other state (good, average, poor), and state transitions occur probabilistically without deterministic cycles, the chain is

- Finite (3 states),
- Irreducible (positive probability transitions between states),
- Aperiodic (self-transitions are possible).

Hence, a unique stationary distribution π exists that describes the long-term behavior of the validator reputation system. ■

Proof of Theorem 1 (Reliability Convergence).
Given that the validator reputation system forms a finite-state, irreducible, aperiodic Markov chain (by Lemma 3), classical Markov chain theory guarantees the existence and uniqueness of a stationary distribution μ.
Thus, as the number of validation cycles $n \to \infty$, the probability of a validator being in any state converges to a stationary value. In particular, since the "Good" state is reachable and not absorbing, we have:

$$\mu_G > 0$$

This guarantees that in the long run, a positive proportion of validators will consistently maintain a trustworthy (good) state.■
Thus, the proposed validator reputation mechanism guarantees the long-term availability of a sufficient pool of trustworthy validators, ensuring continued reliability and security of the blockchain network.