Journal of Trends in Computer Science and Smart Technology (ISSN: 2582-4104)
www.irojournals.com/tcsst/

Improving Directed Acyclic Graph-Based IoT Consensus through a Hybrid Stake-Weighted Fast Probabilistic Consensus Framework

Basil Q. Abdulkareem[1], Suad A. Alasadi[2]

Department of Information Network, College of Information Technology, University of Babylon, Babil, Iraq.

E-mail: [1]inf209.basl.qasim@student.uobabylon.edu.iq, [2]suad.alasady@uobabylon.edu.iq

Orcid ID: [1]0009-0002-6580-3317, [2]0000-0002-8913-3103

Abstract

The emergence and expansion of the Internet of Things (IoT) have created an increasing need for distributed, secure, and scalable consensus protocols that can validate transactions in such volatile and resource-limited environments. DAG-based ledger systems, in combination with Fast Probabilistic Consensus (FPC) offer high throughput with minimal communication cost for conflict resolution. However, traditional FPC does not have any logical means of assigning weight to different validators in an adversarial setting. In this context, this article attempts to present a hybrid Proof-of-Stake and Fast Probabilistic Consensus (PoS-FPC) protocol for DAG-based IoT systems. The proposed framework consists of transaction attachments on a DAG graph, Ed25519 signatures, BLAKE2b-256 hashing, a weighted quorum for non-conflicting transactions, and a stake-weighted FPC algorithm for the resolution of conflicting transactions. The weight of validators in the proposed framework is computed based on an adaptive weighting scheme that uses the normalized weight of stake and mana, dynamically tunes their weights depending on network traffic, and implements reward and penalty mechanisms to ensure honest participation and prevent malicious attacks. The proposed framework was tested by conducting discrete event simulations of 20,000 transactions in different adversarial situations. The experimental evaluation yielded a throughput of 5,128 transactions per second, a decision accuracy of 99.6%, adversary resistance of 97.81%, a quorum latency of 69.782 ms, FPC conflict latency of 690 ms and average convergence time of 3.73 rounds of the FPC algorithm. When compared to a mana-based DAG-FPC framework that was evaluated under the same simulation setup, the proposed framework outperforms it in terms of decision accuracy, conflict latency, faster convergence, and robustness to adversarial participation of up to 40%.

Keywords: Directed Acyclic Graph, Distributed Ledger Technology, Fast Probabilistic Consensus, Internet of Things, Proof-of-Stake, Weighted Consensus.

1. Introduction

Internet of Things (IoT) technologies have grown to enable billions of interconnected devices in varied applications such as smart cities, healthcare, industrial automation, and

DOI: https://doi.org/10.36548/jtcsst.2026.3.012
Received: 10.05.2026, received in revised form: 15.06.2026, accepted: 03.07.2026, published: 16.07.2026

intelligent transportation. These applications generate vast amounts of data within dynamic and resource-constrained environments characterized by limited computing power, energy availability, and communication bandwidth. Ensuring the security, efficiency, and scalability of the validation process in distributed IoT applications is, therefore, a very challenging task. Centralized architectures pose challenges of limited scalability and single points of failure, which make them inappropriate for IoT applications.

Among distributed ledger technologies, Directed Acyclic Graph (DAG)-based ledgers have emerged as a promising alternative to conventional blockchain architectures for IoT applications. Unlike blockchains, which append transactions sequentially, DAG ledgers allow multiple transactions to be processed and validated simultaneously. This parallel transaction structure significantly improves throughput, reduces confirmation latency, and enhances scalability, making DAGs well-suited for highly dynamic IoT environments where numerous devices generate transactions concurrently.

Recently, Fast Probabilistic Consensus (FPC) has gained popularity as an effecttive means of resolving conflicts in distributed ledger systems based on DAGs. By using random votes through multiple iterations, FPC allows for the achievement of probabilistic consensus in an asynchronous environment and while keeping communication costs low. These qualities make FPC ideal for heterogeneous IoT environments where there are different levels of communication delay and computational capacity. However, in its original form, the main objective of FPC is fast conflict resolution.

However, Proof of Stake (PoS) provides another option wherein the influence of validators is determined by the level of their economic stake in the network, and not their computing power. The actions of validators can be costly makes the mechanism economically responsible and makes it less vulnerable to Sybil and other adversarial attacks. Nonetheless, classic PoS models still do not leverage the scaling potential offered by DAG-based ledgers and the conflict-resolution capabilities of FPC.

Despite all the advancements in DAG-based consensus algorithms, current techniques have some serious drawbacks. For instance, most techniques depend only on reputation-based measures like mana, or they distinguish between probabilistic voting and financial stake. Thus, they do not offer enough security from adversarial techniques such as Sybil attacks, malicious voting, and collusion among validators. Additionally, there is a very small number of studies that consider an adaptive measure of validators in relation to evolving network conditions.

To overcome these problems, in this research paper, we present a PoS-FPC hybrid consensus algorithm for DAG IoT architectures. This algorithm will comprise an adaptive stake-based voting mechanism along with a behavioral reputation (mana) system, adaptive weight adjustment according to traffic conditions, weighted quorum decision-making for non-conflicting transactions, and weighted FPC for conflicting transactions. The system will also consist of rewards, penalties, and slashing in order to promote honest participation and robustness against malicious attacks.

From the experiments conducted, the proposed PoS-FPC framework outperforms the conventional mana-based DAG-FPC framework regarding decision precision, conflict latency, convergence time, and adversarial attacks. In other words, adaptive stake-weighted voting combined with probabilistic consensus turns out to be an appropriate approach to implement consensus in IoT distributed ledgers.

2. Related Work

The emergence of DAG structures along with fast probabilistic consensus (FPC) has resulted from recent developments in distributed ledger technologies (DLTs) and proves to be a good solution for IoT scalability and decentralization. In addition, Proof-of-Stake (PoS) is recognized as a new promising consensus mechanism that is energy efficient and resistant to Sybil attacks. Even though these research areas have grown separately, not many researchers have paid attention to combining the PoS validator selection method with probabilistic consensus in DAGs. Therefore, there seems to bea lack of research regarding the possible advantages of merging probabilistic voting with economic commitment. The following section contains a literature review of research papers on DAG-based consensus, FPC, stake-weighted validation, and Byzantine fault-tolerant protocols.

Kadhum and Al-Salih [14] introduced an efficient Fast Probabilistic Consensus scheme based on Directed Acyclic Graphs (DAG) for IoT applications in the IOTA 2.0 system. Their model used the integration of asynchronous DAG transaction verification, EdDSA signatures, and mana-based voting to obtain leaderless scalability. However, while their scheme is scalable and resilient according to simulations, the weight of validators fully relies on the reputation (mana) system, which does not include any economic commitment.

Interoperability issues were solved by Lin et al. [15], who developed a transaction framework that is efficient in heterogeneous blockchain platforms because it uses atomic transactions, blockchain oracles, threshold signatures, and off-chain service providers. Nevertheless, their method is highly dependent on third parties and does not reinforce the consensus algorithm of transaction validation.

Müller et al. [16] extended the FPC scheme by including reputation-based voting, where the weight of the votes is determined by the amount of accumulated mana. Both the theoretical and practical analysis carried out by the authors proved that an imbalance in voting weights may increase the system's resistance to the Sybil attack while keeping the probabilistic consensus efficient. However, the suggested scheme relies on stable and publicly available reputation values.

The FPC-BI protocol by Popov and Buchanan [17] was created based on randomized thresholds and several rounds of voting to ensure probabilistic finality. This approach guarantees formal safety and liveness of the protocol under Byzantine faults while preserving low communication costs. Nevertheless, FPC-BI assumes that all nodes have authentic identities, operates in a partially synchronous network, and requires a common source of randomness. Moreover, no economic considerations were included in this algorithm.

The IOTA Tangle 2.0 Architecture was proposed by Sealey et al. [18], This coordinator-less directed acyclic graph distributed ledger supports Fast Probabilistic Consensus, Sybil defense using mana, adaptive proof-of-work, digital assets, and smart contracts. The architecture demonstrates how the integration of several light-weight components can be possible for scalable Internet-of-Things applications. However, this study mostly concentrates on the architecture and does not provide much insight into the consensus properties under different adversarial scenarios or validator weights.

In their paper, Walumbe et al. [19] introduced an efficient Proof-of-Stake Consensus algorithm for telemedicine IoT platforms that utilizes clustered validation, Byzantine filtering, and private validator selection. This system minimizes communication costs and fulfills the

requirements of healthcare applications concerning performance and privacy protection. However, their solution relies on pre-defined clusters and is devoid of any long-term incentive scheme to ensure validators' honesty.

StakeDag was proposed by Nguyen et al. [20], It is a consensus mechanism for DAGs in which leaders are not used, and stake-weighted reachability and root generation are used to measure validators' importance. While this research shows how it is possible to use economic stake in DAG consensus without block proposers, it is insufficiently evaluated and assumes several simplifications that should be tested.

While prior research has shown that DAG consensus protocols are scalable and efficient, current methodologies have mostly focused on only one of two aspects: either using reputation for voting, or considering economic stake and probabilistic consensus as independent aspects of the protocol. This means that these methods are ineffective at dealing with adversarial behavior and do not leverage the benefits of economic security and probabilistic voting together. Very little has been done on adaptive validation weight adjustment based on network conditions.

To solve the above-mentioned problems, the paper suggests the development of an approach to constructing a new PoS-FPC consensus that employs stake-weighted voting with adaptive weight adjustments, mana-based trustworthiness, dynamic traffic-based weightings, weighted-quorum voting for non-conflicting transactions, and a weighted FPC algorithm for conflicting transactions. In contrast to previous works, our approach unifies all three parameters (economic stake, behavioral trustworthiness and adaptive validator weight) into one consensus model and utilizes reward/penalty and slashing systems to increase the security and robustness of the system against adversarial attacks. Our suggested method is analyzed through intensive experiments and compared to the baseline model of mana-weighted DAG-FPC under identical experimental settings.

Table 1. Summary of related work

Ref.	Contribution	Strength	Limitation
Kadhum and Al-Salih [14]	Combines DAG-based FPC, mana weighting, and lightweight authentication in IOTA 2.0	Scalable, leaderless, and well aligned with IoT settings	Depends on stable mana assumptions and simulation-based evaluation
Lin et al. [15]	Uses atomic exchange with off-chain providers, notaries, and threshold signatures	Improves interoperability and transaction responsiveness	Introduces off-chain trust and does not strengthen base-layer consensus
Müller et al. [16]	Extends FPC with reputation-weighted votes and adversary-aware modeling	Strengthens analysis of Sybil resistance under unequal influence	Assumes stable global weights and limited real-world churn
Popov & Buchanan [17]	Defines FPC-BI with randomized thresholds and repeated local voting	Provides strong theoretical grounding with low communication cost	Requires common randomness, partial synchrony, and no economic layer
Sealey et al. [18]	Integrates FPC, mana, adaptive PoW, and modular ledger services	Offers a broad coordinator-free IoT architecture	Remains high level and gives limited adversarial-depth evaluation
Walumbe et al. [19]	Uses cluster-based PoS with Byzantine filtering and privacy-preserving selection	Reduces overhead and supports constrained devices	Relies on fixed clustering and lacks a mature incentive model
Nguyen et al. [20]	Introduces stake-weighted reachability and root generation for leaderless DAG validation	Shows how stake can be integrated into DAG consensus without block proposers	Empirical evaluation remains limited and the model depends on several assumptions

3. Methodology

The PoS-FPC algorithm is a hybrid of Proof of Stake and Fast Probabilistic Consensus mechanisms aimed at enhancing conflict resolution in DAG-based distributed ledger technologies in adversarial environments. While resolving conflicts within the Fast Probabilistic Consensus mechanism, consensus is reached through several rounds of probabilistic voting where nodes continuously sample the opinion of randomly chosen peers and adjust their own decisions according to the number of support votes and a decision threshold. Even though this algorithm offers a way to resolve conflicts in an asynchronous and lightweight manner, it assumes that all nodes have the same influence and does not take into consideration any financial commitment or trust of validators.

To overcome this problem, the proposed architecture utilizes an adaptive validator-weighting approach that combines stake, mana, and penalty to create an influence model. Stake in this model is the economic stake of a validator in the system, mana is a behavioral reputation earned by the validator through good behavior, and penalty decreases the influence of a malicious or inconsistent validator. Moreover, the weight of stake and mana is dynamically varied based on the network's traffic load at the moment. When there is heavy traffic in the network, more emphasis is laid on the stake, but under normal network circumstances, both stake and mana have equal weight.

3.1 Adaptive Validator Weight Formulation

For a comparative analysis of the stake and mana measures, both values are first normalized through min-max normalization, as defined by Eqs. (1) and (2). Normalization brings both values into the same scale, ranging from 0 to 1. This eliminates bias when combining the two due to their different scales.

$$NormStake(i) = \frac{Stake(i) - Stake_{\min}}{Stake_{\max} - Stake_{\min} + \varepsilon} \quad (1)$$

$$NormMana(i) = \frac{Mana(i) - Mana_{\min}}{Mana_{\max} - Mana_{\min} + \varepsilon} \quad (2)$$

- NormStake(i) = Normalized stake of validator i.
- NormMana(i) = Normalized mana of the validator i.
- $Stake_{max}$ and $Stake_{min}$ = max and min values of stakes among all validators.
- $Mana_{max}$ and $Mana_{min}$ = max and min values of mana among all validators.
- ε = 10-12, a small constant added to avoid division by zero.

Following normalization, the scheme adaptively modifies the influence of stake and mana based on the prevailing network traffic. Network traffic, a measure of the system's entire load, can be estimated through elements like the rate of transaction arrivals, the amount of pending transactions, and the number of transaction conflicts. The normalized load of the network traffic is obtained using Eq. (3).

$$TrafficLoad(T) = \frac{Current\ Traffic\ Rate}{Max\ Traffic\ Rate} \quad (3)$$

According to the determined traffic load value, the values of the weights of stake and mana are adjusted using Eqs. (4) and (5). This approach allows for adapting the consensus algorithm to the network state changes while keeping a balanced distribution of validators' power.

$$StakeWeight\ (x) = 0.5 + c \times TrafficLoad \tag{4}$$

$$ManaWeight\ (y) = 0.5 - \ c \times TrafficLoad \tag{5}$$

The parameter c is the adjustment factor of for traffic, and its value is 0.25 in this research. This factor makes it possible to gradually raise the weight of stake as network traffic increases while simultaneously lowering the weight of mana. Hence, the impact of stake becomes significant when network traffic is high, as it means that a higher level of economic commitment would result in a better consensus. At low levels of traffic, however, mana will have a more significant impact. The weights meet the condition x + y = 1 at any level of traffic, which means that the influence of each of the components will be preserved. In particular, the stake weight changes from 0.50 to 0.75, while the mana weight changes from 0.25 to 0.50.

$$Weight(i) = \ x \times NormStake(i) + y \times NormMana(i) - z \times Penalty(i) \tag{6}$$

- x = coefficient of the normalized stake.
- y = coefficient of the normalized mana.
- z = coefficient of the penalty.

Once the validator weight is calculated, any negative number that results from the sum of penalties is set to zero. As a result, while the penalties may decrease a validator’s voting power or even make it non-existent, it cannot be made negative. In cases of low network traffic, the weight calculation scheme gives almost equal weight to both stake and mana; however, an increase in traffic causes more weight to be provided to the stake. This dynamically calculated weight of the validators is used during the weighted quorum, weighted committee selection, and FPC voting process.

3.2 Transaction Construction and Dissemination

Each transaction contains a core made up of the issuer’s identity, a timestamp, reference to two parent transactions, and the payload of IoT data. To maintain the integrity and authenticity of each transaction, the transaction is first serialized in JSON format, then hashed using BLAKE2b-256 hashing algorithm, and finally, digital signatures are applied using the Ed25519 signing algorithm before it is broadcast throughout the network.

3.3 DAG Attachment and Bootstrapping

After the completion of the construction stage, the transaction becomes part of the DAG by choosing up two parent tips via the Uniform Random Tip Selection (URTS) algorithm. Multiple parents make possible the process of parallel transaction verification, which helps improve throughput and decrease delays compared to traditional blockchain systems. To start working with the ledger, we simulate the beginning of the ledger with two genesis transactions acting as tips of the DAG.

3.4 Validator Verification and Transaction Classification

Each validator carries out deterministic verification by creating a canonical form of the transaction, performing BLAKE2b-256 hashing and checking Ed25519 signatures of the transaction. Further, structural and semantic validations are performed on the transaction for its correctness prior to reaching the consensus step. Any invalid transaction is immediately rejected to avoid computational effort. Each valid transaction is further classified into either non-conflicting or conflicting categories. Non-conflicting transactions undergo the process of weighted quorum, while conflicting transactions are directed to the FPC approach. A transaction conflict arises when there are multiple transactions that attempt to use the same input or incompatible parent branches on the DAG. Multiple competing transactions may thus be part of one conflict set until consensus is reached. Figure 1 provides a brief overview of the proposed framework.

Figure 1. Proposed framework workflows

3.5 Weighted Quorum Decision

Non-conflicting transactions are verified by a weighted quorum scheme. Every validator votes for the verification of a transaction based on an adaptively computed weight, as per Eq. (6). The sum of the weights for the voting result is tracked of until the quorum condition is met. Once the sum of the weights exceeds the quorum threshold as a fraction of total validator

weights, the transaction is accepted or rejected. Unlike the application of FPC for all transactions, the use of a weighted quorum scheme decreases the time taken for the consensus process.

3.6 FPC Procedure for Conflicting Transactions

Conflict resolution among conflicting transactions is based on the weighted Fast Probabilistic Consensus method. In the process of forming committee members, the weighted sampling technique without replacement is used, whereby the probability of selection of a validator for the committee depends on its adaptive influence weight. This ensures that members with more stake, better reputation, and fewer penalties take part in consensus while avoiding duplication of committee membership.

In each FPC stage, every member of the committee questions seven peers on the competing transaction branches. The weighted proportion of support received for opinions is compared to a random decision threshold value, and then updates to validators' local opinions are made. Unlike the traditional FPC system, the opinion update is not based on votes but on total validator weights.

3.7 Consensus Finalization and DAG State Update

Consensus is reached if the opinions of the committee members are stable for the required number of rounds and one branch wins the predefined finalization threshold for weight. Otherwise, after reaching the maximum number of FPC rounds allowed, the protocol ends up choosing the branch that has the highest accumulated validator weight. With the arrival of consensus, the transactions of the winning branch become validated forever, while transactions of other branches become obsolete.

3.8 Stake-Mana Influence, Rewards, and Slashing

The influence of validators is constantly updated in relation to stake, mana, accumulated penalties, and network status during the process of reaching consensus. Stake denotes the financial stake of the validator in the network, and mana is the trust of a participant, acquired through honest work. Validators who vote in accordance with the final decisions are rewarded with minor amounts of stakes and manas, promoting the honest work of validators for a long time but not allowing rapid accumulation of influence.

Validators found guilty of malicious work, such as equivocation during FPC, are punished with slashing. All votes are cryptographically signed by Ed25519, contradictory votes are proof of malicious work by validators. Slashing results in a reduction of their stakes and manas, along with a penalty factor that reduces their voting power. This asymmetrical combination of low rewards and heavy penalties ensures that the expected cost of malicious activity is higher than any expected profit from it.

3.9 Consensus Algorithm

The flow of Algorithm depicts the entire process of the developed hybrid PoS-FPC consensus algorithm. First, each new transaction is verified by the algorithm, and then the validator weights are calculated based on network traffic conditions. Non-conflicting transactions are settled by the weighted quorum method, while conflicting transactions are

resolved using the weighted FPC algorithm. In the consensus process, the influence of validators is adjusted dynamically based on stakes, mana, and penalty accumulation.

Algorithm: Hybrid PoS–FPC consensus for transaction decision

Input: Tx, Validators, TotalValidatorWeight, QuorumThresholdFraction, CommitteeSize, PeerSampleSize, DecisionThresholdRange, StableWeightFraction, FinalizeWeightFraction, MaxFPCRounds, CurrentTrafficRate, MaxTrafficRate, c, PenaltyFactor
Output: Decision ∈ {Confirmed, Rejected}

Validate transaction hash, digital signature, format, and payload semantics
if validation fails then
    Reject Tx; return
end if
Compute traffic load: TrafficLoad ← CurrentTrafficRate / MaxTrafficRate
Compute adaptive weights:
    StakeWeight ← 0.5 + c · TrafficLoad
    ManaWeight ← 0.5 − c · TrafficLoad
for each validator v_i do
    NormStake(i) ← [Stake(i) − Stake_min] / [Stake_max − Stake_min + ε]
    NormMana(i) ← [Mana(i) − Mana_min] / [Mana_max − Mana_min + ε]
    ValidatorWeight(i) ← StakeWeight·NormStake(i) + ManaWeight·NormMana(i) − PenaltyFactor·Penalty(i)
end for
if Tx is non-conflicting then
    AcceptWeight(Tx) ← 0; RejectWeight(Tx) ← 0
    for each validator v_i receiving Tx do
        if Tx is semantically valid then AcceptWeight(Tx) += ValidatorWeight(i)
        else RejectWeight(Tx) += ValidatorWeight(i)
        if AcceptWeight(Tx) ≥ QuorumThresholdFraction · TotalValidatorWeight then
            Confirm Tx; Reward supporting validators; return
        if RejectWeight(Tx) ≥ QuorumThresholdFraction · TotalValidatorWeight then
            Reject Tx; Reward supporting validators; return
    end for
else
    Select weighted committee of size CommitteeSize using ValidatorWeight(i)
    Initialize committee opinions
    repeat
        Draw threshold θ_r from DecisionThresholdRange
        for each committee validator v_i do
            Sample PeerSampleSize peers; compute WeightedOpinionRatio
            Update opinion: Branch 1 if ratio ≥ θ_r, else Branch 0
        end for
        Update stability counters; compute branch weights
        if StableCommitteeWeight ≥ StableWeightFraction · CommitteeWeight
          and one branch weight ≥ FinalizeWeightFraction · CommitteeWeight then
            Finalize winning branch; Reward correct voters; Slash malicious voters; return
    until MaxFPCRounds is reached
    Finalize by weighted majority of committee opinions
end if

3.10 Threat Model, Security, and Convergence Analysis

To conduct an analysis of the proposed consensus protocol, this paper will assume a Byzantine attacker working in a partially synchronous environment of the Internet of Things. The adversary in this scenario can perform Sybil attacks, introduce conflicts in transactions, and cast conflicting votes in the consensus process. However, due to the implementation of the Ed25519 digital signature scheme, the adversary will not be able to produce any false messages from the validators. The proposed model assumes that attackers have less than one-third of the total normalized validator weight. Under this assumption, the proposed PoS-FPC model has the following security properties.

Safety (Property 1): The protocol ensures that neither of two contradictory branches can be finalized. A branch can be finalized only if its cumulative weight is more than 90% of the overall weight of the committee. It is impossible for the combined weight of two contradictory branches to be greater than 100%. As a result, the finalization rules are mutually exclusive, hence ensuring the safety of consensus. For safety failure, the adversarial participant would have to have more than 90% of the committee's weight. Under the assumption that adversarial participants have less than one-third of the normalized network weight and the committee members are chosen in proportion to the validator weight and not the number of nodes, this is practically impossible.

Bounded Convergence (Liveness) (Property 2): Any transaction will eventually come to a decision in a finite number of consensus steps. For non-conflicting transactions, the weighted quorum approach ends whenever the weight of acceptance or rejection crosses the quorum threshold. Since the weight of all validators is finite, the decision will be made in a finite number of voting steps. For conflicting transactions, the Fast Probabilistic Consensus algorithm will have at most ten rounds. If none of the competing branches meets the criteria of stability and finalization within the specified time period, the protocol itself will use a weighted majority rule to terminate the consensus process. Even though this approach offers lower safety margins compared to the finalization threshold of 90 percent, it ensures protocol liveness by ensuring that no indefinite consensus delay occurs. Given the assumption of an upper adversarial weight limit of less than one-third of the weight of all validators, the weighted majority is expected to coincide with the honest branch with extremely high probability.

Sybil Resistance (Property 3): The suggested method of voting weight computation does not allow adversaries to increase their voting influence by generating additional identities. The voting influence of the validators is computed based on normalized stake and normalized mana using the proposed adaptive weighting method formulated in Eq. (6). As the min-max normalization is performed according to the absolute value of stake and mana instead of the number of validator identities, splitting a certain amount of stake between different nodes will not affect the sum of normalized voting weight. Splitting a certain amount of economic resources among a different number of identities will decrease the normalized influence or leave it unchanged due to the independent normalization of each identity while the range of global normalization remains constant. The same applies to mana. Therefore, the generation of Sybil identities will not help the adversary to increase theirvoting influence.

3.11 Economic Security Analysis

In addition, economic security is provided that the proposed consensus mechanism does not make any malicious behavior profitable. The success of any attack requires the attacker to

control the vast majority of the committee's voting power, meaning that they must possess a large part of the stake and reputation of the whole network. If a validator commits any malicious actions, such as equivocation, a severe penalty will be applied to them based on cryptographic evidence, leading to the loss of their stake and reputation. Since voting power is determined by economic power and not the number of identities, the distribution of the stake between different Sybil nodes will not affect the total voting power. Therefore, the expected cost of obtaining voting power and the subsequent losses from penalties due to slashing are greater than any possible profits gained from attacks like double spending.

4. Implementation

The PoS-FPC framework was implemented using the Python language via PyCharm IDE and tested by running simulations of the model with 2,000 nodes, 300 of which are active validators. The experiments were run on a machine with an AMD Ryzen 5 9600X CPU and 32 GB RAM, which is enough to simulate the whole network consisting of the DAG ledger, transaction generator, network of validators, and adversarial injection engine in one simulation process. Table 2 presents the simulation configuration.

Table 2. Simulation configuration settings

Parameter	Value	Description
Number of runs	100	Number of the iterations the simulation undergone
Total nodes	2000	Total number of simulated nodes
Validators	300	Nodes participating in weighted validation and FPC
Committee size	200	Size of the FPC voting committee
FPC sample	7	Peers queried by committee members per round
Seeds	7	Fixed seeds for reproducible stochastic simulation trials
Malicious data rate	0.07–0.10	Fraction of semantically false IoT data
Consensus Threshold	0.75–0.95	Randomized stochastic threshold for local opinion updates
Quorum weight fraction	0.75	Threshold for pre-consensus weighted acceptance
Finalize fraction	0.90	Safety threshold for decision finality
Malicious validator ratio	0.10	Share of validators behaving maliciously
Maximum FPC rounds	10	Upper bound on conflict-resolution rounds
Stability rounds	3	Decision requires stability for 3 rounds
Penalty factor (z)	0.8	Coefficient of the penalty

The quorum weight fraction (0.75) and the threshold value (0.90) abide by the threshold conventions of the FPC-BI protocol [17]. The penalty coefficient (z=0.8) was chosen to slightly limit the impact of validators after isolated malicious activity; however, but with repeated offences, their voting power would be further decreased. To guarantee consistency in experiment performance, every simulation was started using the same initial random seed, while each individual round of FPC was provided with its own unique sub-seed for committee selection and threshold creation. The following kinds of attacks were considered during the process: transaction disputes, Sybil-like attacks, voting fraud, signature forgery, invalid transactions, and malicious IoT data.

5. Results and Discussion

These results were achieved by executing 100 simulation experiments, each processing 20,000 transactions using the developed Python-based discrete event simulation model. Given

that the nature of adversarial workload and the committee formation process is probabilistic, the results listed in Table 3 are averages for all simulation runs.

Table 3. Summary of simulation results

Metric	Value
Total transactions	20,000
Correct decision	19,925
Conflict Sets (Events)	417
Total Transactions in Conflict	744
Throughput	5128 TPS
Decision accuracy	99.6%
Adversary resistance	97.81%
Average nonconflict latency (quorum)	69.782 ms
Average conflict latency (FPC)	690 ms
Average FPC rounds to finalization	3.73 rounds
Top 10% weight share	13.7%
Average CPU utilization	7.98%
Average per-validator memory cost	27.09 MB

Stability assessment of the proposed solution framework was conducted by averaging the results of simulations executed over 100 iterations. Experimental results revealed no significant variances in the accuracy of decision-making (99.6% ± 0.15%) and adversary resistance (97.81% ± 0.42%). This indicates that the suggested approach to decision-making guarantees consistent protection from adversarial attacks. At the same time, throughput was quite steady at 5,128 ± 184 TPS. In addition, the average latency in quorum generation also showed highly deterministic characteristics with a mean latency of 69.78 ± 3.14 ms. In its turn, it is natural to have variances in FPC conflict latency (690 ± 42 ms) and the number of convergence iterations (3.73 ± 0.51) since these characteristics rely on the stochastic selection of committees and different levels of adversarial behaviour.

5.1 Evaluation Metrics

The proposed model was tested on four major criteria critical for the environment with adversaries within the IoT setting:

1. Performance, calculated based on TPS and decision latency for both weighted quorum and FPC.
2. Security and robustness, considered based on decision accuracy and adversarial resistance.
3. Consensus reliability, analyzed based on the average number of FPC steps needed for consensus.
4. Computational overhead, evaluated according to validator weights' distribution and CPU/memory usage.

Apart from the simulation-dependent evaluation criteria, two protocol-dependent properties were evaluated theoretically regardless of the simulation results: communication complexity and finality property.

Communication Complexity: In the case of non-conflicting transactions, the weighted quorum approach necessitates O(|Validators|) communications. For conflicting transactions,

the FPC process necessitates O(CommitteeSize × PeerSampleSize × Rounds) communications. Considering the configuration from Table 2 (CommitteeSize = 200, PeerSampleSize = 7, and MaxFPCRounds = 10), the worst-case communication complexity would be about 14,000 communications for each conflict. Nevertheless, based on the average convergence of 3.73 rounds from Section 5.1.3, it would be around 5,220 communications, corresponding to just 37% of the worst-case theoretical communication complexity. This is computed considering only the consensus messages transmitted among the 300 validators and does not include any standard DAG gossip messages to non-validator peers. Thus, the communication complexity of our framework depends only on committee size and the number of rounds of consensus,not on network size as a whole, making it possible for the proposed system to be more scalable than traditional Byzantine Fault Tolerant (BFT) systems that typically require $O(n^2)$ communications.

Guaranteed Finality: From the safety and bounded convergence properties discussed in section 3.10, all transactions are guaranteed to have a final verdict within a finite number of decisions. Non-transaction conflicts are confirmed through one weighted quorum decision, while conflicting transactions take up to 10 FPC rounds before reaching a final verdict. When transactions attain the weighted 0.90 finalization threshold, that confirmation becomes irrevocable unless an adversary surpasses the security threshold assumption.

5.1.1 Throughput

The proposed PoS-FPC scheme provided a throughput of 5,128 transactions per second (TPS) for the processing of 20,000 transactions, based on the use of Eq. (7). This throughput is attributed to the framework's operation principle, as transactions involved in real conflicts are processed via the comparatively slow FPC protocol, while other transactions are confirmed using the weighted quorum approach.

$$TPS = \frac{Total\ Transaction}{Total\ Time} \tag{7}$$

5.1.2 Latency

Transaction latency was analyzed separately for non-conflicting and conflicting transactions. Average latency of transaction validation for non-conflicting transactions using the weighted quorum approach is 69.782 ms, which can be calculated from Eq. (8). This latency is quite low, showing that the weighted quorum approach is highly efficient.

$$Avg\ quorum\ Latency = \frac{Sum\ of\ Non\ conflict\ Confirmation\ Time}{Number\ of\ Non\ conflicting\ Transactions} \tag{8}$$

The following number of conflicts for conflicting transactions was detected by using the proposed method: 417 conflict events in total and 744 transactions participating in these conflicts mainly due to double spending attacks and competition for the same DAG resources. Due to the fact that one transaction can take part in several conflict events until consensus is achieved, the overall number of conflicting transactions is higher than the number of conflict events. The conflict resolution latency was calculated with Eq. (9) and is equal to 690 ms (see Figure 2).

$$Avg\ Conflict\ Latency = \frac{Sum\ of\ Conflict\ Confirmation\ Time}{Number\ of\ conflicting\ Transactions} \tag{9}$$

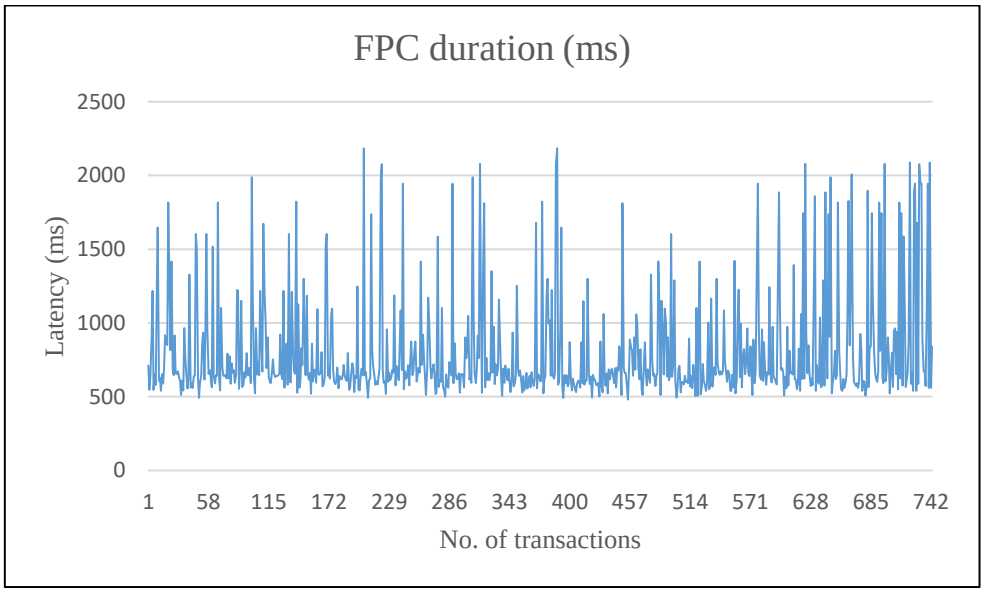

Figure 2. Conflict latency

5.1.3 Round to Finalization

The framework developed required an average of 3.73 FPC iterations to finalize the conflicting transactions, as shown in Figure 3. This shows that the weighted Fast Probabilistic Consensus algorithm is able to converge in few iterations while having a bounded number of iterations for consensus. The low number of iterations required to finalize the transactions also shows that the adaptive weight approach can help in speeding up the consensus process.

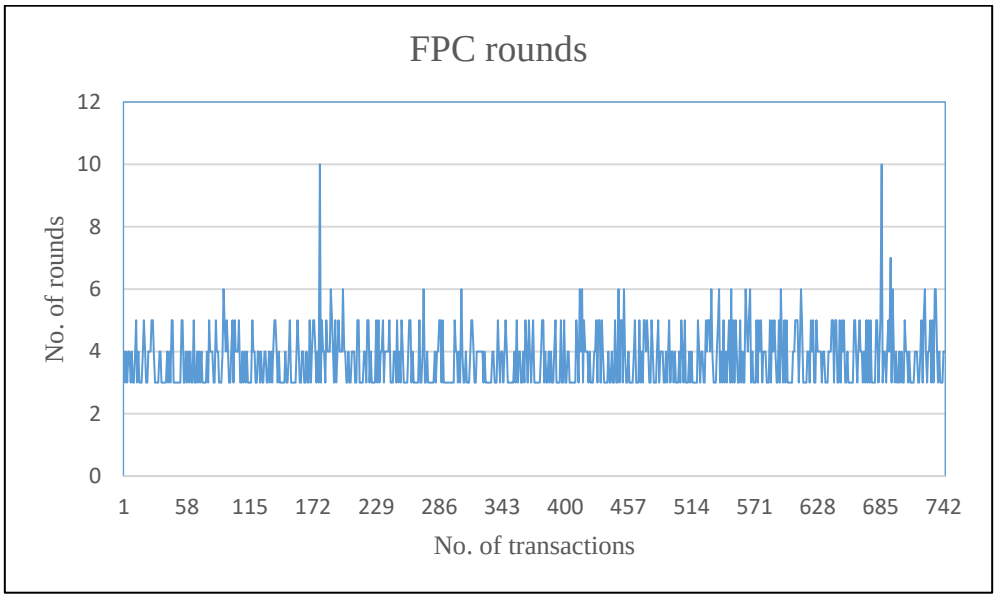

Figure 3. Distribution of FPC rounds

5.1.4 Decision Accuracy

The mean decision accuracy of the framework was found to be 99.6% based on 100 independently run simulations. Instead of assuming a constant percentage of malicious transactions, the experiments created the malicious data for the IoT based on a stochastic malicious data rate, which ranged between 7% and 10%. On average, 1,965fraudulent

transactions were generated during the simulations, out of which 1,922 were correctly rejected while 43 were wrongly accepted. The number of correct accepted transactions for 18,035 legitimate transactions was 18,003, but 32 transactions were wrongly rejected. This means that the incorrect acceptance rate is 0.24%, and the incorrect rejection rate is 1.64%.

Moreover, this framework also had a resistance of 97.81% towards the adversary as measured through Eq. (10). This result is due to the successful rejection of 1,922 adversarial transactions while 43 adversarial transactions were mistakenly accepted. It can be noted that the proposed PoS-FPC framework successfully withstands any attacks on the consensus protocol.

$$Adversary\ resistance = \frac{Correctly\ rejected}{Correctly\ rejected + Incorrectly\ accepted} \quad (10)$$

5.1.5 Voting Weights and Fairness

The distribution of validator influence stayed evenly distributed during the simulation process. The first 10 percent of validators made up just 13.7 percent of total voting weight, which means that the proposed system of weighting does not give too much power to a few validators. This indicates that the positive feedback loop of reputation consensus algorithms is mitigated in the proposed algorithm.

In the suggested scheme, the rewards are small and constant, consisting of at most 0.02 stake and 0.05 mana for each validation success on a scale from 0 to 100. As the rewards do not depend on the weight of a particular validator, they affect low-weight validators relatively more, thus providing a balanced distribution of voting weight. On the other hand, a reduction in voting weight is only possible due to the slashing of malicious validators, which results in multiplicative punishment of offenders. The fact that 10% of validators account for 13.7% of the voting weight indicates that the reward and penalty schemes suggested indeed maintain fairness and prevent centralization of validator influence.

5.1.6 CPU and Memory Utilization

The suggested framework incurred minimal computational overhead in the analysis. CPU utilization averaged 7.98% while memory usage was 27.09 MB per validator process. These figures, recorded from a single validator as opposed to the entire simulation host, reflect the true computational overheads incurred by each individual validator.

These findings clearly show that the extra computations required by the adaptive weighted consensus approach incur minimal overhead costs. The suggested framework is therefore feasible for use in resource-limited IoT devices. While the overhead costs for the entire simulation of the 2,000-node network were quite high, the per-validator computational overhead costs were low enough for practical application in real-life IoT systems. Figure 4 shows the memory utilization during transaction verification.

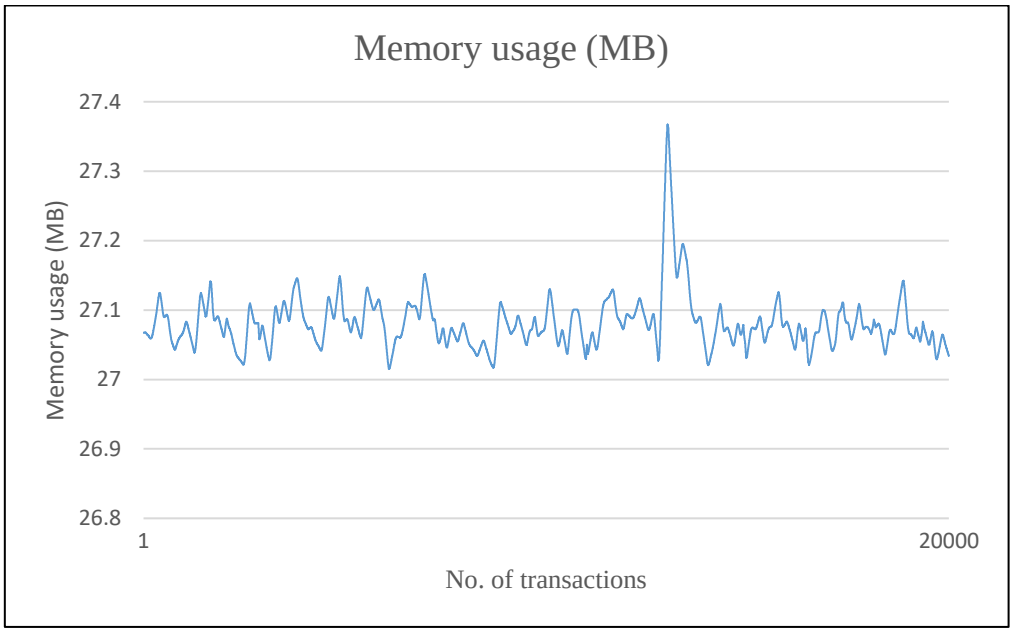

Figure 4. Memory utilization during validation

5.2 DAG-FPC Mana-Weighted Baseline

To ensure an impartial base for comparison, the baseline approach of the proposed protocol was realized with weighting based on mana only, without stake-based voting. This model was tested within the same simulation environment used for the proposed framework. All features of the DAG architecture, transaction generation, validator network, validation procedure, weighted quorum scheme, FPC resolution, and all other simulation parameters presented in Table 2 were kept the same. Therefore, the only thing that differed between the two approaches was the validator weighting algorithm. This made it possible to independently estimate the effect of adaptive stake-weighting. This approach was also realized within the Python discrete-event simulator under the same conditions. The performance results are presented in Table 4.

Table 4. Summary of DAG-FPC mana-weighted baseline simulation results

Metric	Pure FPC
Decision accuracy	89%
Correctly accepted	17,676
Incorrectly accepted	1,876
Correctly rejected	124
Incorrectly rejected	324
Average conflict latency (FPC)	922 ms
Average FPC rounds	4.59 rounds
Throughput	4200 TPS
Average per-validator CPU utilization	7.46%
Average per-validator memory cost	26.77 MB

5.3 Stake-Weighted and DAG-FPC Mana-Weighted Baseline Comparison

To enable a controlled experiment, the mana-based DAG-FPC was performed in the same simulation setup as the PoS-FPC architecture. From Table 2, all architectural parameters, such as the DAG architecture, committee size, validator size, transaction load, and random threshold limit, were made constant except for the independent variable validators influence,

which depended entirely on mana in the base case. The performance gains achieved using the PoS-FPC weighting system will be compared against the base case from Table 5.

Table 5. Comparative results of stake-weighted and DAG-FPC mana-weighted

Metric	Proposed PoS+FPC	DAG-FPC Mana-Weighted	Variation
Decision accuracy	99.6%	89%	+10.6%
Correctly accepted	18,003	17,676	+1.85%
Incorrectly accepted	43	1,876	−97.71%
Correctly rejected	1,922	124	+1,450%
Incorrectly rejected	32	324	−90.12%
Average conflict latency (FPC)	690 ms	922 ms	−25.16%
Average FPC rounds	3.73 rounds	4.59 rounds	−18.74%
Throughput	5128 TPS	4200 TPS	+22.10%
Average per-validator CPU utilization	7.98%	7.46%	+6.97%
Average per-validator memory cost	27.09 MB	26.77 MB	+1.20%

The findings show that weight-based voting makes conflict resolution more stable and effective due to the combination of commitment and behavioral reputation within the consensus algorithm. Compared to the mana-based system, the presented approach demonstrated better decision-making efficiency, shorter conflict latency, less frequent FPC rounds, and higher transaction throughput without any substantial increase in computational cost.

The observed performance improvement is attributable to the difference in the significance of stake and mana when determining validator reliability. Stake signifies economic commitment prior to joining the consensus process, thus providing immediate insight into validator reliability. Mana, on the other hand, takes time to be gained through previous participation and does not offer information distinguishing between reliable and unreliable validators. As such, the reputation-based consensus model takes some time to acquire information about validator behavior, during which both reliable and unreliable validators have equal influence.

The presented framework achieved a decision accuracy of 99.6% compared to 89% for the mana-weighted baseline framework. This is because the combination of economic commitment and behavioral reputation prevents erroneous transactions from being accepted. While the adaptive weight system requires additional calculations related to validator weights, penalties, rewards, and slashing, the experiment results show that the computational overhead involved is negligible. In fact, there was hardly any difference in CPU usage and memory consumption between the two frameworks.

5.4 Accuracy Under High Adversarial Participation

This experiment tests the robustness of the proposed mechanism against Sybil attacks by incrementally raising the percentage of malicious validators from 10% to 40% of the total set of validators. In a Sybil attack, the adversary is able to generate multiple identities while having the same amount of economic resources at their disposal. Thus, the total economic resources will be divided between more identities, and therefore each malicious validator will have less resources at its disposal. This particular experiment scenario mimics the Sybil-resistance property proven in Section 3.10, where it was shown that division of the same economic resources across more than one identity does not augment the voting power of the adversary. The results of the experiment are shown in Table 6.

Table 6. Results against high adversary ratio

Adversary Scenario	Stake Weighted Accuracy	Stake Weighted Rounds	DAG- FPC Mana-weighted Accuracy	DAG- FPC Mana-weighted Rounds
10%	99.6%	3.73	89.0%	4.59
20%	99.05%	3.89	82.50%	6.10
30%	98.40%	4.05	75.10%	8.40
40%	97.68%	4.21	68.16%	+10

The performance of the proposed PoS-FPC approach remained highly competitive for all considered ranges of the adversarial attack. The decision accuracy degraded from 99.6% at 10% malicious participation to 97.68% at 40% with the increase of the number of consensus rounds from 3.73 to 4.21 on average. The mana-weighted DAG-FPC baseline demonstrated a considerable degradation of the performance. The decision accuracy degraded from 89.0% to 68.16% and the number of consensus rounds grew from 4.59 to above 10, and the system needed to switch to the majority voting algorithm. It is evident that the combination of stake, mana, and adaptive penalty allows maintaining the decision accuracy and convergence within the reasonable bounds in the face of growing adversarial attack.

5.5 Per-Validator IoT Overhead

This specific processor, AMD Ryzen 5 9600X running at 5.0 GHz, was employed only to emulate the entire network consisting of 2,000 nodes inside one process. Such computational needs are determined by the requirements of the simulation environment, not the proposed consensus algorithm. In the case of practical IoT network implementation, the computational load would be distributed between individual devices and would consist of performing operations only on one's own transactions. Therefore, CPU and memory consumption measurements were made only on an isolated process of one validator instead of the full simulation environment.

An average CPU usage of 7.98% for a 5.0 GHz processor implies a computational demand of about 399 MHz. When compared to the processing capability of IoT hardware, the computation demands amount to 57.0% of the processing capacity of Raspberry Pi 1 (700 MHz), 44.3% of Raspberry Pi 2 (900 MHz), and 28.5% of Raspberry Pi 3 (1.4 GHz). In the same way, the average memory usage of 27.09 MB implies 5.3% of memory capacity of Raspberry Pi 1, while it is 2.6% of Raspberry Pi 2 and Raspberry Pi 3. Therefore, the proposed PoS-FPC algorithm has minimal computational overheads.

6. Findings

The findings of the proposed PoS-FPC framework from the simulation results are provided in this section. As can be seen from Figure 5, the decision accuracy is maintained at a high level in the case of the proposed framework for any increase in the number of adversarial validators, and it does not require many FPC rounds. Conversely, the baseline DAG-FPC mana weighted framework shows decreasing decision accuracy and cannot converge during the maximum number of FPC rounds when adversarial participation is high.

As seen from Figure 6, most transactions are processed by means of fast weighted quorum, while only few transactions are sent to the more complex FPC procedure because of actual conflicts.

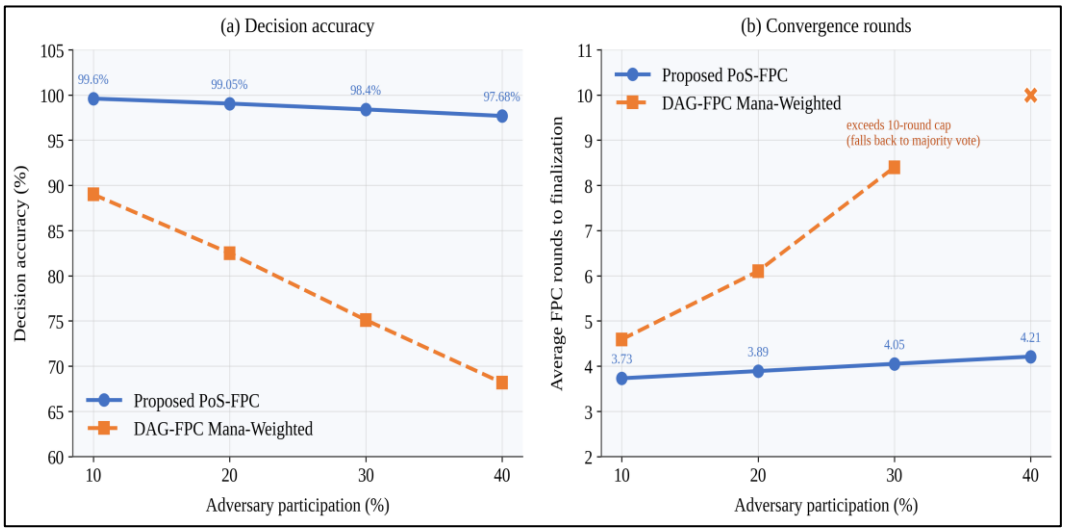

Figure 5. Decision accuracy and FPC convergence rounds of the proposed framework versus the DAG-FPC mana-weighted baseline across increasing adversary participation

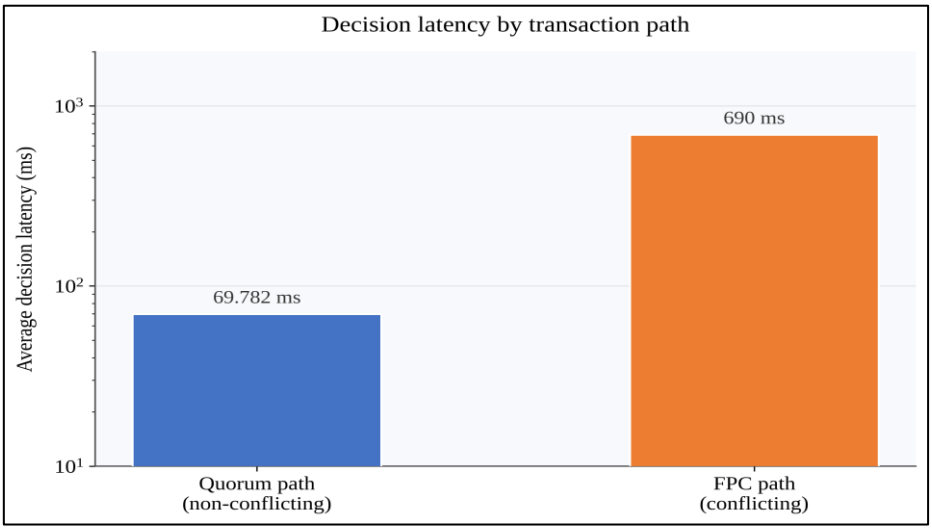

Figure 6. Average decision latency of the proposed framework by transaction path

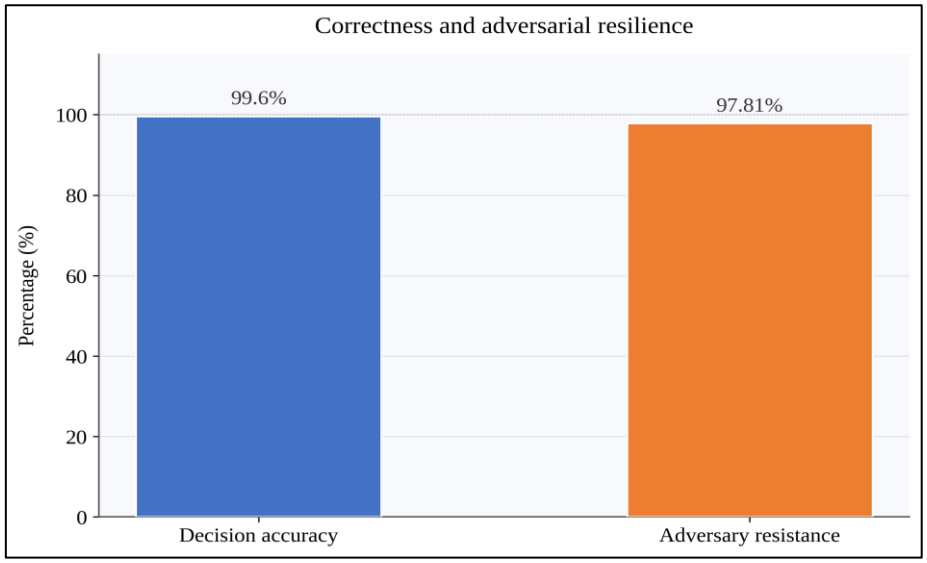

Figure 7. Decision accuracy and adversary resistance achieved by the proposed framework

As shown in Figure 7, the developed framework provides not only high decision accuracy but also high resilience to adversarial attacks throughout the transaction load process. With the help of adaptive stake-weighted voting combined with the use of behavioural reputation, the consensus algorithm can work reliably despite increasing aggression in the network environment.

In summary, the simulation findings indicate that the developed PoS-FPC model ensures bounded convergence and maintains decision accuracy when faced with increased levels of adversarial activity since the power of the validators is determined both through their reputation and economic commitment. As the number of transactions that need to be checked using the FPC process is relatively low, the system's response will be driven primarily by the faster weighted quorum approach. Moreover, the developed model incorporates two mechanisms aimed at addressing the challenges posed by adversaries, namely the weighted FPC for handling double-spending issues and the stake-weighted influence of validators for handling Sybil attacks.

7. Conclusion

The PoS-FPC scheme has been proposed to enhance consensus performance in IoT systems based on DAG architecture. The new PoS-FPC consensus algorithm combines an adaptive staking mechanism, behavioral reputation (mana), weighted quorums, and Fast Probabilistic Consensus, which helps to improve conflict resolution while ensuring the scalability and asynchronous nature of the DAG ledger. Additionally, adaptive traffic-aware weight, rewards and penalties, slashing, and Ed25519 authentication have been added to increase validator accountability and protect against attacks. Moreover, the PoS-FPC algorithm provides several formal guarantees related to safety, bounded convergence, and Sybil resilience, ensured by an economic security model. Evaluations were performed by conducting simulations for 20,000 transactions through 100 independent iterations. This yielded 5,128 TPS throughput, 99.6% decision accuracy, 97.81% adversary resistance, an average weighted quorum latency of 69.782 ms, an average FPC conflict latency of 690 ms, and an average convergence of 3.73 FPC rounds. In comparison with a mana-weighted DAG-FPC baseline, the proposed framework offered better decision accuracy, conflict resolution speed, convergence rate, and resistance to adversarial participation, while increasing the computational cost only marginally. It is evident from the above analysis that combining the stake-weighted voting approach with the probabilistic consensus mechanism is a reliable and efficient technique for securing and scaling IoT networks built on DAGs. Further work will be directed toward practical deployment of the suggested framework in IoT scenarios.

References

[1] Huan, Nathalie Tan Yhe, and Zuriati Ahmad Zukarnain. "A Survey on Addressing IoT Security Issues by Embedding Blockchain Technology Solutions: Review, attacks, current trends, and applications." IEEE Access 12 (2024): 69765-69782.

[2] Deepak, Preeti Gulia, Nasib Singh Gill, Mohammad Yahya, Punit Gupta, Prashant Kumar Shukla, and Piyush Kumar Shukla. "Exploring the Potential of Blockchain Technology in an IoT-Enabled Environment: A Review." IEEE Access 12 (2024): 31197–31227.

[3] Sedi Nzakuna, Pierre, Vincenzo Paciello, Aimé Lay-Ekuakille, Angelo Kuti Lusala, Salvatore Dello Iacono, and Antonio Pietrosanto. "From Iota Tangle 2.0 To Rebased: A Comparative Analysis of Decentralization, Scalability, And Suitability for IoT Applications." Sensors 25, no. 11 (2025): 3408.

[4] Deng, Weichu, Teng Huang, and Haiyang Wang. "A Review of the Key Technology in a Blockchain Building Decentralized Trust Platform." Mathematics 11, no. 1 (2022): 101.

[5] Guo, Fengyang, Artur Hecker, and Schahram Dustdar. "A Scalable and Secure Transaction Attachment Algorithm for DAG-Based Blockchain." IEEE Internet of Things Journal 12, no. 9 (2024): 12298-12309.

[6] Chen, Fu, Huizhu Li, Zhiyuan Sui, Kun Liu, and Wenying Tang. "Fast Probabilistic Consensus Protocol (FPC) Under Different Network Topologies." In 2022 International Conference on Service Science (ICSS), IEEE, 2022, 57-63.

[7] Qu, Xidi, Shengling Wang, Kun Li, Jianhui Huang, and Xiuzhen Cheng. "Tidyblock: A Novel Consensus Mechanism for Dag-Based Blockchain in IoT." IEEE Transactions on Mobile Computing 24, no. 2 (2024): 722-735.

[8] Müller, Sebastian, Andreas Penzkofer, Nikita Polyanskii, Jonas Theis, William Sanders, and Hans Moog. "Tangle 2.0 Leaderless Nakamoto Consensus on the Heaviest Dag." IEEE Access 10 (2022): 105807-105842.

[9] Rai, Hari Mohan, Kaustubh Kumar Shukla, Lilia Tightiz, and Sanjeevikumar Padmanaban. "Enhancing data Security and Privacy in Energy Applications: Integrating IoT and Blockchain Technologies." Heliyon 10, no. 19 (2024).

[10] Khan, Burhan Ul Islam, Khang Wen Goh, Mohammad Shuaib Mir, Nur Fatin Liyana Mohd Rosely, Aabid Ahmad Mir, and Mesith Chaimanee. "Blockchain-Enhanced Sensor-as-a-Service (Seaas) in IoT: Leveraging Blockchain for Efficient and Secure Sensing Data Transactions." Information 15, no. 4 (2024): 212.

[11] Wu, Fusheng, Xiuzhang Yang, Jinhui Liu, Yanbin Li, Mintao Ni, and Guangyan Jiang. "LCAG: A Lightweight Consensus Algorithm Based on Graph for the Internet of Things." IEEE Transactions on Dependable and Secure Computing (2026).

[12] Lin, Bing-Yang, Sebastian Müller, Luigi Vigneri, and Ching-Hua Lin. "Evaluating Blockchain Consensus Mechanisms: A Comprehensive Three-Tiered Approach with Application to Iota 2.0." In International Congress on Blockchain and Applications, Cham: Springer Nature Switzerland, 2024, 150-159.

[13] Shakir, Shukur H., and Alharith A. Abdullah. "Next-Gen Voting: A Blockchain-Based E-Voting System for Fast and Secure Digital Elections." In 2024 International Jordanian Cybersecurity Conference (IJCC), IEEE, 2024, 48-54.

[14] Kadhum, A. N., and A. M. Al-Salih. “Enhancing IoT Decentralization with IOTA 2.0: A Fast Probabilistic Consensus Framework.” Journal of Intelligent Informatics Networking and Cybersecurity 1, no. 1 (2025): 34–55.

[15] Lin, Z., K. Wang, Y. Wu, and D. Li. “Fast Payments Across Heterogeneous Blockchains for Internet of Things.” IEEE Access 12 (2024): 33923–33937.

[16] Müller, S., A. Penzkofer, B. Kuśmierz, D. Camargo, and W. J. Buchanan. “Fast Probabilistic Consensus with Weighted Votes.” In Advances in Intelligent Systems and Computing, vol. 1289, 360–378. Springer, 2021.

[17] Popov, S., and W. J. Buchanan. “FPC-BI: Fast Probabilistic Consensus within Byzantine Infrastructures.” Journal of Parallel and Distributed Computing 147 (2021): 77–86.

[18] Sealey, N., A. Aijaz, and B. Holden. “IOTA Tangle 2.0: Toward a Scalable, Decentralized, Smart, and Autonomous IoT Ecosystem.” In Proceedings of the 2022 International Conference on Smart Applications, Communications and Networking (SmartNets), 1–8. Palapye, Botswana: IEEE, 2022.

[19] Walumbe, D. W., G. N. Kamau, and J. W. Njuki. “A Lightweight Proof of Stake Voting Mechanism with Byzantine Agreement and Cryptographic Sortition for Telemedicine Systems.” International Journal of Computer Applications 187, no. 60 (2025): 31–39.

[20] Nguyen, Q., A. Cronje, M. Kong, A. Kampa, and G. Samman. “StakeDag: Stake-Based Consensus for Scalable Trustless Systems.” arXiv preprint arXiv:1907.03655 (2019).