PoMC: An Efficient Blockchain Consensus Mechanism for the Agricultural Internet of Things

Shuming Xiong, Zeyu Wang, Qiang Ni, *Senior Member, IEEE,* and Xue Han

***Abstract*—Blockchain-based agricultural IoT systems face key challenges such as high delay and low transaction throughput. Existing complicated consensus mechanisms can cause IoT devices work inefficiently due to the limited computing, storage and energy resources. Additionally, many message exchanges can lead to high latency in the consensus process, which hinders the real-time applications of the agricultural IoT. Therefore, we propose Proof-of-Multifactor-Capacity (PoMC), an efficient and secure consensus mechanism for the agricultural IoT. It uses the communication capacity and credibility of a node as the evidence for making consensus. Moreover, a senator node lottery algorithm based on a credit mechanism and a new distributed incentive mechanism are designed to enhance security and motivate nodes to actively maintain the system. This paper analyses the performance of PoMC theoretically, including security, latency and system throughput, and presents a comparison of its asymptotic complexity with some existing consensus mechanisms. The simulation results demonstate that the average transaction validation latency and average consensus latency of PoMC have decreased by 10% and 23%. In addition, PoMC outperforms SENATE, PoQF and PBFT by 56%, 60% and 64% in terms of the system throughput, respectively.**

***Index Terms*—Blockchain, consensus mechanism, distributed system, agricultural IoT, credit-based.**

I. Introduction

THE application of IoT technologies to agriculture can improve the efficiency and safety of agricultural production, as well as make big contributions to environmental protection [1]. However, there are still certain challenges in data management in the existing agricultural IoT. Agricultural IoT devices are vulnerable to failures and attacks, leading to the risk of data tampering in the outdoor application scenarios. In addition, as low-power WAN technology continues to advance, the number of agricultural IoT devices increases exponentially, resulting in high costs for centralized data management. Also, due to the heterogeneity of the IoT systems, the data flow between different departments and data centers is inefficient and poorly shared [2].

With the development of blockchain technology, some researchers have used it to solve the above-mentioned agricultural IoT problems. Blockchain is deemed as a decentralized distributed ledger, where all nodes in the network maintain one same blockchain and each node keeps a complete copy locally [3]. As an append-only supported ledger, blockchain is tamper-proof and traceable. Through blockchain technology, IoT data will be jointly maintained by all nodes in the P2P network, which can effectively achieve agricultural IoT data tracking and traceability, prevent data tampering and promote the flow of value in an untrusted environment [4], [5]. Miguel et al. [6] proposed a software architecture specifically designed for water management systems in untrusted environment, where resource-constrained IoT devices can process sensory data directly on a public blockchain, with the aim of stimulating a more sustainable approach to water management in agricultural production. On the other hand, Mohsin et al. [7] proposed a scalable distributed data sharing system based on smart contracts in an agricultural IoT system, enabling agricultural data to be shared among resource owners in a more secure manner.

Instead of relying on a central service provider, blockchain technology enables IoT to coordinate various nodes within a system to cooperate with each other through the consensus mechanism, in contrast to the traditional centralized structure. Consensus mechanism is one of the core technologies of blockchain, which enables all nodes within a decentralized peer-to-peer network to reach consensus on a transaction. However, some traditional blockchain consensus mechanisms generally need to consume a large quantity of computing resources or carry out complicated communication processes to achieve consensus. And worst of all, a large amount of nodes in IoT are resource-constrained and can hardly afford intensive computing or communication. Therefore, as average sensors, control devices, farm equipment, etc., their computing capability and energy provision are highly limited to run complex consensus mechanisms [8]. Bitcoin applying PoW has an average block generation interval of 10 minutes/block [9], i.e., a transaction requires about 10 minutes from being proposed to being included in a block and stored on the chain. Additionally, in some agricultural IoT scenarios such as greenhouse environment monitoring system and remote-control system for farming machinery, IoT devices are required to provide real-time feedback on the commands from the application-layer. Thus, the time-consuming consensus mechanisms are difficult to meet the low latency demands [10].

To solve the above problems, this paper proposes a blockchain consensus mechanism named Proof-of-Multifactor-Capacity (PoMC). PoMC exploits the credit capacity value and Signal to Interference plus Noise Ratio (SINR) capacity value as consensus evidence to reduce the time and computational

This work was supported in part by Special Science and Technology Innovation Program for Carbon Peak and Carbon Neutralization of Jiangsu Province (Grant No. BE2022610).

Shuming Xiong, Zeyu Wang, and Xue Han are with the School of Computer Science and Communication Engineering and Jiangsu Key Laboratory of Security Technology for Industrial Cyberspace Jiangsu University, Zhenjiang 212013, China (e-mail: xsm@ujs.edu.cn; 18852863599@163.com; hx3331048873@163.com).

Qiang Ni is with the School of Computing and Communications, InfoLab21, Lancaster University, Lancaster LA1 4WA, U.K. (e-mail: q.ni@lancaster.ac.uk).

load spent by nodes in the consensus procedure. The main contributions of this paper are as follows:

1) A new PoMC consensus mechanism is proposed for the resource-constrained agricultural IoT systems. In contrast to the PoW-like models where block generation is driven by computional power, PoMC enables senators to allocate the accounting rights based on the multifactor capacity value of the mining nodes. The proposed mechanism tends to select nodes with good communication conditions and high credibility as accounting nodes, which enhances the security and robustness of the system while reducing the resource consumption and transaction validation delay.
2) A senator node lottery algorithm is designed to select partial nodes as senators in the network based on the credibility. In order to participate in the competition for the senator nodes, a node needs to solve a hash puzzle related to the credibility, which allows PoMC to resist Sybil attacks. Senators jointly manage the credibility mechanism in the system and elect the accounting nodes based on the multifactor capacity, decreasing the network overhead for consensus.
3) A distributed incentive mechanism is proposed to reward honest nodes for their positive contributions to the system while limiting the attacks of malicious nodes.
4) We theoretically analyze the latency and transaction throughput of the system, as well as the security in the face of different attacks, and conduct simulation experiments using OMNeT++. Compared with some existing consensus schemes, the results demonstrate that PoMC has advantages in latency, system throughput and consensus success rate.

II. RELATED WORK

As one of the core technologies of blockchain, the consensus mechanism ensures that all nodes reach consensus on a transaction within a decentralized distributed system. The traditional consensus mechanism generally consumes significant computing and communication resources to achieve agreement among nodes, which is not fully applicable to the IoT devices with limited resources. So, the transaction processing speed and scalability need to be further improved as the number of IoT devices increases. According to working principle, consensus mechanisms can be divided into three main categories, including PBFT type, proof-based type, and credit-based type.

The PBFT-type consensus mechanism is derived from the Practical Byzantine Fault Tolerance (PBFT) algorithm, which is a form of state machine replication that models services as a state machine replicated between different nodes in distributed system [11]–[15]. Laphou et al. [11] proposed a consensus protocol named G-PBFT. In G-PBFT, the system selects nodes that are location-fixed, honest and resourceful as accounting nodes, thus reducing the overhead of validating or processing transaction. Jiang et al. [12] presented a consensus protocol named SENATE, in which the system draws lots to select some nodes as candidates for senator nodes based on the ALOHA communication protocol. After completing selection, some appropriate senator nodes are elected based on the ranging of the nodes, and finally consensus is reached among these senator nodes based on BFT. In the above-mentioned solutions, although the computation load and transaction latency of the nodes are alleviated, selecting some nodes to participate in consensus in a large-scale network leads to the risk resulted from system centralization.

The most typical consensus mechanism for the proof-based type is Proof-of-Work(PoW) [16]–[18]. A common form of the PoW algorithm is like $H(param||nonce) < target$, where param denotes the data associated with the block, nonce is a random value, and target denotes the target value. Mining nodes continuously compute to find the nonce that satisfies the condition to obtain the accounting right. Yazdinejad et al. [16] proposed the SLPoW algorithm to enhance the security of IoT system while reducing the computation and energy load of the IoT devices. To alleviate the computation overhead in PoW, Proof of Stake(PoS) introduces the concept of coinage, which is the product of the number of tokens held by nodes and the time for holding them. The larger the coinage, the higher the probability of winning the competition for the accounting rights [19]. Wei et al. [20] improved DPoS by designing a data detection algorithm to identify anomalous IoT data from malicious attacks and selfish behaviors of nodes, which is used to resist attacks in DPoS. For PoS, the larger the coinage held by a node, the higher the probability of obtaining accounting rights, which leads to the stake accumulation problem. It is noted that some nodes hold high quantities of tokens, which results in a disadvantage for new nodes joining the network in the competition for the accounting rights.

Credit-based-type consensus mechanisms exploit the credibility of nodes as the support for agreements. Huang et al. [21] proposed a credit-based Proof-of-Work mechanism, in which the mining difficulty of each node is related to its own credibility, and the higher the credibility, the lower the mining difficulty of the node and the higher the possibility of generating blocks. Zhang et al. [22] presented a consensus protocol based on credibility and established a credibility review mechanism. Nodes maintain a list containing the credibility of all their neighbor nodes, and consensus is accomplished through credibility comparison. Liu et al. [23] designed a green consensus mechanism named collaborative multiple proofs, which relies on a collaborative index framework. Each node's collaborative index is influenced by the node's behaviors, and the nodes use their own collaborative index as evidence to participate in consensus. Credit-based-consensus mechanism is characterized by low computation overhead and low latency, but the disadvantage is that the credibility of nodes is difficult to manage in large-scale IoT application scenarios.

The PBFT-type consensus mechanisms rely on multiple data exchanges among nodes to reach consensus. However, in some application scenarios of agricultural IoT, the communication between nodes is conducted through wireless transmission, and unstable networks may lead to message delays and packet loss. Proof-based consensus mechanisms, which take computing power as the core of block generation, are difficult to meet the requirements of low resource consumption and

Table I
COMPARISON OF MAINSTREAM CONSENSUS MECHANISMS

Consensus mechanism	PoW [9]	PoS [29]	PoA [30]	PBFT [14]	RAFT [31]
Decentralization	Complete	Complete	Complete	Partial	Partial
Computation consumption	High	Middle	Middle	Low	Low
TPS(tx/sec)	7-56	70-8000	33-57	1000	1000
Consistency	High	High	High	Middle	Middle
Access authorization	Unnecessary	Unnecessary	Necessary	Necessary	Necessary

low transaction validation latency of agricultural IoT devices. Although PoET [24] and PoR [25] have abandoned the idea of using computational power as the driving force, their evidence for consensus is respectively time and storage space, which are also unsuitable for agricultural IoT. Although credit-based consensus mechanisms have the advantages of low cost and low latency, relying solely on credibility as the evidence for consensus may lead to centralization risks. The theoretical comparison of mainstream consensus mechanisms is presented in Table I, illustrating the limitations of existing consensus mechanisms when integrating with Internet of Things technologies.

Currently, there is ongoing research exploring the integration of blockchain and IoT facilitated by technologies such as Deep Learning(DL) and edge computing. Prabhat Kumar et al. [26] established a one-to-one mapping relationship between digital twin edge nodes and physical entities in the Industrial Internet of Things (IIoT). They ensured data security and consistency based on Proof of Authority (PoA). However, in comparison to PoW and PoS, PoA exhibits a limited degree of decentralization. Literature [27] introduces an enhanced PoW algorithm, ePoW, which employs smart contracts to verify the validity of data in the IIoT, preventing data tampering or forgery. Furthermore, the integration of DL techniques is applied for privacy protection and threat detection, thereby improving the security of the network. Literature [28] similarly adopts the ePoW consensus mechanism and combines it with Principal Component Analysis (PCA) technology. This integration avoids the inference of sensitive information from IoT data by DL-based systems, simultaneously reducing the storage load and energy consumption of blockchain nodes. While ePoW improves the performance of PoW by dynamically adjusting difficulty and reward coefficients, balancing security and efficiency remains a challenge.

The consensus mechanism proposed in this paper does not require nodes to perform complex computations, but instead uses multifactor capacity values as the core evidence to drive block generation. Multifactor capacity values can be used as a comprehensive indicator to measure the communication ability and credibility of nodes, which encourages the consensus mechanism to select nodes with better network conditions and higher contributions to the system as accounting nodes. Additionally, this paper designs a lottety algorithm for selecting some nodes as senators to represent other nodes in the allocation of accounting rights, reducing network overhead in the consensus process.

III. SYSTEM MODEL

We design an architecture of the blockchain-based secure data collection and transmission system for agricultural IoT. Fig. 1 illustrates the three-layer architecture of the system, including sensing layer, blockchain layer, and application layer. The devices in the sensing layer, which do not engage in the processes of blockchain due to the constrained resources, include sensors, RFID tags, agricultural equipments, etc. They mainly collect the environment data and transmit them to the blockchain layer. Nodes in the blockchain layer are devices with relatively abundant computational and storage resources, such as IoT gateways and edge computing nodes. These nodes are peers that directly participate in the blockchain consensus mechanism, organizing IoT data into transactions and storing them on the blockchain. Each node in the blockchain layer keeps a copy of the complete blockchain and interacts with the sensing layer and the application layer through APIs. The application layer contains IoT cloud platforms, data centers, and management platforms, through which users can get the required data and develop corresponding functions by accessing the blockchain. All instructions and operations from the application layer will be encapsulated as transactions and stored on the blockchain by nodes in the blockchain layer to achieve the traceability and tamper-proof features. This system can be deployed in application scenarios that require real-time monitoring of environmental data, such as greenhouses, forests, and farms. In the event of an emergency, the system can promptly notify users to take appropriate actions.

To implement the consensus function, we set two types of nodes with special identities in the system, namely, the senator node and the accounting node. Senators are responsible for managing the credibility of all nodes in the system and selecting the accounting node based on the multifactor capacity value. The system model has the following assumptions:

1) There is a possibility of abnormal nodes existing in the system, including faulty and malicious nodes.
2) The purpose of the malicious nodes' attacks is to manipulate the consensus process so that the result is beneficial to themselves.
3) The behaviors of the malicious nodes are restricted to themselves, i.e., the malicious nodes will comply with the communication protocol rather than disrupting it and will not perform eclipse attacks [32] or DDoS attacks [33], etc.

When any node joins the network, it is required to broadcast a message containing information about its neighbors, its own location, timestamp, etc. The message is included in the block by the accounting node and broadcasted in the network to

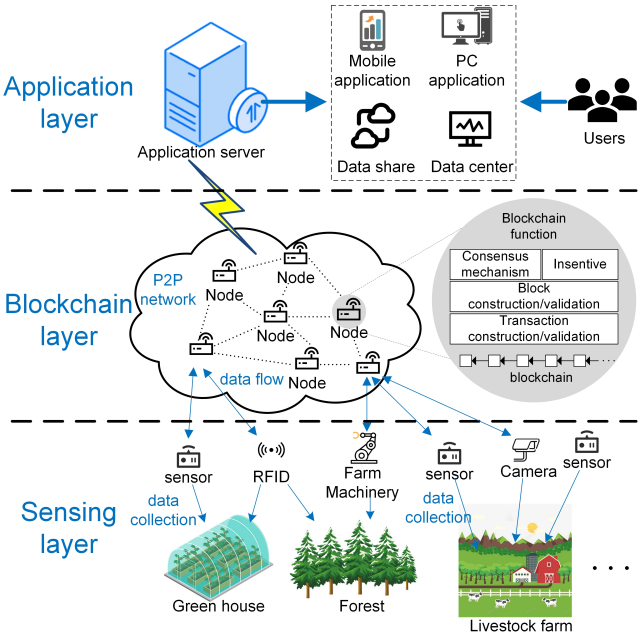

Figure 1. Architecture of the blockchain-based secure data collection and transmission system for agricultural IoT.

complete the registration. Each node keeps a view of the overall nodes in the network, and the view keeps dynamically updated.

IV. DESIGN OF PoMC CONSENSUS MECHANISM

This section gives the detailed design of PoMC consensus mechanism. The key notations and corresponding explanations are presented in TABLE II.

A. Senator Nodes Lottery

In a distributed system, the credibility reflects the degree of contribution of a node to the system. As an ordinary node in Fig.2 it shows the process that the senators derive the credibility of the node i. The rating depends on the node history behaviors, and we can calculate the credibility C_i by combining the ratings from all senators. The node history behaviors include constructive behaviors that are positive to the system and destructive behaviors negative to the system, which are both called Credibility Influence Behaviors (CIBs). CIBs are marked differently in the system and the senators send them to the accounting nodes in transactions which are in the form of $Trans\{addr, R, timestamp, Rating_i\}$, where $addr$ is the public key of the senator node, R is the type of the CIB, and $Rating_i$ is the credit rating given from this senator to i. When a node conducts a Credibility Influence Behavior, the senators re-evaluate the credibility of it to dynamically update its credibility value.

We design a lottery algorithm based on the credibility mechanism to select some nodes as senators in the network. The senator has a term of office, and one term of the senator node is regarded as one round. In round r, node i participates in the lottery according to the following steps:

Table II
KEY SYMBOLS

Symbol	Description
d_{ij}	Distance between node i and node j
β_1	$SINR$ threshold
β_2	Credibility threshold
α	Path loss exponent
P_{noise}	Noise Power
n_{itf}	Number of Interference Nodes
N	Total number of nodes
N_0	Number of nodes in interference area D_0
γ	Distribution density of nodes
N_S	Number of senator nodes
N_C	Number of candidate accounting nodes
N_m	Number of mining nodes
T_{delay}	Transaction validation latency
L_B, L_T	Standard block, transaction size
W	System throughput

1) The system publishes a difficulty value $target$ for the rth round.
2) Node i performs the following operation:

$$\mathrm{H}(param||nonce) < C_i \cdot target \tag{1}$$

where $param$ is an attribute parameter of node i and $nonce$ is a random number. The first N_S nodes that find the $nonce$ satisfying the conditions win the senator node lottery and become the senators in the rth round. The higher the C_i, the higher the probability that node i will win the senator node lottery.

The individuals that joined the network earlier will accumulate more and more credibility as the system runs continuously, putting the newly joined nodes at a disadvantage in the senator node lottery. Therefore, we set every passing P rounds as one credibility cycle, and the credibility of the whole network will be cleared at the end of the credibility cycle.

B. PoMC Consensus Mechanism

Since there could still be malicious nodes among the senators, further consensus within the system is needed to prevent malicious nodes from obtaining the accounting rights. This subsection proposes a consensus mechanism based on Proof-of-Multifactor-Capacity (PoMC).

1) Consensus processes based on Proof of Multifactor Capacity: In PoMC, the nodes that compete for the accounting rights are mining nodes. For a mining node i, its multifactor capacities value consists of two parts: 1) Signal to Interference plus Noise Ratio capacity value, i.e., the probability that node i can successfully communicate with the nearest senator, denoted as $\Pr(SINR_i \geq \beta_1)$; 2) the credit capacity value, i.e., the probability that the credibility C_i satisfies the system credibility threshold β_2, denoted as $\Pr(C_i \geq \beta_2)$. Hence the multifactor capacities value MC_i is the product of the above two value, i.e., $MC_i = \Pr(SINR_i \geq \beta_1) \cdot \Pr(C_i \geq \beta_2)$. The Signal to Interference plus Noise Ratio capacity value reflects the communication capability of the node, and the credit capacity value shows the degree of contribution of node i to the system. The steps for consensus based on the multifactor capacities value MC_i are as follows.

Step 1: A mining node i sends a message containing $SINR_i$ and C_i to the nearest senator. After receiving the message

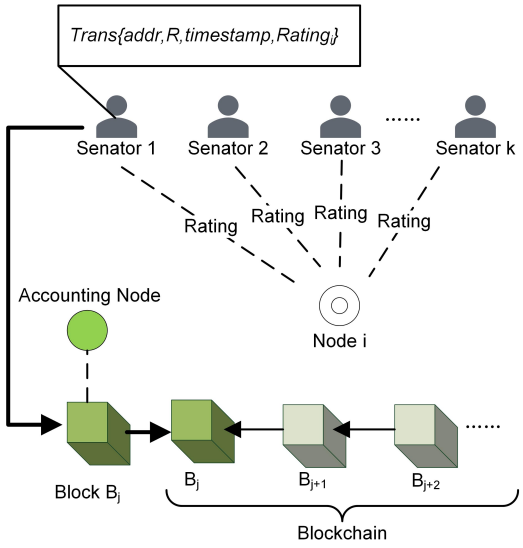

Figure 2. Senator nodes evaluate the credibility of node i.

from node i, the senator calculates MC_i. Since each mining node communicates with the senator nearest to it, the number of consensus messages received by each senator is different. The senator takes the node with the largest MC value in the consensus messages received within the limited time as a candidate accounting node and broadcasts its information among senators. Finally, N_C candidate accounting nodes will be selected out. There is a possibility that some senators do not receive consensus messages, thus $N_C \leq N_S$.

Step 2: For the purposes of preventing malicious candidates from gaining accounting rights and restraining evil behaviors of adversary senators, each senator verifies all candidates and votes for them according to the verification results. Taking a candidate n_1 for instance, the set of its votes is $V_{n_1} = \{S_1Y, S_2N, \cdots, S_{N_S}Y\}$, where S_1Y indicates that senator S_1 casts positive vote for it, but S_2N means that senator S_2 casts negative vote for it, and so on. Node n_1 will win the accounting rights and become the accounting node under the following conditions: 1) the number of its positive votes is greater than $\frac{2}{3}N_S$; 2) it has the highest MC among all candidates. The accounting node also has a term of office and can generate multiple blocks during its term.

If a mining node wins the accounting rights competition successively or in a short time slot, its energy consumption will be fast, and the possibility of it acting evilly will increase. Therefore, we set the random waiting time τ_i which satisfies the uniform distribution $\tau_i \sim U(a, b)$. The accounting node enters a random waiting time at the end of its term and is unable to compete for the accounting rights until the end of τ_i. In addition, if an accounting node is found to perform malicious behaviors during its term, the senators will promptly issue a message to terminate its term and start selecting the next accounting node.

2) Calculation of MC_i: According to the above, the multifactor capacities value MC_i of a node i contains two components, i.e., $\Pr(SINR_i \geq \beta_1)$ and $\Pr(C_i \geq \beta_2)$. $SINR_i$ in the case of i communicating with the nearest senator node

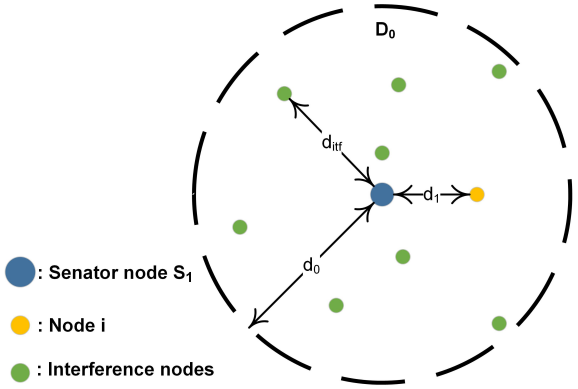

Figure 3. Interference area for senator node S_1.

(taking S_1 for example) can be expressed as

$$SINR_i = \frac{P_{tr} d_1^{-\alpha}}{P_{noise} + P_{tr} \sum_{k=1,k\neq i}^{n_{itf}} {d_{kS_1}}^{-\alpha}} \tag{2}$$

where α is the path-loss exponent, P_{noise} denotes the noise power, and P_{tr} is the node transmission power [34]. As shown in Fig.3, nodes can cause interference to S_1 in the area D_0, and n_{itf} denotes the number of interference nodes. Number d_1 denotes the distance value between nodes i and S_1. So, $\Pr(SINR_i \geq \beta_1)$ can be expressed as

$$\begin{aligned} &\Pr(SINR_i \geq \beta_1) \\ &= \Pr\left(d_1 \leq {P_{tr}}^{\frac{1}{\alpha}} [\beta_1 (P_{noise} + \sum_{k=1,k\neq i}^{n_{itf}} {d_{kS_1}}^{-\alpha})]^{-\frac{1}{\alpha}} \right) \end{aligned} \tag{3}$$

According to [35], the geographical distribution of nodes in spatial domain can be modeled as Poisson point process (PPP), and the probability density function of d_1 can be expressed as

$$f_{d_1}(x) = \frac{2\pi N_S x}{N} \mathrm{e}^{-\frac{\pi N_S}{N} x^2} \tag{4}$$

where N denotes the number of nodes in the system. Therefore, $\Pr(SINR_i \geq \beta_1)$ can be further expressed as

$$\begin{aligned} \Pr(d_1 \leq \eta) &= \int_0^{\eta} f_{d_1}(x)\, \mathrm{d}x \\ &= 1 - \mathrm{e}^{-\frac{\pi N_S \eta^2}{N}} \end{aligned} \tag{5}$$

and

$$\eta = {P_{tr}}^{\frac{1}{\alpha}} [\beta_1 (P_{noise} + \sum_{k=1,k\neq i}^{n_{itf}} {d_{kS_1}}^{-\alpha})]^{-\frac{1}{\alpha}} \tag{6}$$

where d_{kS_1} denotes the distance of the kth interference node from the central senator node S_1. Considering the agricultural IoT application scenarios for an example, the IoT devices are not always active and will enter a sleep mode to reduce power consumption when no messages arrive or no transactions need to be handled. Therefore, the nodes in the sleep mode will not cause interference. According to [35], the probability that a node is active follows a Poisson distribution with intensity λ_i,

i.e., $\Pr(active) = \mathrm{e}^{-\lambda}$. Therefore, the number of nodes that can make interference is

$$n_{itf} = (N_0 - 1)\Pr(active) \tag{7}$$

where N_0 denotes the number of nodes in region D_0.

We model the nodes' credibility C_i as a normal distribution, i.e., $C_i \sim N(\mu, \sigma^2)$, where μ and σ^2 denote the mean and variance of C_i in the past n_P credibility cycles. Then

$$\Pr(C_i \geq \beta_2) = \frac{1}{2} - \frac{1}{2}\mathrm{erf}\left(\frac{\beta_2 - \mu}{\sqrt{2}\sigma^2}\right) \tag{8}$$

3) Senators Voting Rules: Senators need to verify the multifactor capacities value of the candidate accounting nodes and vote for them according to the verification. Assuming that each node can obtain ranging estimation with another node by the pilot signals, which can be based on the Received Signal Strength (RSS), the Time of Arrival (ToA), or other approaches which have been studied extensively [36]. The ranging estimation is modeled as

$$\hat{d}_{ij} = \omega_{ij} d_{ij} + \upsilon_{ij} \tag{9}$$

where d_{ij} denotes the geographical distance between node i and node j. Node i can derive d_{ij} by calculating the position coordinates registered by node j and its own position coordinates. The distance estimation between i and j is denoted as $\hat{d}_{ij}$. The estimation error is introduced by multiplicative and additive random coefficients ω_{ij} and υ_{ij}, respectively, which vary in different ranging approaches. Senator S first checks the $SINR_i$ of candidate node i. If $SINR_i \geq \beta_1$, S continues with the following check:

$$|d_{Si} - \hat{d}_{Si}| \leq \varepsilon_1 \tag{10}$$

where ε_1 is a constant value related to ω_{ij} and υ_{ij} [12]. Combining Eq.3 and Eq.5, if Eq.10 holds, the $SINR_i$ capacity value for node i is verified to pass.

To verify the credit capacity value, the senator S searches for the CIBs of node i on the blockchain and calculates the credibility estimation C_i, the credibility mean estimation $\hat{\mu}_i$, and the credibility variance estimation ${\hat{\sigma}_i}^2$. If $C_i > \beta_2$, S will perform the following probability check:

$$|\Pr(C_i \geq \beta_2) - \Pr(\hat{C}_i \geq \beta_2)| \leq \varepsilon_2 \tag{11}$$

where ε_2 is a constant value which indicates the probability error. If Eq.11 holds, the credit capacity value for node i is verified to pass.

Finally, a positive vote is cast for a candidate accounting node only if the senator passes both the verification of the SINR capacity value and the credit capacity value. Otherwise, a negative vote from senator is provided.

C. Distributed Incentive Mechanism

When a node maintains or destroys the system, it can only gain or lose credibility before implementing the distributed incentive mechanism. However, clearing the credibility at the end of each credibility cycle is unfair to the nodes that keep maintaining the system. In addition, clearing the credibility reduces the cost of malicious behaviors indirectly. Therefore, we design a distributed incentive mechanism to give token rewards for honest behaviors to motivate the nodes to actively maintain the system and to deduct the tokens of nodes that perform malicious behaviors.

There are two ways for a node to obtain tokens: 1) to vote for candidate accounting nodes honestly as a senator node, and 2) to actively publish blocks as an accounting node. The total incentive assigned by the system to the rth round is I_r. And I_r is divided into two parts IM_r and IC_r, which indicate the tokens assigned to the senators and the accounting node respectively.

An expression $V_S = 0$ denotes that senator S votes positively for a candidate accounting node, but $V_S = 1$ means that the senator S votes negatively. For a single candidate accounting node, the incentive $U_S(r)$ that can be obtained by S is

$$U_S(r) = \begin{cases} \frac{IM_r}{N_C n_h}, & V_S = 0 \ and \ B_c = 0 \\ \frac{IM_r}{N_C n_h}, & V_S = 1 \ and \ B_c = 1 \\ \frac{IM_r}{N_C n_h} - q, & V_S = 1 \ and \ B_c = 0 \\ q, & V_S = 0 \ and \ B_c = 1 \end{cases} \tag{12}$$

where $B_c = 0$ states that the target node is honest, $B_c = 1$ implies that the target node is malicious, n_h is the number of honest nodes in the current round of voting, and q is a positive value that serves to penalize the node in the case of dishonest performance or misjudgment.

On the other hand, the incentive $UM_i(r)$ that can be obtained with node i as an accounting node is

$$UM_i(r) = IC_r \frac{BC_i(r)}{BC(r)} \tag{13}$$

where $BC(r)$ is the total number of blocks generated in the rth round, and specifically $BC_i(r)$ is the number of blocks produced by node i during its tenure.

V. THEORETICAL PERFORMANCE ANALYSIS

This section conducts the theoretical analysis of consensus mechanisms in terms of security, working performance and asymptotic complexity.

A. Security

1) Sybil Attack: A sybil attack refers to a malicious node disguising multiple identities in the system to manipulate the consensus results [37]. It mainly works on the blockchain systems adopting vote-based consensus mechanisms [38]. Assuming that all nodes have the same hash power (the number of hash computations that can be performed per unit of time), according to Eq.1, for a node i, its probability of winning the senator node lottery can be expressed as

$$\Pr(S_{suc}) = \frac{C_i}{\sum_{k=1}^{\mathrm{E}(N_0)} C_k} \tag{14}$$

where S_{suc} denotes the successful winning event in the senator lottery. $\mathrm{E}(N_0)$ is the expected value of the number of nodes

Table III
PAYOFF MATRIX OF SENATOR NODES

		Any other senators	
		H	M
Senator S	H	$(\frac{IM_r}{n_h N_C}, \frac{IM_r}{n_h N_C})$	$(\frac{IM_r + n_f q}{n_h N_C}, -q)$
	M	$(-q, \frac{IM_r + n_f q}{n_h N_C})$	$(-q, -q)$

in the area D_0 centered on the senator node nearest to node i, which can be expressed as

$$\mathrm{E}(N_0) = \sum_{k=1}^{\pi {d_0}^2 \gamma} \frac{(\pi {d_0}^2 \gamma)^k}{k!} \mathrm{e}^{-\pi {d_0}^2 \gamma} \tag{15}$$

where γ is the distribution density of the nodes. According to Eq.14, the probability of a node winning the lottery is the ratio of the node's credibility to the sum of the credibility of all nodes in D_0. The higher the credibility of a node, the higher its probability of winning the lottery. In this scheme, the node's credibility value is closely related to the CIB, which means the malicious nodes can fake numerous node identifications to participate in the senator lottery but cannot accumulate credibility for multiple fake nodes simultaneously. Hence it is difficult for the malicious nodes to successfully implement the sybil attack in the senator node lottery stage.

In the accounting rights competition phase, the senators can obtain the ranging estimations with other nodes, so sybil nodes can be easily detected when they disguise themselves as other nodes in this phase. Additionally, Elliptic Curve-based Digital Signature techniques can be deployed to prevent nodes from publishing fake messages or tampering with the content of messages [39].

2) Collusion Attack: If the malicious nodes behave honestly before becoming senators, they will not be completely eliminated in the senator node lottery. The malicious nodes aim to make the consensus results favorable to themselves. Accordingly, there are two types of possible collusion attack: a) malicious senators vote negatively for honest candidates so that honest nodes cannot become accounting nodes; b) malicious senators vote positively for malicious candidates to make them win the accounting rights competition. If a malicious node wins the competition for accounting rights, it can implement attacks such as forging transactions and tampering with the transaction contents during its tenure. In the scenario mentioned above, regardless of the faulty nodes, the number of malicious nodes N_f needs to satisfy the two cases shown in Eq. (16) in the face of those two collusion attacks. At least $\frac{1}{3}N + 1$ malicious nodes are needed for the attack to succeed.

$$\begin{cases} N_f > \frac{2}{3}N, case\ 1 \\ N_f > \frac{1}{3}N, case\ 2 \end{cases} \tag{16}$$

3) Game Theory Analysis of Distributed Incentive Mechanism: We analyze the distributed incentive mechanism of the scheme based on the game theory, focusing on the benefits that senators can obtain. Table III presents a payoff matrix for the game.

1) Players: There are N_S (the number of senators) players in the game, including n_h honest senators and n_f malicious senators.
2) Actions: A symbol H indicates that the senator behaves honestly in the voting, i.e., the senator votes positively for honest nodes, and vice versa. And, a symbol M represents that the senator node behaves maliciously in the voting.
3) Utilities: The element a in tuple (a, b) denotes the benefits available to a senator and the b means the benefits available to other senator nodes.

According to the payoff matrix shown in Table III, for senator node S, its payoff maximum is $(IM_r + q)/(n_h N_c)$, and for other senator nodes, their payoff maximum is $(IM_r + n_f q)/(N_c)$. When S chooses either strategy H or M, other senators will choose H to maximize the payoff. Thus, the strategy (H, H) achieves both a Pareto optimal solution and a pure strategy Nash equilibrium solution.

If a malicious senator votes positively for the all candidates, the incentive obtained by the senator is $(n_{hc} IM_r)/(n_h N_C) - n_{ac} q$, where n_{hc} and n_{ac} are the numbers of honest nodes and abnormal nodes among the candidates respectively. Therefore, to make being honest as the best response action of the senators, it is required that $q > [(n_{hc} IM_r)/(n_{ac} n_h N_C)]$. Thus, the distributed incentive mechanism proposed in this paper can adjust the values of q and IM_r to inhibit malicious senators from conducting attacks during voting.

B. Latency and Throughput

The transaction validation latency T_{delay} represents the time it takes for a node to generate a transaction until the transaction is validated on the chain. Within a blockchain system based on the proof-based consensus mechanisms, consensus is performed once for each generated block, resulting in significant latency. In this scheme, an accounting node can generate multiple blocks during its tenure. Before the end of its term, the next accounting node selection has already started, thus a new accounting node will continue to generate blocks when the term of the last accounting node ends.

Suppose the number of transactions generated by a node per unit time is n_t, then the number of transactions generated by all nodes per unit time is $n_t N$. And, T_{delay} can be expressed as

$$T_{delay} = T_0 + T_1 + T_m \tag{17}$$

where $T_m \in \left(0, \frac{L_B}{n_t N L_t}\right]$ and T_m means the time that the transaction is waiting to be packed in the transaction pool. L_B and L_t denote the size of a block and a transaction, respectively. T_0 indicates the time required for a transaction to be generated and received by the accounting node, and T_1 shows the time required for the accounting node to pack the block and finish broadcasting.

Transaction throughput depicts the number of transactions that a blockchain system can process per unit of time. The maximum data throughput of the accounting node can be

Table IV
COMPARISON OF ASYMPTOTIC COMPLEXITY FOR SEVERAL SCHEMES

Consensus	Latency	Security	Communication
PoW [40]	$\Theta(\kappa)$	$\Omega\left(\frac{N_m}{2}\right)$	$\Theta(1)$
PBFT [40]	$N_m O(1)$	$\Omega\left(\frac{N_m-1}{2}\right)$	$O(N_m^2)$
SENATE [12]	$O(N_m)+O(\kappa^2)$	$\Omega\left(\frac{N_m-1}{3}\right)$	$\Omega(N_m)$
PoQF [41]	$\kappa O(1)$	$\Omega\left(\frac{N_m}{2}\right)$	$O(N_m)$
PoMC	$O(N_m)+O(\kappa^2)$	$\Omega\left(\frac{N_m-1}{2}\right)$	$\Theta(1)$

expressed as λ_{max}(bit/s), and the transaction throughput W of the system can be expressed as

$$W = \min\left\{\frac{n_t N L_t}{L_B}, \frac{8\lambda_{max}}{L_B}\right\} \tag{18}$$

C. Asymptotic Complexity

This subsection evaluates the scalability of the proposed scheme by analyzing latency complexity, security complexity, and communication complexity of the consensus protocol and compares it with several other schemes. Latency complexity means the time consumed by a consensus mechanism, and security complexity indicates the maximum number of abnormal nodes that a consensus algorithm can tolerate. Communication complexity denotes the number of messages required for a transaction to be validated.

Supposing the number of mining nodes as N_m and the consensus core parameter as κ, we derive the complexity results shown in Table IV. The symbols $\Omega()$, $O()$ and $\Theta()$ denote at least, at most, and at exactly, respectively [40]. The parameter κ has different meanings for different consensus mechanisms. For example, κ in PoW denotes the difficulty of the hash puzzle, while in PBFT and PoQF it denotes the threshold of vote, and in SENATE and PoMC it represents the number of senator nodes. Therefore, the complexity affected by κ cannot be directly compared. The first node to find the answer of the hash puzzle in PoW wins the consensus competition. PoQF needs to wait for the number of votes to satisfy the threshold κ before the next relay node can be selected. And the latency complexity of SENATE and PoMC is affected by both N_m and κ. As shown in Table IV, the security of PoMC is better than SENATE, but slightly inferior to PoW and PoQF. For communication complexity, PoW and PoMC perform better than PBFT, PoQF and SENATE, because, for a single node, PoW and PoMC do not require a large number of messages to be exchanged during transaction validation. But for total system overhead, PoMC has a lower communication complexity than PoW.

VI. EXPERIMENTAL RESULTS AND EVALUATIONS

In this section, the performance of the proposed PoMC blockchain consensus scheme is analyzed based on the OMNeT++, and the source code of the simulation project is released[1]. The parameters of the experiments are shown in Table V. The nodes are randomly distributed in a fixed size experimental area and generate transactions at 10∼15s intervals to simulate the process of randomly produced transactions by IoT devices.

Table V
SIMULATION PARAMETERS

Parameter	Value	Description
T_{sim}	1000 s	Simulation runtime
A_N	1 km×1 km	Size of nodes distribution area
$R_{channel}$	7200 kbps	Node-to-node channel transmission rate
L_{pac}	1500 bytes	Packet size
C_{Block}	50	Number of transactions that can be contained in a single block
L_T	1000 bytes	Size of single transaction
P_{noise}	-10 dBm	Noise power
P_{tr}	1 dBm	Transmitting power
R_{Com}	200 m	Communication range of nodes
α	1	path-loss exponent
I_T	10 15 s	Time interval for nodes to generate transactions
T_S	100 s	Senator Node Term
T_A	100 s	Accounting Node Term

A. Simulation Results of the Proposed Consensus Mechanism

Fig.4(a) shows the variation of consensus latency of PoMC with increasing number of senator nodes. When the number of senator nodes increases, the consensus latency increases accordingly. Because there are more senator nodes proposing the candidate mining nodes, more messages need to be exchanged to achieve consensus. The size of the candidate node list is only related to the number of senator nodes, so the number of mining nodes has less impact on the consensus latency. Fig.4(b) depicts the effect of the change in the number of senator nodes in PoMC on the system transaction throughput and total network overhead. As the number of senator nodes increases, the system transaction throughput does not change significantly. When the number of senator nodes increases to 35, the system transaction throughput has a decreasing trend instead, because excessive senators lead to a decrease in the efficiency in selecting the accounting node. Meanwhile, the number of messages to be dealt with by the senators becomes larger, so the processing time will be longer, and the system network overhead will rise. Fig.4(c) describes the changes of network-wide credibility and individual node credibility in one credibility cycle (the length of the credibility cycle is set as 1000s), respectively. The accumulation rate of single node credibility is much smaller than network-wide credibility, which indicates that the probability of a malicious node successfully conducting a sybil attack decreases over time within a credibility cycle.

B. Comparative Analysis

The transaction validation latency represents the time taken for a transaction to be proposed and achieve consensus on the blockchain, and the consensus latency indicates the time required for miners to achieve consensus on a transaction (or a block). Fig.5(a) gives the transaction validation latency

[1]Source code is available at https://github.com/crocodileWang/PoMC.

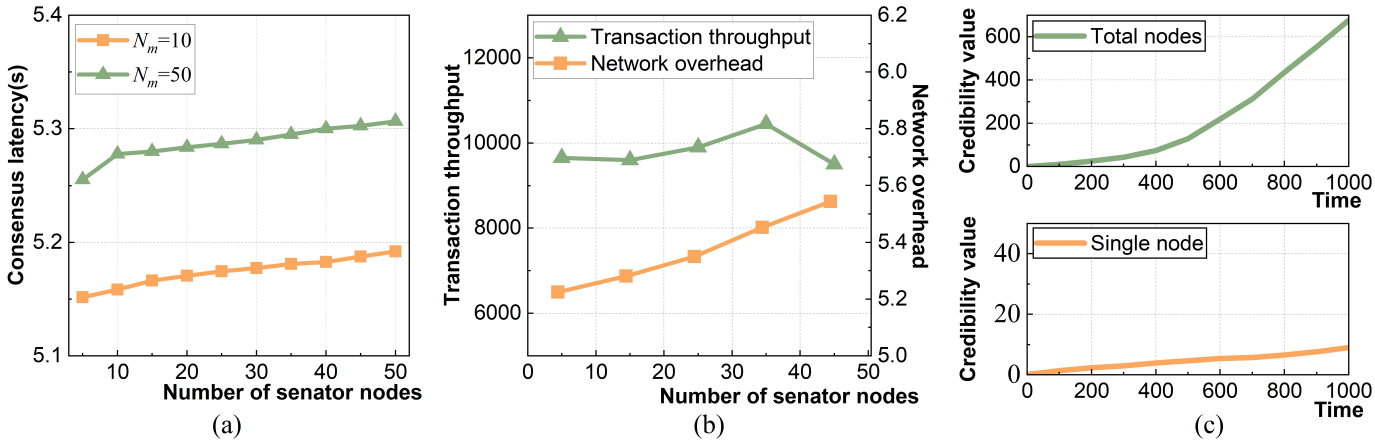

Figure 4. The performance of PoMC. (a) The latency variation of the consensus process with different number N_m of mining nodes. (b) Transaction throughput and network overhead. (c) Credibility value changes with running time.

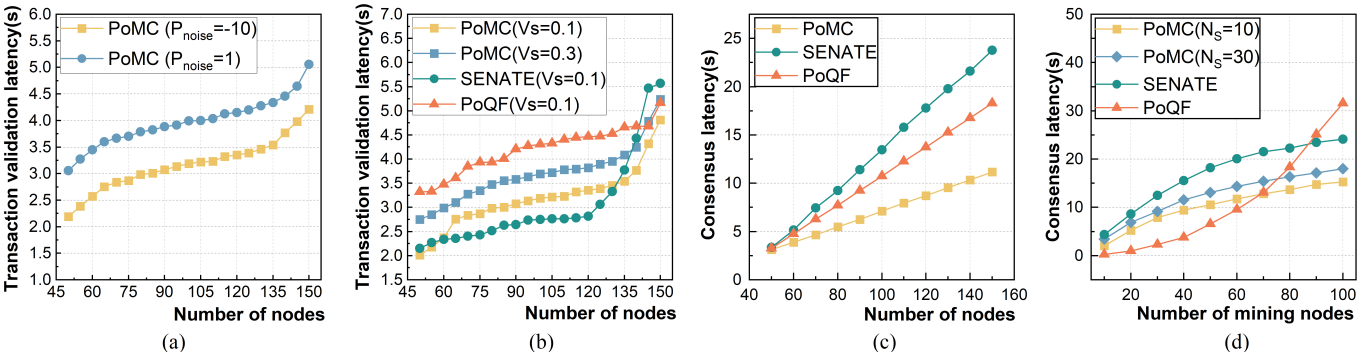

Figure 5. The latency experimental results. (a) The transaction validation latency of PoMC under different noise conditions. (b) The transaction validation latency comparison with SENATE and PoQF with $N_S = 10$ and $N_M = 10$. (c) The consensus latency with different number of nodes. (d)The consensus latency with different number of mining nodes.

of PoMC under different noise conditions. Senators tend to select nodes with good communication conditions as accounting nodes, indirectly leading to lower transaction validation latency in an overall environment with less noise. Fig.5(b) shows the variation of the transaction validation latency with increasing number of nodes for PoMC, SENATE, and PoQF with $N_S = 10$ and $N_m = 10$. V_S is the transaction generation rate. The average transaction validation latency of PoMC is about 24% lower than PoQF, but about 4% higher than SENATE with $V_S = 0.1$. Although the transaction validation latency of PoMC increases rapidly when the number of nodes exceeds 135, the increase in transaction generation rate does not significantly affect the transaction validation latency of PoMC. This is because the primary goal of PoMC is to determine the accounting nodes. The accounting nodes continuously process blocks within their terms without the need for consensus on every block, unlike other consensus mechanisms.

In the SENATE scheme, the senator nodes are required to perform cross-validation and seesaw tests on each participating miner node to eliminate potentially malicious nodes. Furthermore, the senator nodes need to conduct Byzantine consensus internally to determine the allocation of accounting rights. This process results in low consensus efficiency for SENATE. In PoQF, We linearly increased the number of mining nodes within each hop range. Achieving consensus is faster for nodes located closer to the message origin, while nodes farther away from the source address experience longer wait times for confirmation from the next hop relay nodes. As a result, the average consensus latency performance of PoQF is inferior to that of PoMC. Based on the results shown in Fig.5(c), it can be observed that the increase in the number of nodes has a negligible impact on the consensus latency of PoMC. The consensus latency in PoMC is primarily influenced by the number of mining nodes and senator nodes. Therefore, we conducted the experiment depicted in Fig.5(d) to analyze the variation of consensus latency with an increasing number of mining nodes. PoMC demonstrates the lowest average consensus latency among the three approaches, being 39% lower than SENATE and 7% lower than PoQF. The latency trends of SENATE and PoMC are similar, but PoMC generally exhibits lower overall consensus latency compared to SENATE. As the number of mining nodes increases in PoQF, the number of microblocks to be collected in each hop also increases. This results in an accelerating growth of consensus latency in PoQF.

Fig.6(a) provides the comparison of transaction throughput and network overhead between PoMC and SENATE, PoQF, and PBFT. The network overhead of the three schemes is similar with $N_S = 10$ and $N_m = 10$, but the transaction

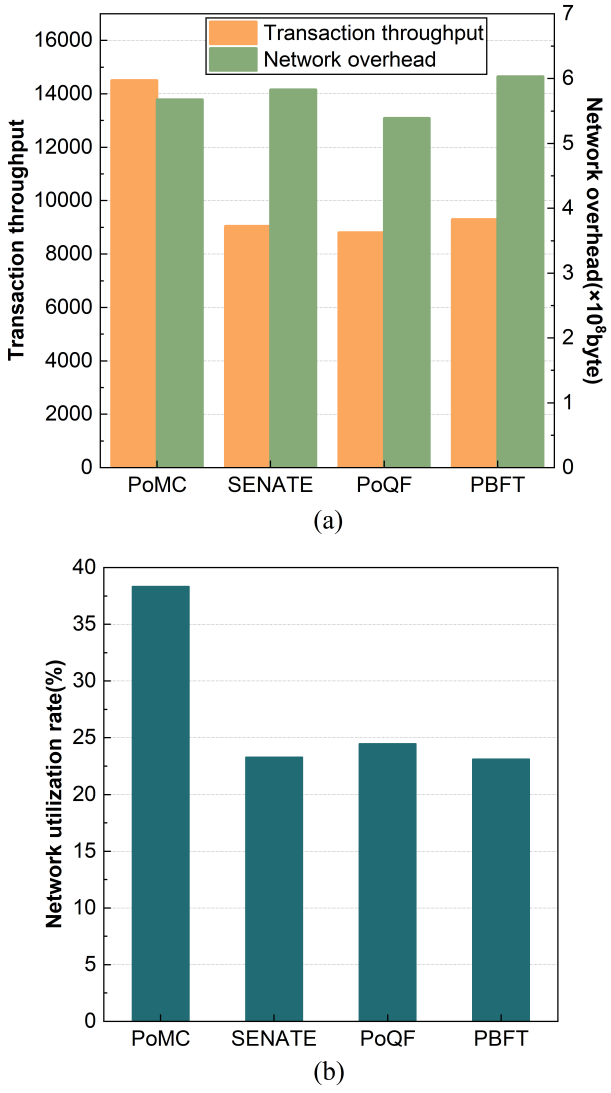

Figure 6. The comparison of system throughput. (a) Transaction throughput and network overhead. (b) Network utilization rate.

throughput of PoMC is about 56%∼64% higher compared to the other three schemes. This is attributed to the fact that PoMC does not require cross-validation of a transaction like SENATE and PBFT, thereby enhancing transaction throughput and reducing transaction validation latency. Although PoQF does not necessitate consensus for each transaction, it involves a substantial number of nodes in one consensus process when the hop count is large, thereby diminishing transaction validation efficiency. Network utilization rate R_{Net} refers to the ratio of the size E_{trans} of validated transactions to the total network overhead E_{total} in the same time, i.e., $R_{Net} = E_{trans}/E_{total}$. As shown in Fig.6(b), PoMC has the highest network utilization among the four schemes because the nodes generate transactions and send them directly to the accounting nodes instead of broadcasting them over the network, which significantly reduces the system network overhead.

The ability of the consensus mechanism to resist malicious attacks can be evaluated by the probability of successful

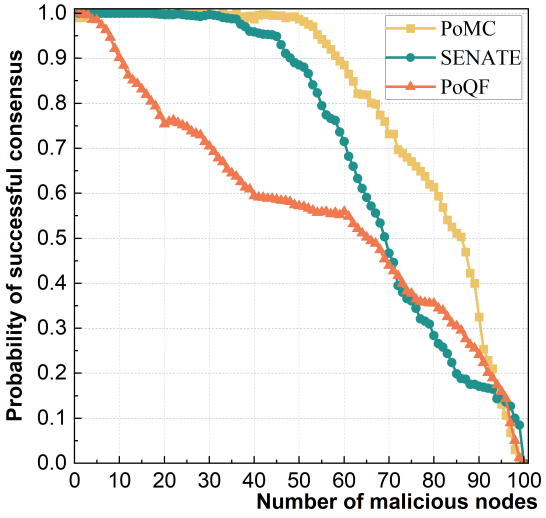

Figure 7. The variation of the probability of successful consensus with the change of number of malicious nodes.

consensus, which refers to the percentage of honest accounting nodes among all accounting nodes selected by the system in a fixed time frame. The total number of nodes is set to 100, and as shown in Fig.7, the probability of successful consensus of all the three schemes tends to decrease as the number of malicious nodes increases. PoMC has a higher consensus success rate than SENATE and PoQF as the number of malicious nodes below 90, which means PoMC outperforms SENATE and PoQF in resisting to Sybil attacks. What's more, it is observed that the probability of successful consensus of PoMC drops at a slower rate than PoQF and SENATE. That's because not only does it rely on the consensus protocol to restrain malicious nodes, but the distributed incentive mechanism also inhibits evil behaviors in PoMC.

C. Discussion

In the context of agricultural IoT applications, the proposed PoMC consensus mechanism integrates real-world SINR and abstract-world credibility as two capability factors. The system determines the ownership of accounting rights based on the multifactor capability value, leading to a significant reduction in the computational and communication resources required for agricultural IoT devices to participate in consensus. We have not only established a node credit model but also developed a lottery algorithm for senator nodes based on the credit mechanism. This design aims to enhance system security while ensuring the degree of decentralization. Furthermore, a novel distributed incentive mechanism has been devised for PoMC. This mechanism incentivizes honest nodes to actively maintain the blockchain network while simultaneously restricting malicious nodes from engaging in attack behaviors. Benefiting from the favorable performance of PoMC, there is promising potential for its application in various scenarios within the agricultural IoT. With its lower latency and reduced resource requirements, PoMC could be applied in large-scale agricultural IoT clusters to ensure the security of data

throughout the process from generation to transmission and storage.

VII. CONCLUSION

In this paper, a consensus mechanism PoMC based on Proof-of-Multifactor-Capacity is proposed to address the problems of heavy resource consumption, high transaction validation latency and low throughput of blockchain consensus mechanism in agricultural IoT. The mechanism selects some nodes in the network as senators by drawing lots based on the credibility. Two metrics, SINR and credibility, are chosen for the multi-factor capacities value to reflect the communication capability of the node and the degree of contribution to the system, respectively. Senators select accounting nodes based on the multifactor capacities value, which decreases the computational power demand and communication overhead of the nodes and reduces the malicious behaviors of abnormal nodes. From the simulation analysis, under similar network overhead, PoMC achieves a transaction throughput that is 56%∼64% higher than that of SENATE, PoQF and PBFT, respectively, thereby exhibiting better network utilization. Although the proposed scheme does not have the lowest average transaction validation latency, its consensus success rate is 10.3% and 23.9% higher than that of SENATE and PoQF, respectively, implying that PoMC has better security.

In the future work, we will explore efficient data synchronization schemes among nodes in blockchain-based systems and some methods to reduce the communication and storage overhead of blockchain networks for agricultural IoT to achieve higher consensus efficiency.

REFERENCES

[1] O. Elijah, T. A. Rahman, I. Orikumhi, C. Y. Leow, and M. N. Hindia, "An Overview of Internet of Things (IoT) and Data Analytics in Agriculture: Benefits and Challenges," *IEEE Internet Things J.*, vol. 5, no. 5, pp. 3758–3773, oct 2018.

[2] H. N. Dai, Z. Zheng, and Y. Zhang, "Blockchain for Internet of Things: A Survey," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8076–8094, oct 2019.

[3] E. Munsing, J. Mather, and S. Moura, "Blockchains for decentralized optimization of energy resources in microgrid networks," in *Annu. IEEE Conf. Control Technol. Appl., CCTA*. IEEE, aug 2017, pp. 2164–2171.

[4] C. Walsh, P. O'Reilly, R. Gleasure, J. McAvoy, and K. O'Leary, "Understanding manager resistance to blockchain systems," *European Management Journal*, vol. 39, no. 3, pp. 353–365, jun 2021.

[5] Y. Yu, Y. Ding, Y. Zhao, Y. Li, Y. Zhao, X. Du, and M. Guizani, "LRCoin: Leakage-resilient cryptocurrency based on bitcoin for data trading in IoT," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4702–4710, jun 2019.

[6] M. Pincheira, M. Vecchio, R. Giaffreda, and S. S. Kanhere, "Exploiting constrained IoT devices in a trustless blockchain-based water management system," in *IEEE Int. Conf. Blockchain Cryptocurrency, ICBC*. IEEE, may 2020, pp. 1–7. [Online]. Available: https://ieeexplore.ieee.org/document/9169404/

[7] M. Ur Rahman, F. Baiardi, and L. Ricci, "Blockchain Smart Contract for Scalable Data Sharing in IoT: A Case Study of Smart Agriculture," in *IEEE Glob. Conf. Artif. Intell. Internet Things, GCAIoT*. IEEE, dec 2020.

[8] M. Salimitari, M. Chatterjee, and Y. P. Fallah, "A survey on consensus methods in blockchain for resource-constrained IoT networks," sep 2020.

[9] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," Tech. Rep., 2008. [Online]. Available: www.bitcoin.org

[10] Y. Huang, J. Zhang, J. Duan, B. Xiao, F. Ye, and Y. Yang, "Resource allocation and consensus on edge blockchain in pervasive edge computing environments," in *Proc Int Conf Distrib Comput Syst*, vol. 2019-July. IEEE, jul 2019, pp. 1476–1486.

[11] L. Lao, X. Dai, B. Xiao, and S. Guo, "G-PBFT: A Location-based and Scalable Consensus Protocol for IoT-Blockchain Applications," in *Proc. - IEEE Int. Parallel Distrib. Process. Symp., IPDPS*. IEEE, may 2020, pp. 664–673.

[12] Z. Jiang, Z. Cao, B. Krishnamachari, S. Zhou, and Z. Niu, "SENATE: A Permissionless Byzantine Consensus Protocol in Wireless Networks for Real-Time Internet-of-Things Applications," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 6576–6588, jul 2020.

[13] X. Xu, G. Sun, and H. Yu, "An Efficient Blockchain PBFT Consensus Protocol in Energy Constrained IoT Applications," in *Int. Conf. UK-China Emerg. Technol., UCET*. IEEE, 2021, pp. 152–157.

[14] J. Misic, V. B. Misic, X. Chang, and H. Qushtom, "Adapting PBFT for Use with Blockchain-Enabled IoT Systems," *IEEE Trans. Veh. Technol.*, vol. 70, no. 1, pp. 33–48, jan 2021.

[15] H. Qushtom, J. Misic, X. Chang, and V. B. Misic, "A Scalable Two-Tier PBFT Consensus for Blockchain-Based IoT Data Recording," in *IEEE Int Conf Commun*. IEEE, jun 2021.

[16] A. Yazdinejad, G. Srivastava, R. M. Parizi, A. Dehghantanha, H. Karimipour, and S. R. Karizno, "SLPoW: Secure and Low Latency Proof of Work Protocol for Blockchain in Green IoT Networks," in *IEEE Veh Technol Conf*. IEEE, may 2020, pp. 1–5. [Online]. Available: https://ieeexplore.ieee.org/document/9129462/

[17] G. Sagirlar, B. Carminati, E. Ferrari, J. D. Sheehan, and E. Ragnoli, "Hybrid-IoT: Hybrid Blockchain Architecture for Internet of Things - PoW Sub-Blockchains," in *Proc. IEEE Conf. Internet Things, Green Comput. Commun., Cyber, Phys. Soc. Comput., Smart Data*. IEEE, jul 2018, pp. 1007–1016. [Online]. Available: https://ieeexplore.ieee.org/document/8726649/

[18] A. Anand, W. Asif, and M. Lestas, "Performance Evaluation of PoW Blockchain in Wireless Mobile IoT networks," in *Proc. - Annu. Int. Conf. Distrib. Comput. Sens. Syst., DCOS*. IEEE, 2021, pp. 396–403.

[19] L. M. Bach, B. Mihaljevic, and M. Zagar, "Comparative analysis of blockchain consensus algorithms," in *Int. Conv. Inf. Commun. Technol., Electron. Microelectron., MIPRO - Proc.* IEEE, jun 2018, pp. 1545–1550.

[20] Y. Wei, L. Liang, B. Zhou, and X. Feng, "A Modified Blockchain DPoS Consensus Algorithm Based on Anomaly Detection and Reward-Punishment," in *Int. Conf. Commun. Softw. Networks, ICCSN*. IEEE, jun 2021, pp. 283–288.

[21] J. Huang, L. Kong, G. Chen, M. Y. Wu, X. Liu, and P. Zeng, "Towards secure industrial iot: Blockchain system with credit-based consensus mechanism," *IEEE Trans. Ind. Inf.*, vol. 15, no. 6, pp. 3680–3689, jun 2019.

[22] J. Zhang, Y. Huang, F. Ye, and Y. Yang, "A Novel Proof-of-Reputation Consensus for Storage Allocation in Edge Blockchain Systems," in *IEEE/ACM Int. Symp. Qual. Serv., IWQOS*. IEEE, jun 2021.

[23] Y. Liu, K. Wang, Y. Lin, and W. Xu, "Lightchain: A lightweight blockchain system for industrial internet of things," *IEEE Trans. Ind. Inf.*, vol. 15, no. 6, pp. 3571–3581, jun 2019.

[24] L. Chen, L. Xu, N. Shah, Z. Gao, Y. Lu, and W. Shi, "On security analysis of proof-of-elapsed-time (poet)," in *Stabilization, Safety, and Security of Distributed Systems: 19th International Symposium, SSS 2017, Boston, MA, USA, November 5–8, 2017, Proceedings 19*. Springer, 2017, pp. 282–297.

[25] A. Juels and B. S. Kaliski Jr, "Pors: Proofs of retrievability for large files," in *Proceedings of the 14th ACM conference on Computer and communications security*, 2007, pp. 584–597.

[26] P. Kumar, R. Kumar, A. Kumar, A. A. Franklin, S. Garg, and S. Singh, "Blockchain and deep learning for secure communication in digital twin empowered industrial iot network," *IEEE Trans. Netw. Sci. Eng.*, vol. 10, no. 5, pp. 2802–2013, sep 2023.

[27] P. Kumar, R. Kumar, G. P. Gupta, R. Tripathi, and G. Srivastava, "P2tif: A blockchain and deep learning framework for privacy-preserved threat intelligence in industrial iot," *IEEE Trans. Industr. Inform.*, vol. 18, no. 9, pp. 6358–6367, sep 2022.

[28] P. Kumar, R. Kumar, G. Srivastava, G. P. Gupta, R. Tripathi, T. R. Gadekallu, and N. N. Xiong, "Ppsf: A privacy-preserving and secure framework using blockchain-based machine-learning for iot-driven smart cities," *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 3, pp. 2326–2341, sep 2021.

[29] S. King and S. Nadal, "Ppcoin: Peer-to-peer crypto-currency with proof-of-stake," *self-published paper*, vol. 19, no. 1, pp. 1–6, aug 2012.

[30] M. M. Islam, M. M. Merlec, and H. P. In, "A comparative analysis of proof-of-authority consensus algorithms: Aura vs clique," in *Proc. - IEEE Int. Conf. Serv. Comput., SCC*. IEEE, 2022, pp. 327–332.

[31] D. Huang, X. Ma, and S. Zhang, "Performance analysis of the raft consensus algorithm for private blockchains," *IEEE Trans. Syst. Man. Cybern. Syst.*, vol. 50, no. 1, pp. 172–181, jan 2019.
[32] B. Alangot, D. Reijsbergen, S. Venugopalan, P. Szalachowski, and K. S. Yeo, "Decentralized and Lightweight Approach to Detect Eclipse Attacks on Proof of Work Blockchains," *IEEE Trans. Netw. Serv. Manage.*, vol. 18, no. 2, pp. 1659–1672, jun 2021.
[33] W. Guo, J. Xu, Y. Pei, L. Yin, and C. Jiang, "LDBT: A Lightweight DDoS Attack Tracing Scheme Based on Blockchain," in *IEEE Int. Conf. Commun. Workshops, ICC Workshops - Proc.* IEEE, jun 2021.
[34] M. Haenggi, "Twelve reasons not to route over many short hops," in *IEEE Vehicular Technology Conference*, vol. 60, no. 5, 2004, pp. 3130–3134.
[35] Y. Sun, L. Zhang, G. Feng, B. Yang, B. Cao, and M. A. Imran, "Blockchain-enabled wireless internet of things: Performance analysis and optimal communication node deployment," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 5791–5802, jun 2019.
[36] N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero, R. L. Moses, and N. S. Correal, "Locating the nodes: Cooperative localization in wireless sensor networks," *IEEE Signal Process Mag.*, vol. 22, no. 4, pp. 54–69, 2005.
[37] S. Asiri and A. Miri, "A Sybil Resistant IoT Trust Model Using Blockchains," in *Proc. IEEE Conf. Internet Things, Green Comput. Commun., Cyber, Phys. Soc. Comput., Smart Data*. IEEE, jul 2018, pp. 1017–1026. [Online]. Available: https://ieeexplore.ieee.org/document/8726529/
[38] Z. Ma, L. Wang, and W. Zhao, "Blockchain-Driven Trusted Data Sharing with Privacy Protection in IoT Sensor Network," *IEEE Sens. J.*, vol. 21, no. 22, pp. 25 472–25 479, nov 2021.
[39] J. Kolb, M. Abdelbaky, R. H. Katz, and D. E. Culler, "Core concepts, challenges, and future directions in blockchain: A centralized tutorial," *ACM Comput. Surv.*, vol. 53, no. 1, feb 2020.
[40] A. Durand, E. Ben-Hamida, D. Leporini, and G. Memmi, "Asymptotic Performance Analysis of Blockchain Protocols," feb 2019. [Online]. Available: http://arxiv.org/abs/1902.04363
[41] F. Ayaz, Z. Sheng, D. Tian, and Y. L. Guan, "A Proof-of-Quality-Factor (PoQF)-Based Blockchain and Edge Computing for Vehicular Message Dissemination," *IEEE Internet Things J.*, vol. 8, no. 4, pp. 2468–2482, feb 2021.

Shuming Xiong received the B.E., M.E., and Ph.D. degrees from Jiangsu University, Zhenjiang, China, in 1998, 2003, and 2011, respectively.,He is an Associate Professor with the School of Computer Science and Communication Engineering, Jiangsu University. He has published over 20 papers in WSNs and IoT. His research interests mainly include data security, connectivity control in WSNs, IoT, and other wireless networks.

Zeyu Wang is a Master student in Jiangsu University, Zhenjiang, China. His current research interests include blockchain and Agricultural Internet of Things.

Qiang Ni received the B.Sc., M.Sc., and Ph.D. degrees in engineering from the Huazhong University of Science and Technology, China. He led the Intelligent Wireless Communication Networking Group, Brunel University London, U.K. He is currently a Professor and the Head of the Communication Systems Group, InfoLab21, School of Computing and Communications, Lancaster University, Lancaster, U.K. His main research interests lie in the areas of wireless communications and networking, including green communications, cognitive radio systems, 5G, the Internet-of-Things, and vehicular networks. He was an IEEE 802.11 Wireless Standard Working Group Voting Member and a Contributor to the IEEE Wireless Standards.

Xue Han is a Master student in Jiangsu University, Zhenjiang, China. Her current research interests include blockchain and Internet of Things security.