A Fast and Secured Vehicle-to-Vehicle Energy Trading Based on Blockchain Consensus in the Internet of Electric Vehicles

Yingsen Wang, *Student Member, IEEE*, Leiming Yuan, Weihan Jiao, Yan Qiang, *Graduate Student Member, IEEE*, Juanjuan Zhao, Qianqian Yang, and Keqin Li, *Fellow, IEEE*

***Abstract*—The organization and management of electricity markets worldwide are rapidly evolving, moving towards decentralized, distributed, and renewable energy-based generation with solutions based on real-time data exchange. A Vehicle-to-Vehicle (V2V) energy trading has emerged as one of the most promising alternatives for relieving the load imposed on the traditional grid enabling two individuals to buy and sell energy directly without intermediaries. However, the Internet of Electric Vehicles (IoEV) environment is trustless, and such P2P energy trading is prone to different kinds of cyber attacks. Blockchain technology has lately been proposed to implement V2V energy trading to securely and fairly share energy. The consensus mechanism is one of the most important modules of blockchain applied to the V2V network. It determines the efficiency and security among untrustworthy EVs of the energy trading blockchain (ETB). Nevertheless, most works on ETB have currently adopted traditional consensus mechanisms. Due to high computing power and communication overhead, these consensus algorithms are unsuitable for applications requiring real-time services such as energy trading. The efficient and secured Hashgraph is the revolutionary technology of consensus in blockchain and a promising technology suitable for V2V energy trading with frequent transactions. However, Hashgraph does not support the dynamic addition and deletion of nodes and is completely decentralized and vulnerable to Sybil Attack in a large-scale blockchain. Furthermore, this "complete decentralization" model may result in states losing the ability to macro-control the energy industry and even systemic energy security issues. Therefore, we propose a Block Alliance Consensus (BAC) mechanism to solve these problems. BAC can maintain the throughput of Hashgraph and resist Sybil Attack in a large-scale P2P energy trading network. We design a cryptography-based leader election mechanism and adopt a reputation incentive mechanism to motivate honest and cooperative electric vehicles (EVs). Finally, we implement ETB and the BAC consensus mechanism on the Hyperledger Fabric platform. The high efficiency and security of BAC and the blockchain-based V2V energy trading platform are verified through experiments.**

***Index Terms*—Blockchain, consensus, electric vehicles, energy trading, Hashgraph, Hyperledger Fabric, Internet of Electric Vehicles, Vehicle-to-Vehicle.**

I. Introduction

THERE are centralized third-party control centers in the traditional energy trading markets to complete all energy management processes and decide the power generation, transmission, distribution, and delivery. Sharing data on these centralized servers involves various security and privacy issues, resulting in users not being motivated to share their data [1]. In addition, these centralized servers are prone to a single point of failure, and problems with the central server might bring the whole network to a halt [2]. In recent years, traditional energy trading systems have begun to look beyond their capabilities due to the surge in electricity and the promotion of new energy power generation [3].

EVs have emerged as an effective way to satisfy energy demands by using renewables such as solar and wind power. The emergency of EVs is a promising alternative for improving resource usage, eliminating hazardous emissions, and maximizing revenue [4]. Moreover, traditional grids are evolving into smart grids, including information and communication technology, electronic devices, and interconnected power systems to maximize the usage of renewable energy supplies and alleviate energy problems in some way [5], [6], [7].

A V2V energy network allows EVs to dynamically charge their batteries with additional energy from the boards of other EVs while both sellers and buyers are in motion [8]. A V2V energy network could significantly minimize EVs' range anxiety while requiring minimum infrastructure investment. Though decentralized V2V energy trading could solve the problems in the traditional structure, it brings new difficulties such as security, privacy, and trust nightmare, which need the development of new technologies [9]. The application of blockchain to V2V energy trading has become a promising technology. It offers new opportunities to enable secure energy transfer between energy buyers and sellers and helps curb the penetration and disruption

Manuscript received 7 June 2022; revised 30 September 2022 and 22 November 2022; accepted 7 January 2023. Date of publication 26 January 2023; date of current version 20 June 2023. This work was supported by the National Natural Science Foundation of China under Grants U21A20469, 61872261, and 61972274. The review of this article was coordinated by Prof. Amiya Nayak. *(Corresponding author: Yan Qiang.)*

Yingsen Wang, Leiming Yuan, Yan Qiang, and Juanjuan Zhao are with the College of Information and Computer Science, Taiyuan University of Technology, Taiyuan 030024, China (e-mail: wangyingsen0065@link.tyut.edu.cn; yuanleiming0453@link.tyut.edu.cn; 552383989@qq.com; zhaojuanjuan@tyut.edu.cn).

Weihan Jiao is with the School of Electric and Electronic Engineering, North China Electric Power University, Beijing 102206, China (e-mail: 120212201610@ncepu.edu.cn).

Qianqian Yang is with the Jinzhong College of Information, Jinzhong 030605, China (e-mail: evie_yang@qq.com).

Keqin Li is with the Department of Computer Science, State University of New York, New Paltz, NY 12246 USA (e-mail: lik@newpaltz.edu).

Digital Object Identifier 10.1109/TVT.2023.3239990

of cyber attacks [10]. Blockchain is a decentralized, distributed, and immutable ledger made up of an irrevocable sequence of blocks [11]. It allows mutually distrustful vehicles to keep transparent transaction records. Attackers in blockchain must possess a majority of the network's mining power to conduct a successful attack [12]. Although the blockchain originated from digital currencies, it is now used in many other non-monetary scenarios. Blockchain is attracting enormous attention to P2P energy trading and promoting trusted smart grid developments toward decentralization.

As one of the most vital parts of the blockchain, the consensus mechanism is the core technology enabling distributed nodes to agree on the new block waiting to be published to the blockchain [13]. It maintains trust between distrustful EVs and is for fault-tolerant to agree on the same state of the blockchain network, such as a single state of all transactions in a cryptocurrency blockchain [14]. The efficiency and security of the blockchain depend significantly on the consensus module, impacting the reliability and scalability of the ETB [15]. The Byzantine problem is always the most challenging in the distributed consensus protocol. Researchers have developed popular Byzantine Fault Tolerance (BFT) based consensus protocols, including Proof-of-Work (PoW), Proof-of-Stake (PoS), and Practical Byzantine Fault Tolerance (PBFT). However, consensus mechanisms designed for the V2V blockchain in the IoEV are still rare [16]. Besides, PoW and PoS are mostly adopted in traditional blockchains, primarily for digital currencies. They are not suitable for Internet of Things (IoT) applications such as energy trading in the IoEV. In addition, many studies have adopted PBFT in the V2V network. However, the communication complexity of PBFT is too high to be directly used in a P2P network. In a word, most consensus mechanisms currently utilized in V2V energy trading are not efficient due to their great computational resources and communication complexity [6], [15].

Therefore, we are encouraged to develop a secure and efficient consensus mechanism for the V2V ETB implemented in the IoEV based on BFT to solve the aforementioned issues in the energy trading process. Hashgraph is a revolutionary distributed ledger technology and a promising consensus for energy trading. However, the total number of nodes N is a fixed value that needs to be preset, i.e., the dynamic addition and deletion of nodes are not permitted in Hashgraph. In addition, the roles and statuses of nodes in energy trading are different. In energy trading, ordinary households, regulators, power grid departments, etc., have their responsibilities and roles. The purpose of P2P energy trading is efficiency, security, and ease of the load on the grid, rather than blindly pursuing decentralization. The significance of decentralization is that it can ensure security and improve efficiency. So in this study, we propose a consortium blockchain structure for energy trading and a data Block Alliance Consensus (BAC) algorithm to ensure the data's accuracy and reliability and the blockchain network's security. The main contributions of our study are as follows:

1) A distributed V2V Energy Trading Blockchain (ETB) structure is proposed; all the participants record and maintain the ledger together. We utilize the blockchain sharding technique and design a detailed structure of nodes in ETB. The nodes of the V2V energy trading are classified into three categories: Primary Node (P), Candidate Primary Node (CP), and Consensus Node (CS) in each shard according to EVs' location, velocity, and direction. The primary election, block validation and propagation are performed in each shard.
2) We propose the Hashgraph-based Block Alliance Consensus (BAC) algorithm suitable for the V2V ETB. The BAC reduces the time complexity of traditional BFT to $O(N)$, where N is the number of EVs. BAC can significantly improve the throughput and security of ETB like Hashgraph, support the dynamic addition and deletion of nodes, and prevent the V2V ETB from Sybil Attack in a large-scale network, which is not available in Hashgraph. BAC does not rely on high computational power, which is suitable for resource-constrained industrial ETB.
3) We design a primary and candidate primary election mechanism based on cryptography and reputation. Cryptography guarantees the randomness of the elected primary EVs, thus preventing attackers from predicting and attacking the future primary EVs in advance. The election mechanism is not completely random, and EVs with high reputation values are more likely to be elected leaders. The experimental results show that the election mechanism we design guarantees the security of the V2V system and motivates EV users.
4) We implement the V2V ETB on the Hyperledger Fabric. We first write a python program combined with a blockchain-dedicated simulator to demonstrate our superior performed BAC mechanism. Then we add the BAC to the consensus module of the Fabric. The Hyperledger Caliper experimental results show the security and efficiency of the V2V ETB.

The remainder of the paper is organized as follows. Section II presents the background and related works. Section III introduces the system model of the V2V ETB, including the network and node model, ETB sharding technique, and blockchain-enabled V2V energy trading. Section IV discusses our BAC consensus mechanism, and Section V provides the performance evaluation of our BAC consensus mechanism and the ETB platform. The last section summarises our study and describes our future research.

II. Background and Related Works

This section reviews relevant research efforts on energy trading, blockchain in V2V, and consensus mechanisms commonly adopted in the V2V ETB based on the BFT.

A. V2V Energy Trading and Blockchain

Several researchers have proposed innovative energy trading schemes for V2V and vehicle-to-grid (V2G). We move this subsection to the Appendix A because of the page limit.

B. Consensus Mechanism Based on Byzantine Fault Tolerance

The consensus mechanism is one of the most vital parts of the blockchain. It is the core but the bottleneck of distributed

TABLE I
COMPARISON WITH VARIOUS IMPORTANT CONSENSUS MECHANISMS FOR V2V ENERGY TRADING NETWORK

	PoW	**PoS**	**DPoS**	**PBFT**	**Hashgraph**
Node Management	Permissionless	Permissionless	Permissionless	Permissioned	Permissioned
Transaction Latency	High	Low	Low	High	Low
Throughput	Low	High	High	Low	Extremely High
Energy Consumption	High	General	Low	Low	Low
Fault Tolerance	1/2	1/2	1/2	1/3	1/3
Scalability	General	General	General	Poor	General

ledger technology. The inherent encryption characteristics of blockchain ensure those data blocks that already existed in the blockchain can not be tampered with, while consensus methods provide the validity of the new data block. Table I demonstrates the differences between the important consensus mechanisms frequently used in V2V energy trading. We move the detailed descriptions of Table I to the Appendix B because of the page limit. The significance of BFT is to solve the consensus of decentralized systems. BFT originated from the Byzantine Generals Problem (BGP) proposed by Lamport [17]. The BGP is a common challenge that decentralized computer systems must overcome. The generals must devise a unified combat strategy while communicating just via messenger. One or more of them may, however, be traitors who will attempt to mislead the rest. The objective is to develop an algorithm to guarantee loyal generals achieve a consensus. The BGP is insurmountable with three generals. However, the asynchronous BGP was not considered. The time threshold t in the traditional BFT (BGP) is a fixed constant value, but there is no concept of the time threshold in asynchronous systems. Then came the Practical Byzantine Fault Tolerance (PBFT). The approach taken by PBFT [18] is as follows: the threshold t in PBFT will increase if the system times out. It ensures that no matter how considerable the system's delay is, nodes in PBFT can eventually reach a consensus as long as the delay does not increase indefinitely.

The development of BFT seems to have reached its end - problems that PBFT cannot solve are only theoretical problems and are unlikely to appear in practice. Then the proposal of Zyzzyva [19] further promoted the development of BFT. "Zyzzyva" is the last word in the dictionary. The author believed that the algorithm might be the final solution to the BFT problem. The time complexity of PBFT is $O(N^2)$. Zyzzyva is a speculative-based algorithm: if the primary node is credible, there is no need for such a complex algorithm; it is enough to do a round of regular broadcasts with $O(N)$ message complexity. If nodes find the primary faulty (malicious or down), the system returns to the PBFT algorithm. We move the detailed description of BGP and Zyzzyva to the Appendix M because of the page limit.

Finally came the advent of Bitcoin (blockchain), a milestone for the development of BFT. Bitcoin adopts PoW as its consensus mechanism, and the incentive mechanism is the genius of Bitcoin. A cost is attached to each block in terms of computing power, and the incentive mechanism is used to motivate honest nodes and penalize malicious ones. The idea of Bitcoin extends the BFT problem to a field that BFT has never dabbled in before - consensus in large networks.

C. Consensus Mechanism in V2V Energy Trading

Some consensus mechanisms were proposed to enhance the security and efficiency of the blockchain after the advent of Bitcoin. Known as the blockchain 2.0 era, Ethereum has presented the Proof of Stake (PoS) [20] consensus mechanism to address the issues of wasteful resources in PoW. Similarly, Ekparinya et al. [21] proposed Proof of Authority (PoA), which utilizes a fair incentive mechanism to ensure that most nodes remain online. Online miners get a certain probability of gaining revenue even if they do not possess strong computing power.

Current research efforts on ETB have been adopting traditional consensus algorithms. For example, Sun et al. [10] utilized the PBFT-based Delegated PoS (PDPoS) in IoEV. Su et al. [22] proposed a reputation-based BFT to efficiently reach consensus in the permissioned energy blockchain. Feng et al. [23] introduced a scalable, dynamic multi-agent hierarchical PBFT method (SDMA-PBFT) that decreases the communication overhead from $O(N^2)$ to $O(nk\log_k n)$. Yang et al. [24] presented a PBFT-based algorithm for multi-energy interactive entities.

Although the above research solves the problem of low participation of nodes, the issue of high transaction delay and low throughput has not been completely solved. Existing consensus mechanisms still have a significant gap in achieving the security and efficiency of the ETB. Most studies implement the consensus mechanism as a small part of their research, and most adopt the traditional or improved PBFT as their consensus mechanism. They are incapable of meeting the requirements of large-scale energy transactions. As a result, the current energy trading platform urgently needs to improve the performance of the consensus mechanism.

III. SYSTEM MODEL AND PRELIMS

Fig. 1 shows the proposed ETB system model. We group the EVs into different shards based on their location, direction, and velocity. Each shard consists of sellers, buyers, and block validators. We have utilized the Hyperledger Fabric, a consortium blockchain platform with smart contract (SC), to guarantee the security, efficiency, and transparency of our ETB. The characteristic of consortium blockchains ranks between public blockchains and private blockchains. Consortium blockchains are more secure than public and more transparent than private blockchains. While a single institution maintains a private blockchain, a consortium blockchain is controlled by specific groups (shards), where only authorized nodes can participate in the consensus process. The V2V consortium blockchain allows energy buyers continually interact with the energy trading

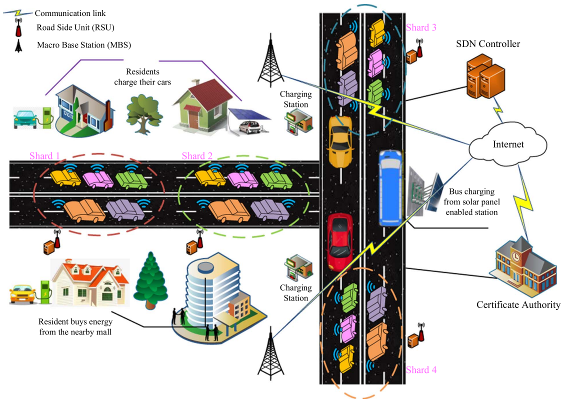


Fig. 1. System model for V2V energy trading blockchain.

system to charge their vehicles by choosing the most reasonable offer among the bids. Energy sellers could purchase electricity directly from the power grid at a relatively low price or obtain electricity from renewables such as solar and wind. Then energy sellers could sell electricity to other vehicles at maximum revenue prices.

The SC is a digital version (in the form of code) of traditional contracts deployed in EVs of ETB. It is the chain code in the blockchain to automate the implementation of rules and terms to settle claims automatically. Overall, the SC improves accountability, interoperability, and trustworthiness for all participants in the decentralized blockchain environment. Smart Meters (SM) are used by both buyers and sellers to capture, meter, and transmit the consumed and produced energy. SM is a critical component of the V2V energy system that tracks energy transfers from a predefined energy purchase and sale agreement between buyers and sellers.

A certificate authority (CA) is implemented to manage the identity certificate of each entity (node) in the ETB. The CA is responsible for managing the network infrastructure, registering all entity identities in the blockchain, issuing digital certificates, and renewing or revocation of certificates. All nodes that join the ETB must be registered and obtain a certificate issued by CA. CA maintains a list of registered EVs and their associated SCs, which are used to enforce entities to comply with their agreements. Users could charge their EVs from nearby charging stations. The road side unit (RSU) with microwave antenna and read/write controller and the macro base station (MBS) are implemented to provide wireless communication for EVs. The Software Defined Network (SDN) defines and controls the V2V network through software programming, dynamically controlling traffic flows for maximum performance benefits. Data and control are separated in V2V SDN networks, providing substantial flexibility, security, and reliability. The "data control separation" feature also makes SDN networks simple to upgrade and extend [25]. We move the detailed descriptions of SDN to the Appendix C because of the page limit.

A. Network Model

The transaction data and virtual trading procedure of V2V are illustrated in Fig. 2. The energy transfer procedure is initiated between EV users and power grids to address the issue of demand-supply mismatch. The power grid supplies energy to users. However, EV users' demand for energy is intermittent, resulting in unstable operation of the power grid. The grid is overloaded during peak power consumption, and the excess demand could be provided by extra power from consumers. In "General Procedure of Blockchain", the consensus is the process of data transmission and verification, which is the soul of the blockchain.

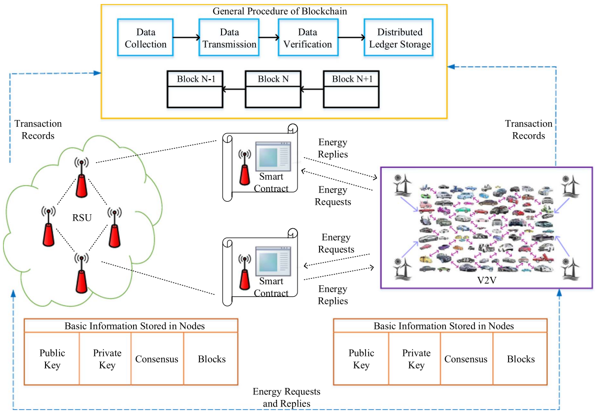


Fig. 2. The blockchain-based V2V energy trading process.

In V2V ETB, EVs connected to RSUs are sharded based on their state (current driving information) and roles in energy trading. Along with acting as buyers or sellers, EVs can act as block validators and generators. Block generators are EVs elected to the committee (specifically, the leader), whereas block validators serve as universal EV supervisors. In practice, the block generator can be the EV and a "central node" in the government or technology sector to maintain national macroeconomic control over the energy economy. Let $i \in S = \{1, 2, \ldots, A\}$ be an energy seller, and $j \in B = \{1, 2, \ldots, K\}$ be an energy buyer. The block validators include candidate primary nodes and consensus nodes (we will discuss in detail in subsection B, Node Model), $k \in C = \{1, 2, \ldots, L\}$ denote the block validator in each shard. We move the detailed energy trading process between buyer and seller to Appendix G because of the page limit. There are no specific equations for $role_q \in \{-1, 1\}$ with its outputs as 1 or -1. The outputs are the judging conditions used to determine EV users' roles.

In our Fabric V2V ETB, each EV needs a distinctive authenticated identification (ID) to be a blockchain member legally. CA manages the identity certification of each EV within the V2V network. Each EV is required to register through CA to obtain a new account and a pair of encryption keys (public key and private key) that could be uniquely identified in the ETB. A seller S_i must demonstrate that it has enough energy to sell to obtain the unique certificate. A buyer B_j must prove that it has enough electronic money in its e-wallet to buy the power it needs. Participants obtain their ID certificates $Cert_{S_i}$, $Cert_{B_j}$, and $Cert_{C_k}$ respectively from CA. A B_j could participate in the ETB with its ID certificate and the encryption keys pair (PK_{B_j}, SK_{B_j}), and e-wallet address $AddB_j$. The account of B_j includes its account balance Bal_{B_j}, certificate $Cert_{B_j}$,

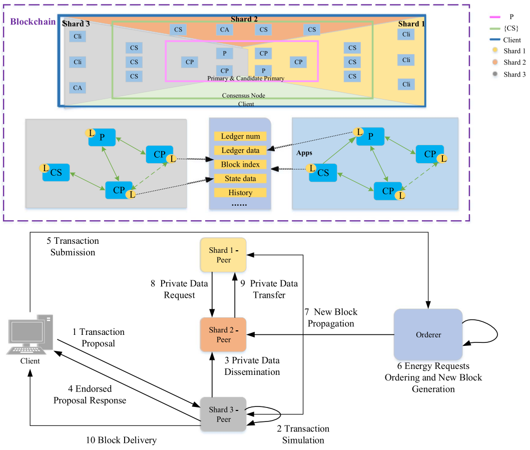


Fig. 3. The deployment of the shard and EV nodes.

current energy coin value e_j, encryption keys pair (PK_{B_j}, SK_{B_j}), and e-wallet address $AddB_j$. Similarly, the account of S_i's account includes its account balance Bal_{S_i}, available energy AV_i, encryption keys pair (PK_{S_i}, SK_{S_i}), and e-wallet address $AddS_i$. We utilize the asymmetric encryption technology [26] as (1) to guarantee the authenticity and validity of the message (transaction or block) between sender and receiver:

$$D_{PK_x}(Sig_{SK_x}(H(m))) = H(m) \tag{1}$$

where $Sig_{SK_x}()$ denotes a message sender x's digital signature using its private key, $D_{PK_x}()$ denotes that message receivers could decrypt the message hash value using x's public key, and $H(m)$ denotes the hash value of the message m.

Furthermore, we utilize the verifiable random function (VRF) [27] to randomly elect the primary node according to its reputation value without interaction. VRF is divided into two parts: proof generation and verification. The generation process is expressed as:

$$P = VRF_{proof}(SK, M) \tag{2a}$$

$$R = VRF_{P2H}(P) \tag{2b}$$

where M is the original input message of an EV user (EV_1), P is the proof generated by the SK and M, and $P2H$ is the process of converting the proof to hash value.

Other EV users utilize the EV_1's public key PK to check whether the P is a proof generated from the original message M: $VRF_{verify}(PK, M, P)$. In our V2V consortium blockchain, the block generator (leader) generates a random value and proof stored in the generated block. Block validators verify the proof and random value when they receive the block.

B. Node Model

The consensus in blockchain refers to nodes having to agree on the validity of a block. The BAC consensus mechanism is proposed in the context of the consortium blockchain. The detailed structure of nodes in "General Procedure of Blockchain" in Fig. 2 is illustrated in Fig. 3. Cli refers to the client that initiates transaction requests. Ledger (L) is a channel's chain and current state data maintained by each node in the channel. The role of the channel is to realize the isolation of the business in the blockchain. The channel could be understood as the private blockchain. A consortium blockchain has multiple channels, and each channel represents a business. Members in the channel are organizations within the consortium blockchain, and an organization can join multiple channels. We move the detailed descriptions of Fig. 3 to the Appendix D because of the page limit.

The V2V energy trading allows an EV to verify the record's validity without relying on intermediaries. The EVs of our ETB are divided into three categories: P, CP, and CS. The ETB is deployed in a consortium blockchain, i.e., a permissioned blockchain. Each EV participating in ETB must be authenticated, whereas EVs without permission can not join. This inherent property of the consortium blockchain initially ensures the security of the ETB. The shard of Fabric blockchain in Fig. 3 could be interpreted as the organization that manages a series of cooperative enterprises. The ETB allows organizations to participate in multiple independent blockchain networks simultaneously through channels, which provide effective infrastructure sharing while maintaining data and communication privacy. Our main work focuses on the consensus mechanism between nodes, which is currently the most crucial property in the ETB.

The EV energy buyers and sellers are considered to be in two-dimensional space. Their location coordinates could be expressed as (r_i, g_i), and (r_j, g_j), $\forall i \in S, \forall j \in B$, where (r_i, g_i) denotes the location of i^{th} seller, and (r_j, g_j) denotes the location of j^{th} buyer. Then the Euclidean distance between supplier i and requester j in each shard is expressed as:

$$d_{i,j} = \sqrt{(r_j - r_i)^2 + (g_j - g_i)^2}, i \in S, j \in B \tag{3}$$

The effect of Cross-Channel Interference (CCI) [16] between energy sellers and buyers is examined. The Signal-to-Interference-plus-Noise Ratio (SINR) of the j^{th} buyer connected to the i^{th} seller is expressed as:

$$l_{i,j} = \frac{SPR(i,j)}{IE_{Agg}(i) + N_0} \tag{4}$$

where $SPR(i,j)$ represents the signal power acquired by the j^{th} buyer from the i^{th} seller, $IE_{Agg}(i)$ represents the aggregate interference experienced by the j^{th} buyer, N_0 denotes the Gaussian noise's power spectral density [28]. Moreover, the transmission rate from the seller i to buyer j is represented as:

$$\delta_{i,j} = W \log_2 \left(1 + \frac{\aleph g_{ij}}{\digamma \varpi^2}\right) \tag{5}$$

where W is the obtainable bandwidth for each connection, $\aleph$ signifies the channel power gain for power conversion from the seller i to buyer j, $\digamma$ indicates the SINR disparity, and ϖ^2 represents the receiver's noise power.

C. Power System Model

We move this subsection to the Appendix E because of the page limit.

D. Sharding Technique and V2V Energy Trading Blockchain

We move this subsection to the Appendix F because of the page limit.

E. Threat Model

The threat model is used to identify the cyber security risks and possible attacks to which our V2V ETB is exposed, so that we can determine which threats need to be addressed and develop solutions accordingly. Two different attacks that could affect our V2V ETB system are described as follows.

1) Denial of Service (DoS) Attack: The DoS Attack occurs when a malicious node attempts to prevent the blockchain system from offering services to authorized users. In V2V ETB, a malicious node can make a request to the target node with little resource consumption, while the target node may need to consume a significant amount of resources to process and reply to the request. The traditional BFT system is challenging to resist DoS Attack because the traditional BFT system has a precise sequence of the block proposer. If a node replies with a delay, other nodes cannot determine whether the problem is with the network or the node is down due to a DoS Attack. This will reduce the number of available nodes and make the system more dangerous. There are two circumstances in which nodes in blockchain systems are resistant to DoS Attack. The first case is public or consortium blockchains. Public and consortium blockchains are less vulnerable to DoS Attack because they are completely decentralized, and there is no central server in the public blockchain that can be attacked by malicious nodes. Even if malicious nodes conduct attacks on specific nodes, the failure of a few nodes has little impact on the overall system's availability. The second case is a blockchain system where the order of the block proposers is uncertain, so a malicious node that wants to perform a DoS Attack cannot have a clear target.

2) Sybil Attack: In V2V ETB, an attacker may disguise a node as multiple nodes to enter the blockchain network, and in ETB the attacker has multiple identities; such an attack is known as a Sybil Attack. The malicious node enters the network by forging multiple identities to interfere with routing and disrupt message delivery. The Sybil Attack is characterized by using one node to forge multiple identities to enter the network, so the witch attack can be avoided by raising the threshold for nodes to enter the blockchain network. Consortium blockchains can effectively defend against Sybil Attack, and any node entering a consortium blockchain network needs to pass authentication to enter the network.

IV. Proposed BAC Consensus Mechanism

The grid cannot supply electricity normally due to the high load on the grid during peak electricity consumption. The energy needed for an EV and extra power required to meet the peak load could be obtained from other EVs, requiring two-way (P2P) communication between suppliers and demanders and between the grid and users. An immutable data ledger needs to be formed to ensure that the P2P bidirectional network operations are not affected by malicious attacks. It can be achieved through the blockchain. The safe operation of the blockchain needs to rely on the joint decision-making of equal EVs in the blockchain on a transaction, which is the consensus mechanism in the blockchain. The BFT-based consensus mechanism refers to how the nodes in the blockchain reach a consensus in the presence of Byzantine nodes (evil or downtime). The decision taken by these equal EVs affects stages in energy trading. Similarly, the judgment of a Byzantine army's commander to assault or withdraw determines the success or failure of the operation, which inspires the similarities between V2V energy trading and BFT.

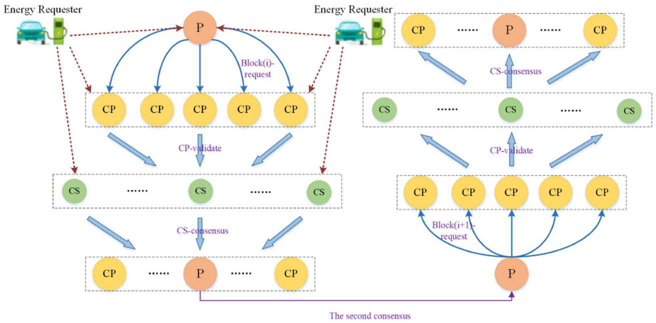


Fig. 4. The basic BAC consensus process.

A. BAC Consensus Principle and Process

This subsection discusses the principle of the proposed BAC model and the Block Consistency process.

1) The EV P in a shard packages energy transaction messages from energy requesters into a $block(i)$ and calculates the hash value. The i denotes the height of a block in the ETB. The EV P and CP store the whole block (both block header and block body). Most EV CS only need to store the block header, and a few CS store the whole block. In V2V energy trading, some residential EV users only need to save the block header, while some commercial EVs and those with a narrow range of activities for a certain period could choose to keep the whole block's information (they could gain additional revenue by competing to be elected as the P or CP, which stores all the information about the block).
2) The CP verifies the block from the P and checks whether the transaction data contained in $block(i)$ is correct. Then CP calculates the hash value of the block and compares it with the $block(i)$'s hash value.
3) A CS receives the block from CP, verifies the block information like the CP does and verifies the authentication messages sent by P to prevent CP from doing evil (CP is a Byzantine node). The $block(i)$ will be formally published into the blockchain in the second round of consensus if the $block(i+1)$ is valid.

Fig. 4 shows the basic BAC consensus model. Both malicious (tampering with data) and faulty (system downtime) EV nodes are considered at each stage of the BAC. The details of the basic BAC consensus mechanism are as follows:

1) The EV requesters broadcast the energy transaction information to all nodes. The shard leader P collects transactions from the energy demanding EVs based on their current position and driving route, packages them into the current $block(i)$, and broadcasts the $block(i)$ to CP in the shard with its signature for validation.
2) *Block-request:* A CP verifies the $block(i)$. Let $block(i) = (header(i), data(i))$ denote an unvalidated block that requests to be connected at height i in the blockchain. Let $BLOCK(i)$ denote a validated block at height i in the ETB. A CP receives $block(i)$ from the P. The CP validates $block(i)$ as follows. The basis of the validation is the $Prev\_Hash$ stored in $header(i)$. The CP calculates the hash $H(II)$ of $BLOCK(i-1)$ which the CP has already received and validated. If $H(II) \neq Prev_{H}ash$ the CP declines to validate $block(i)$ because the CP suspects that the P tampered with the data in $block(i)$. If $H(II) = Prev\_Hash$, the CP broadcasts the authenticated message $\langle$CP $-validate$, CP$(j), H(block(i))$,CP (r), CP$(c)\rangle_{\sigma_j}$ to CS, where CP(j) denotes the j^{th} CP, CP(r) denotes the reputation of the CP, CP(c) indicates whether the CP accepts the Block-request sent by P, and σ_j denotes the authenticated message signed by the j^{th} CP.
3) P and CP keep the complete blockchain. Blocks published to the V2V ETB need double rounds of consensus by the whole network. A CS collects $\langle$CP $-validate\rangle$ message set, verifies the authenticity of transaction proposals in the $block(i)$, and broadcasts the authenticated message $\langle$CS$-consensus, H(block(i))$,CS (i), CS(r),CS$(c)\rangle_{\sigma_i}$ to P.
4) *Block-commit:* The V2V ETB completes the first round of consensus in the BAC Block-commit stage. The P and CP collect block authenticated messages broadcast from CS. A CS may act in a Byzantine behavior: it deliberately withholds (does not broadcast) the authenticated message for the $block(i)$ or cannot send a message when it fails (downtime). The P and CP obey the "majority" rule in the Block-commit stage: the block can be officially published in the blockchain as long as P and CP receive sufficient authenticated messages from more than 50% of CS.
5) *Block-on-chain:* As shown in the above process, the pairwise communication between CS is avoided in the BAC. However, it results in only the P and CP knowing whether the whole network has agreed on the $block(i)$. The CS cannot acquire the consensus result. So the new $block(i)$ cannot be published into the ETB authentically in the first round of BAC consensus. In the second round of consensus, the P broadcasts the $block(i+1)$ which contains the hash value of the $block(i)$ in the $block(i+1)$'s header. A CP checks whether the $H(block(i))$ in the $block(i+1)$'s header is the same as the hash value of the CP's local $block(i)$ as **Block-commit** described. A CS receiving sufficient $\langle$CP$-validate\rangle$ messages from P and CP proves that the $block(i)$ has been approved by the "majority" of CS and can be officially published into the blockchain. It can be concluded that the new block in Bitcoin requires the confirmation of six blocks before it can be published into the blockchain, while the new block in the V2V ETB only needs the confirmation of two blocks. Algorithm 1 illustrates the details of the BAC basis process. (→: broadcasting authenticated messages; ←: receiving authenticated messages)

Algorithm 1: Block Consistency.

```
Input: $T$: a set of transactions; $t$: $t \in T$; $H()$: the hash
       function; $P = \{\mathrm{P}, \mathrm{CP}_1, \mathrm{CP}_2, ..., \mathrm{CP}_\mathrm{p}\}$;
       $C = \{\mathrm{CS}_1, \mathrm{CS}_2, ..., \mathrm{CS}_\mathrm{s}\}$
Output: $out$
$P, C \leftarrow T$;
while P calculates $H(I) = H(block(i-1))_\mathrm{P}$ do
    $block(i) \rightarrow P$;
    calculate $H(II) = H(block(i-1))_\mathrm{CP}$;
    if $H(II) \neq H(I)$ then
        ⟨CP-validate⟩: CP$(c)_{reject} \rightarrow C$
    end
    if $H(II) == H(I)$ then
        ⟨CP-validate⟩: CP$(c)_{accept} \rightarrow C$
    end
end
while $P \leftarrow \langle$CS$-consensus\rangle$ do
    obey "majority" rule;
    if $|C(c)_{accept}| > |C(c)_{reject}|$ then
        packages the next new $block(i+1)$
    end
    while $P \leftarrow block(i+1)$ do
        verify $H(block(i))$ in $block(i+1)$;
        if $|CP(c)_{accept}| > |CP(c)_{reject}|$ then
            publish $block(i)$ into blockchain
        end
    end
    out = success
end
return out
```

Thus the time complexity of BAC consensus could be calculated. Suppose there are c CP nodes and n CS nodes in a shard, where c is a fixed constant value and $c \ll n$. The rounds of communication are c and cn in the **Block-request** and CP-validate stage, respectively. The CS-consensus communication's rounds are $n(c+1)$. So the total rounds of communication are:

$$T = 2[c + cn + (c+1)n] = C_1 n + C_2 \tag{6}$$

where $C_1 = 4c + 2, C_2 = 2c$. So the time complexity of the BAC is $O(n)$.

B. BAC 2.0 - Improved BAC Using Hashgraph

This subsection discusses the optimized BAC 2.0 utilizing the Hashgraph based on the Direct Acyclic Graph (DAG) to improve the transaction rate and security of the BAC mechanism.

The P and CP conduct consensus on transaction data instead of blocks from EV requesters through Hashgraph to improve

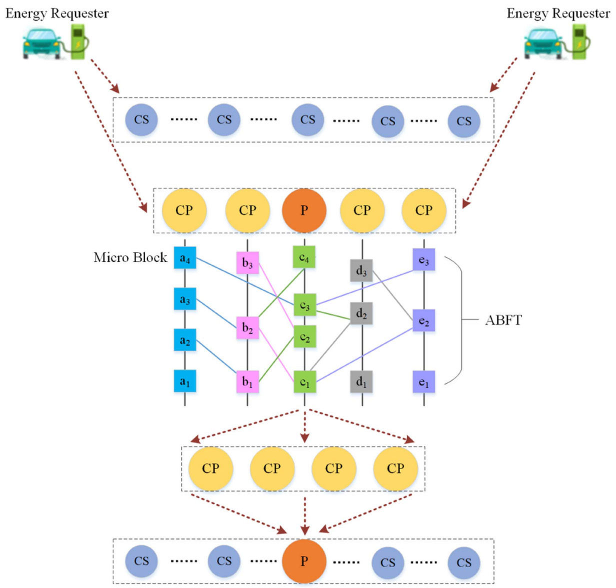


Fig. 5. BAC 2.0 consensus mechanism for V2V energy trading.

the throughput of ETB. The Hashgraph is completely decentralized because there is no primary node. However, complete decentralization is unrealistic given the current social form. The participation of primary nodes is always required. For example, regulators and grid departments are always necessary for V2V energy trading scenarios. Therefore, there is a primary packaging transactions into blocks after the Hashgraph stage. Furthermore, the primary could get more rewards than other nodes, which motivates distributed nodes to maintain the blockchain actively. Fig. 5 shows the BAC 2.0 for V2V energy trading.

Compare: In the basic BAC, there is only the concept of blocks. The block packaged by the P needs to be voted by all other EVs. The role of CP is to verify the block in advance and prevent the P from doing evil. The BAC confirms the validity of the first block in the second round of consensus (Bitcoin needs at least six rounds). In BAC 2.0, there are no block and primary in the DAG stage, so there is no need to consider the leader's evil. The Hashgraph is Asynchronous Byzantine Fault Tolerance (ABFT); nobody can prevent the network from reaching a consensus or modifying the data. The blockchain is still used in the subsequent stages. One reason is the social dimension that we discussed before. Another reason is that from a technical point of view, the Hashgraph is currently only suitable for private blockchains – its throughput in public blockchains is uncertain. And it is unclear whether Hashgraph's gossip algorithm is still ideal for large-scale networks such as V2V energy trading. Moreover, the total number of nodes N in Hashgraph needs to be preset, i.e., dynamic addition and deletion of nodes are not supported, which is the opposite of the characteristics of V2V energy trading. Therefore, only the P and CP participate in the Hashgraph consensus in BAC 2.0. In this way, P and CP constitute a private blockchain environment in their shard suitable for Hashgraph. However, the C nodes do not participate in the Hashgraph consensus, so they cannot obtain the global agreement of the transaction. Therefore, we still utilize the blockchain – the P broadcasts the block, and the distributed storage of the block is realized in CS. Moreover, the blockchain could support the dynamic addition and deletion of nodes, which is ideal for the high mobility of EVs.

C. Asynchronous Byzantine Fault Tolerance

Blockchain technology is developing rapidly and is currently the most dominant form of distributed ledger implementation. However, researchers gradually discover blockchain's limitations, such as scalability bottlenecks. The chained storage structure of the blockchain makes it impossible for nodes to generate blocks in parallel, which results in low throughput in the blockchain. And the confirmation of transactions in the blockchain is quite slow. The traditional blockchain takes at least six blocks (1 h) for a block to be officially published into the blockchain. These factors greatly limit the practical application of blockchain, such as V2V energy trading.

The performance bottleneck of blockchain is mainly caused by its consensus mechanism. The current consensus mechanism of blockchain is essentially a block packed by a leading node, and the other nodes of the whole network verify this block. To improve the efficiency of the blockchain, an asynchronous consensus method DAG has emerged to realize the concurrent writing of transactions (blocks), which has become the most promising technology to solve the blockchain scalability problem. The Hashgraph was proposed in 2016. As one of the typical applications based on the DAG, the Hashgraph realizes leaderless BFT consensus through virtual voting. In the Hashgraph, each member (node) maintains a "chain" of its own, and members interact with each other through the gossip protocol. The Hashgraph requires a fixed number of nodes to achieve BFT by the principle of greater than 2/3 of the total number of nodes.

Micro blocks: the fundamental element in BAC 2.0 mechanism. We define the block in the ABFT stage as a micro stage as a micro block that contains four components: a collection of EV energy transactions, timestamps, and a hash of references to two parent micro blocks. In the basic BAC, a new block has only one previous block. In BAC 2.0, each micro block needs to link two parent blocks, one of which is the previous micro block of itself, and the other is the micro block of any other node. Fig. 6 shows the difference between the structure of the block and the micro block.

Gossip about Gossip: The information to be synchronized in a Gossip protocol spreads like a rumor. The first EV to propagate a message m chooses a fixed propagation period (e.g., 1.0 s) and randomly propagates the m to the k nodes connected to it. If an EV receives the m and has not received m before, it will send the m to k neighboring EVs other than the one that sent the message last round. Eventually, all EVs in the blockchain will receive the message m. The gossip process in V2V ETB is that an EV randomly selects a neighboring EV and broadcasts a micro block to it. Gossip about Gossip means an EV node signs the gossip information it receives, packages the signature into a new message M, and randomly broadcasts the M to other nodes. Each gossip information includes the signature verification of its previous gossip information.

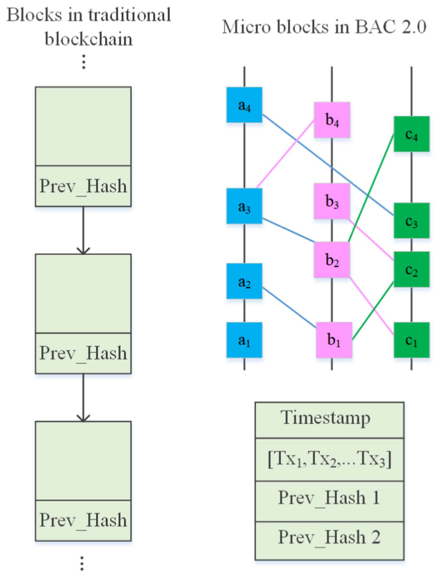


Fig. 6. Blocks in blockchain and microblocks in BAC 2.0.

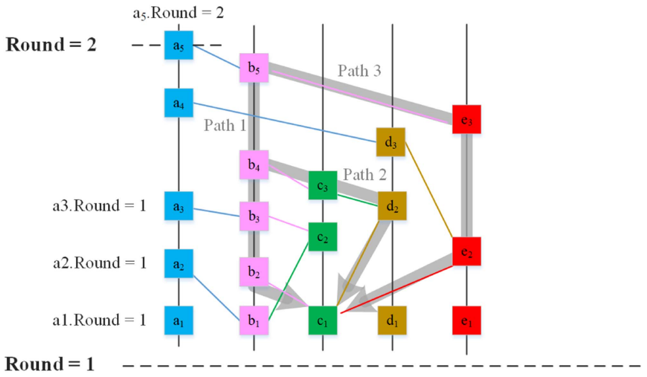


Fig. 7. Association between micro blocks.

Association between micro blocks: A child micro block y can see x if y can be traced back to some ancestor block x from a particular path. The y can strongly see x if all paths from y to x pass through most ($> 2/3$) EV nodes. As shown in Fig. 7, b_5 strongly sees c_1: The b_5 can see c_1 through 3 paths. Path 1 passes through EVs B and C, Path 2 passes through EVs B, D, and C, and Path 3 passes through EVs B, E, and C. Therefore, these three paths pass through four EVs B, C, D, and E in total, satisfying the condition of more than $2/3$ of the total number of EVs. In the initial state, all EVs are in the same round, denoted as $R = 1$. If a micro block x strongly sees the previous blocks of most EVs, the x is in a new round. The micro block a_5 strongly sees a_1, c_1, d_1, and e_1, so a_5 is in a new round $R = 2$.

Virtual Voting: A witness is the first micro block created in round R. If a witness of round R is strongly seen by a majority (more than $2/3$) of witnesses of round $R+1$, it becomes a famous witness. In traditional consensus mechanisms of blockchain, each EV collects votes on blocks from others, leading to low throughput and poor scalability. In BAC 2.0, EVs could calculate others' votes through virtual voting instead of broadcasting and collecting votes across the whole blockchain network. If a micro block y strongly sees the majority of witnesses, its vote for a witness x is valid. Then if the number of votes on the x exceeds $2/3$, the x can be marked as a famous witness; that is, the micro block x cannot be changed.

The BAC 2.0 needs two rounds of consensus like basic BAC. The first round is to vote on micro blocks, and the second round is to collect the voting results. For instance, y_1, y_2, y_3, and y_4 in round $R+1$ strongly see a witness x in round R, making x a famous witness. A witness z in round $R+2$ strongly see y_1, y_2, y_3, and y_4, then z can immediately confirm the x, i.e., the micro block x has reached the consensus of the whole network and cannot be changed.

D. Primary Election Based on Reputation Incentive and Cryptographic Sortition

This subsection delves into the process of selecting EVs to publish new blocks to the ETB system. In BAC 2.0, there is no puzzle-solving competition, and P are chosen based on EV's reputation value. EVs with higher reputation are more likely to be elected as P. Moreover, the P election process is random based on cryptography to prevent malicious nodes (hackers) from predicting (by calculating nodes' reputation value) the next round of P and thus attacking them.

We primarily analyse EVs' participation and success rate [29]. Let $\pi_l \in [0, 1]$ denote the EV's participation rate, and $\zeta_l \in [0, 1]$ represent the honest consensus EVs' successful validation rate. Let π denote the number of tasks an EV participates in, and ζ denote the number of successful tasks. Thus, the participation and success rate of EVs could be expressed as:

$$\pi_l = \frac{\pi}{K} \tag{7a}$$

$$\zeta_l = \frac{\zeta}{\pi} \tag{7b}$$

where K is the total block consensus tasks.

To determine an exact reputation value, we utilize a reputation management strategy based on subjective logic [16]. The evidence space and opinion space between EV sellers and buyers could be expressed as $\{\Phi_{i,j}, \eta_{i,j}, \varphi_{i,j}\}$ and $\{b_{i,j}, d_{i,j}, u_{i,j}\}$, respectively. The mapping of the opinion space to the evidence space could be expressed as:

$$b_{i,j} = \frac{\Phi_{i,j}}{\Phi_{i,j} + \eta_{i,j} + \varphi_{i,j}}, b_{i,j} \in [0, 1] \tag{8a}$$

$$d_{i,j} = \frac{\eta_{i,j}}{\Phi_{i,j} + \eta_{i,j} + \varphi_{i,j}}, d_{i,j} \in [0, 1] \tag{8b}$$

$$u_{i,j} = \frac{\varphi_{i,j}}{\Phi_{i,j} + \eta_{i,j} + \varphi_{i,j}}, u_{i,j} \in [0, 1] \tag{8c}$$

$$b_{i,j} + d_{i,j} + u_{i,j} = 1 \tag{9}$$

where $\Phi_{i,j}$, $\eta_{i,j}$, and $\varphi_{i,j}$ denote the number of honest behaviors, dishonest behaviors, and doubtful behaviors, respectively. Thus $b_{i,j}$, $d_{i,j}$, and $u_{i,j}$ denote the probabilities of "belief," "distrust," and "uncertainty," respectively. Finally, the reputation value of an EV could be calculated:

$$\rho_l = b_{i,j} + \varepsilon u_{i,j} \tag{10}$$

Algorithm 2: Primary Election.

```
Input: $sk$, $seed$, $\tau$, $role$, $\rho_i$, $R$
Output: $\langle hash, \mu, \lambda \rangle$
$\langle hash, \mu \rangle \leftarrow VRF_{sk}(seed \| role)$;
$p \leftarrow \frac{\tau}{R}$;
$\lambda \leftarrow 0$
while $\frac{hash}{2^{hashlen}} \notin \left[\sum_{k=0}^{\lambda} B(k;\rho_i,p), \sum_{k=0}^{\lambda+1} B(k;\rho_i,p)\right)$ do
    $\lambda$ ++;
    out $\leftarrow \langle hash, \mu, \lambda \rangle$
end
return out
```

where ε is a preset constant that expresses the degree to which unknown behaviours impact the trust value.

Cryptographic sortition is used for randomly choosing P according to EVs' reputation. Let ρ_i denote the EV i's reputation, and $R = \sum_i \rho_i$ denote the reputation of all EVs. The probability that EV i is selected as a P or CP is proportional to ρ_i/R. The randomness in primary election comes from a publicly known random $seed$. To prove an EV is selected, each EV i has a public/private key pair, (pk_i, sk_i).

The VRF [27] is utilized in primary election. Informally, let x be any input string, then $VRF_{sk}(x)$ returns two results: a hash and a proof. The hash is uniquely determined by sk and x. The proof μ allows EVs besides EV i to validate that the hash corresponds to x, without knowing the private sk.

The primary election is shown in Algorithm (2). The election requires a $role$ parameter that distinguishes different roles that an EV may be selected for. For example, an EV user may be selected as a primary to package and publish blocks in some round, or as a candidate to supervise and validate. BAC 2.0 specifies a threshold τ that determines the expected number of EVs elected for P or CP.

An EV user may be elected more than once by the election algorithm because it has a high reputation value. The election realizes this by returning the λ parameter, which denotes how many times an EV is elected. If an EV i owns ρ_i (integral) units of its reputation, then (i, λ) with $\lambda \in \{1, 2, \ldots, \rho_i\}$ represents the λ^{th} unit of reputation EV i owns, and i is elected with probability $p = \frac{\tau}{R}$, where R is the total amount of reputation units in BAC.

An EV_i performs election by computing $\langle hash, \mu \rangle \leftarrow VRF_{sk}(seed \| role)$, where sk is the i's private key that only i knows. The probability that exactly k out of the ρ_i (the i's reputation) units are elected follows the binomial distribution:

$$B(k; \rho_i, p) = C_{\rho_i}^{k} p^k (1-p)^{\rho_i - k} \tag{11}$$

where $\sum_{k=0}^{\rho_i} B(k; \rho_i, p) = 1$. To determine how many of an EV's ρ_i units are elected, the election algorithm of BAC divides the interval [0,1) into consecutive intervals and it could be expressed as:

$$I^{\lambda} = \left[\sum_{k=0}^{\lambda} B(k; \rho_i, p), \sum_{k=0}^{\lambda+1} B(k; \rho_i, p)\right) \tag{12}$$

where $\lambda \in \{0, 1, \ldots, \rho_i\}$. If $hash/2^{hashlen}$ ($hashlen$ is the bit -length of hash) falls in the interval I^{λ}, then the EV i has exactly λ selected units, i.e., the EV i is elected λ times.

Algorithm 3: Primary Election.

```
Input: $pk$, $seed$, $hash$, $\tau$, $\mu$, $role$, $\rho_i$, $R$
Output: $out$
$p \leftarrow \frac{\tau}{R}$;
$\lambda \leftarrow 0$
while an EV receives i's election do
    if $VRF_{pk}(\mu, hash, seed \| role)$ fail then
        return 0;
    end
end
while $\frac{hash}{2^{hashlen}} \notin \left[\sum_{k=0}^{\lambda} B(k;\rho_i,p), \sum_{k=0}^{\lambda+1} B(k;\rho_i,p)\right)$ do
    $\lambda$ ++;
    out $\leftarrow$ success
end
return out
```

The election mechanism provides two essential properties. First, EVs are elected at random based on their reputation. Second, a cyber attacker who does not know sk_i cannot guess how many times an EV i is chosen or if i is chosen at all.

The procedure for verifying an election proof is shown in Algorithm 3. If the hash and the proof are mismatched, no further validation is required. The verification function returns the number of elected units of an EV (or zero if the EV is not elected at all). Moreover, the election mechanism of BAC could defend against Sybil Attacks. An attacker deploys only one entity but broadcasts multiple identities (IDs) to the blockchain network to act as several different nodes. These forged identities are generally referred to as Sybil nodes. Splitting an EV's reputation among Sybils does not affect the number of elected units under the EV user's control. We move the detailed pricing mechanism based on Bayesian Game to the Appendix H because of the page limit.

V. Performance Evaluation

A. Simulation Environment

We write a Python program and combine the VIBES blockchain simulator to demonstrate the proposed BAC consensus mechanism working for the ETB. We compare our proposed BAC consensus mechanism with the PoW, PBFT, and Hashgraph. All computations are done on a Lenovo computer with Windows Ultimate 64- bit, Intel i7-8550 CPU @ 1.80 GHz and 8.0 GB 2133 MHz LPDDR3, Java JDK Version 11.0.10, Scala Version 2.13.5, and Akka Version 2.6.14. Our simulation considers a V2V blockchain network with four separate shards linked to a single MBS through RSU over wireless connectivity. Our simulation supports large-scale networks with thousands of EVs (nodes), and we simulate 20 to 160 nodes considering the performance of our device. Each shard has a diameter ranging from 0 to 3 km. Each EV travels at 45 to 60 miles per hour

TABLE II
COMPARISON WITH DIFFERENT LEDGER TECHNOLOGIES FOR V2V ENERGY TRADING NETWORK

Features	Immutability	DoS Resistance	Fair Ordering	Fair Timestamps	Dynamicity
Central Server	×	×	×	×	✓
Leader Based	✓	×	×	×	\
Traditional Blockchain	✓	✓	×	×	✓
Hashgraph	✓	✓	✓	✓	×
Our model	✓	✓	✓	✓	✓

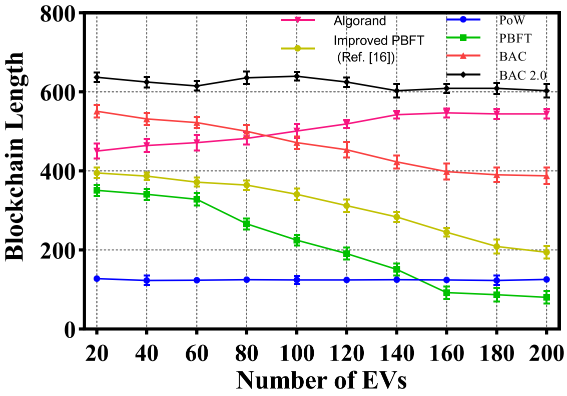


Fig. 8. The comparison of blockchain length.

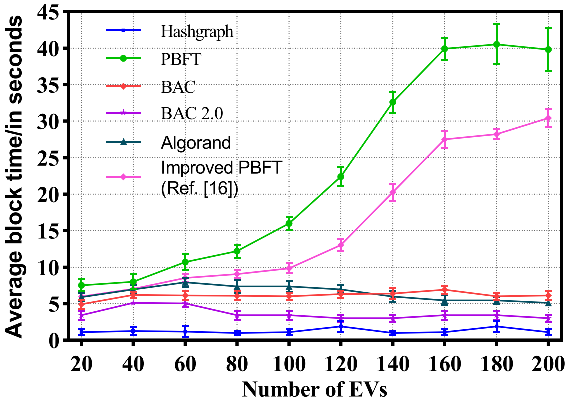


Fig. 9. Average time for a block published, with 20 to 200 EVs.

and charges at 22KWh. The RSUs transmit at a 300 metres radius [30].

Between EV_i and EV_j, the Euclidean distance varies between 5 and 100 metres and $l_{i,j}^{min}$ is set to 5 dB. We implement our V2V ETB system in the Hyperledger Fabric and test the performance of the ETB utilizing the Hyperledger Caliper. The energy block size is 2.0 MB, with a propagation latency of 0.8 s. The energy micro block size is 0.5 MB, with a propagation latency of 0.5 s. We move the list of key abbreviations and notations to the Appendix I because of the page limit.

B. Performance Analysis

Table II compares the BAC model to other distributed ledger technologies. Our model accomplishes Immutability, Denial of Service resistance, Fair ordering, Fair timestamp, and Dynamicity of nodes. The experiment of each method has been replicated ten times under the condition of the same number of EVs. Then we calculate the average of these 10 data points of each method and compare them. We compare and analyze the "Blockchain Length" of PoW, PBFT, BAC, and BAC 2.0. We consider the Hashgraph's "Average Block Time" instead of "Blockchain Length" since the micro block's size is not the same as the block in the blockchain. Figs. 8 and 9 demonstrate the performance comparison of these consensus mechanisms.

1) Blockchain Length in Energy Trading Blockchain: The average time interval between block generation in traditional blockchain (PoW) is constant, so the traditional blockchain's length is almost unchanged for a given period. The blockchain length of PBFT is much higher than PoW with a small quantity of EVs since the PBFT does not rely on computing power. However, the communication overhead of PBFT is too high since its time complexity is $O(N^2)$, where N is the number of EVs. As the number of EVs increases, the block's propagation delay extends, and the blockchain length drops sharply with more than 80 EVs. The BAC does not rely on computing power, and its time complexity is $O(N)$, so its communication overhead is significantly reduced compared with PBFT. The BAC 2.0's blockchain length is the highest among the four since the ABFT stage dramatically improves the throughput of BAC (it achieves a speed of 50,000 transactions per second compared to 15 for Bitcoin, and 30 for Ethereum). In the subsequent blockchain stage, the block packaged by the P will be directly distributed to the CS after authentication by CP to realize distributed storage.

2) Average Block Time in Energy Trading Blockchain: The average block time is the average time it takes for a new block to be added to the blockchain. The blockchain length is the number of blocks generated during a time interval $T(s)$, and the average block time $t(s)$ is the time interval $T(s)$ divided by the blockchain length. The average block time of PoW remains constant because it automatically adjusts the puzzle difficulty. In PBFT, the block time depends on EVs' behavior and processing efficiency. As the number of EVs increases, the average block time gradually grows. The block time increases sharply with more than 80 EVs. Though a block in BAC has to be confirmed by two rounds of consensus, the average block time is much less than PBFT because of its $O(N)$ time complexity. The advantage is evident with more than 100 EVs. The Hashgraph's block time is the shortest. In the above algorithms, the second block cannot be published until the first block is verified as valid, while in Hashgraph, EVs publish blocks in parallel without verification of block by block. Only specified EVs in Algorand actually participate in blockchain consensus (we set it to 10% of the total number of EVs, as well as BAC 2.0), so if the number of EVs is minimal, its advantages are not obvious. Once the number of nodes exceeds 100, the advantages of Algorand become apparent. However, Algorand's consensus method is too

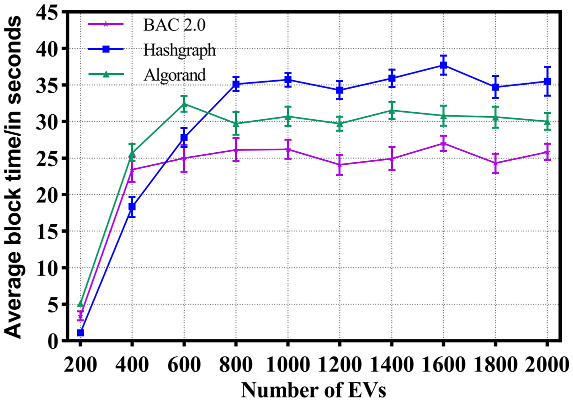


Fig. 10. Average time for a block published, with 200 to 2000 EVs.

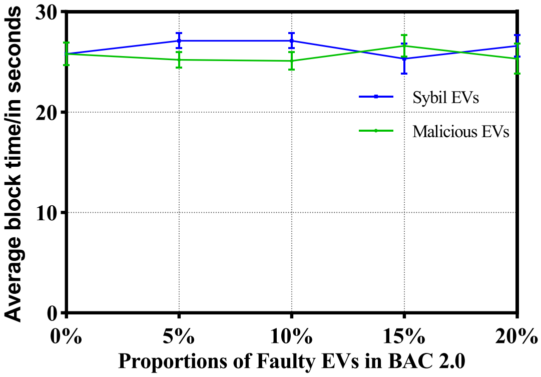


Fig. 11. The performance of the ETB with different proportions of dishonest EVs and malicious EVs executing Sybil Attack.

complicated to be applied to industrial applications, particularly for blockchains with significant communication complexity and latency restrictions. In contrast to the two rounds of communication (prepare, commit) in PBFT, the consensus process in Algorand takes at least five rounds of communication to reach consensus. In contrast, BAC 2.0 uses a "majority vote" approach to minimize complex communication among EVs while yet accomplishing the same result.

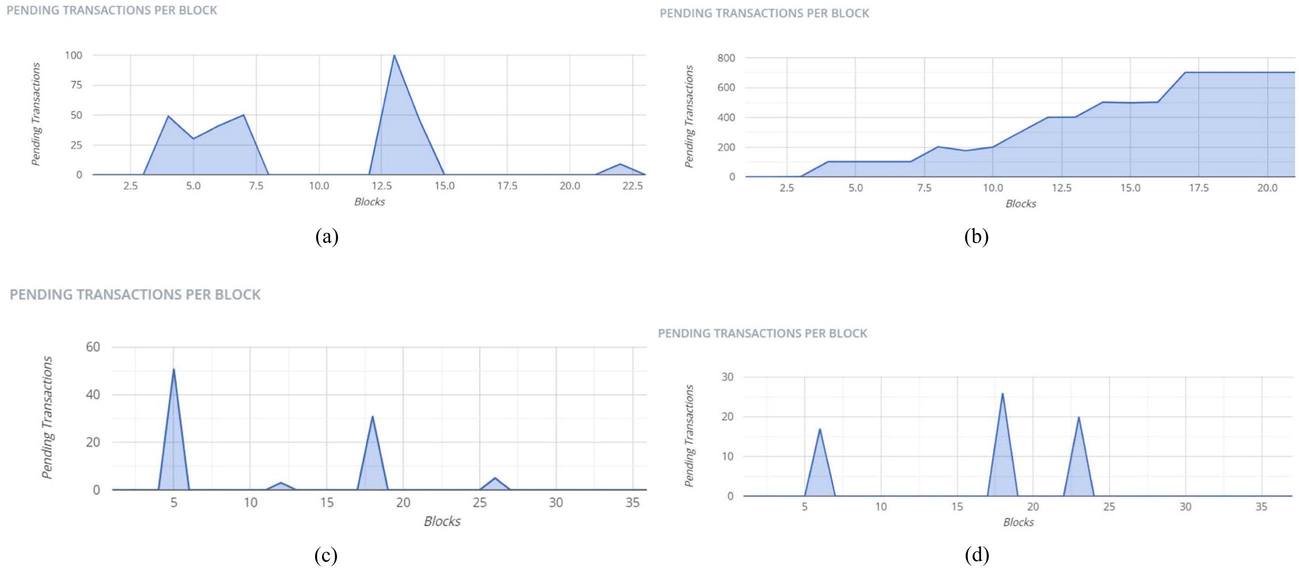
3) Scalability: Fig. 10 demonstrates the scalability of BAC 2.0, Hashgraph, and Algorand, scaling the number of EVs from 200 to 2,000. The latency of BAC 2.0 is about 5 times higher than that shown in Fig. 9. However, the scaling performance remains relatively flat all the way to 2,000 users, indicating that BAC 2.0 scales effectively. The latency of Hashgraph is about 18 times higher than Fig. 9. When the number of EVs exceeds 600, BAC outperforms Hashgraph since the majority of EVs implement distributed storage. Each EV in Hashgraph must perform the gossip and "gossip about the gossip" protocol, resulting in two bottlenecks: CPU time and bandwidth. BAC 2.0 consistently outperforms Algorand because of its sharding technique, and BAC 2.0, like Hashgraph, has extraordinarily high throughput and transaction speed. We move the detailed analysis of BAC's scalability to the Appendix J.

4) Pending Transactions and Reputation Incentive in Energy Trading Blockchain: Pending transactions are those waiting for EVs to confirm the transfer information and package them into the block. If the balance of an EV requester j is insufficient, then the transaction request sent by j is pending until he/she has a sufficient balance. If the network is congested during peak trading hours, trading requests that have not yet been packaged during this period are also called pending transactions. Fig. 12 shows these consensus mechanisms' pending transactions in our V2V ETB system. Every block in the blockchain has its transaction pool size minus the number of transactions already included in this block at the block creation. The vertical and horizontal coordinates represent the pending transactions per block and the number of blocks, respectively. So the shaded region represents the total pending transactions in the blocks. The PBFT gets the maximum number of pending transactions due to its high communication complexity and poor scalability. Then we calculate the number of pending transactions of PoW as 295, BAC as 107.5, and BAC 2.0 as 75. EVs involved in PBFT consensus cannot efficiently conduct an excessive number of transaction requests. The system will stop generating blocks to maintain consistency in the case of view change due to the leader's crash or misbehavior. PBFT is highly centralized and poorly scalable: the number of nodes cannot be excessive to ensure frequent and complex communication between EVs. The threshold for nodes to participate in PBFT is high since PBFT is not secured against Sybil Attack. EVs must be verified before entering since they cannot defend against the falsification by a malicious user generating multiple identities. PoW consensus results in a waste of resources. Finding the proper hash does nothing more than the numerous hash operations required for mining. PoW's network performance is inadequate. It takes at least 10 minutes to confirm a transaction in Bitcoin, only 7 transactions can be processed on average per second, and there is no guarantee that the leading will always pack all transactions.

We illustrate four sub-graphs of Fig. 12 since the performance gap between the consensus mechanisms in Fig. 12 is rather evident. To more clearly show the advantages of BAC 2.0 over Hashgraph and Algorand, we exhibit the pending transactions of the three algorithms over 2,000 nodes and 90 blocks in one graph, as shown in Fig. 13. And we compare the pending transactions of improved PBFT adopted by Ref. [16] and PBFT in Fig. 14. We move the detailed understanding drawn from the pattern of the curves to the Appendix K because of the page limit.

There are two types of incentives in the traditional blockchain. One is the block reward through mining, and the other is the transaction fee. A requester who broadcasts a transaction has to pay a certain amount of bitcoin to the miner as a reward. The more rewards he/she sets, the greater chance that the transaction would be packaged into the block by the miner. There are also two kinds of incentives in BAC. One is the reputation reward given by the ETB system based on the effectiveness of the block's final state. The other is the reward of CP or P broadcasting the required blocks to other EVs. As shown in Fig. 15, an EV uses its credits to attach a fee to a transaction and thus persuades the P and CP to include the transaction in the following block to be published.

5) Security Evaluation: Fig. 11 demonstrates the performance of ETB with different proportions of dishonest EVs and



(a) (b) (c) (d)

Fig. 12. Pending transactions. (a) PoW. (b) PBFT. (c) BAC. (d) BAC 2.0.

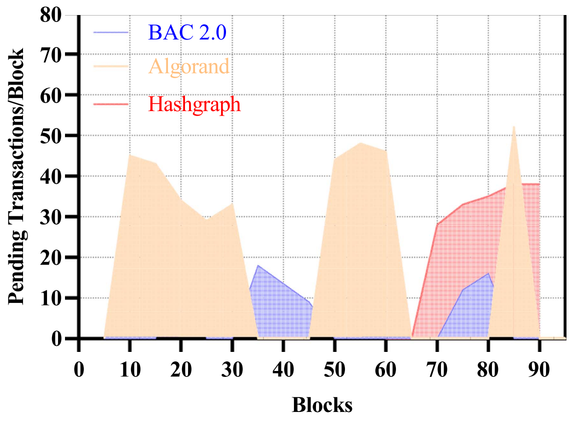


Fig. 13. The pending transactions of BAC 2.0, algorand, and hashgraph.

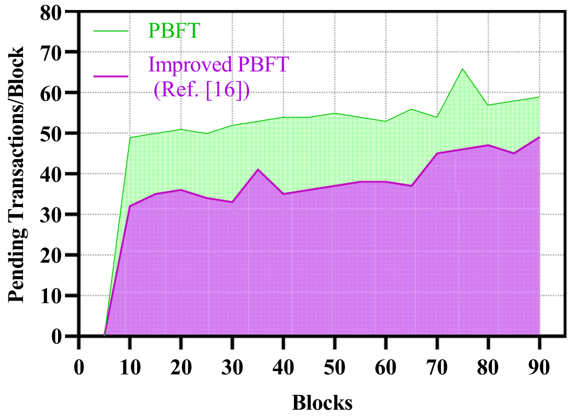
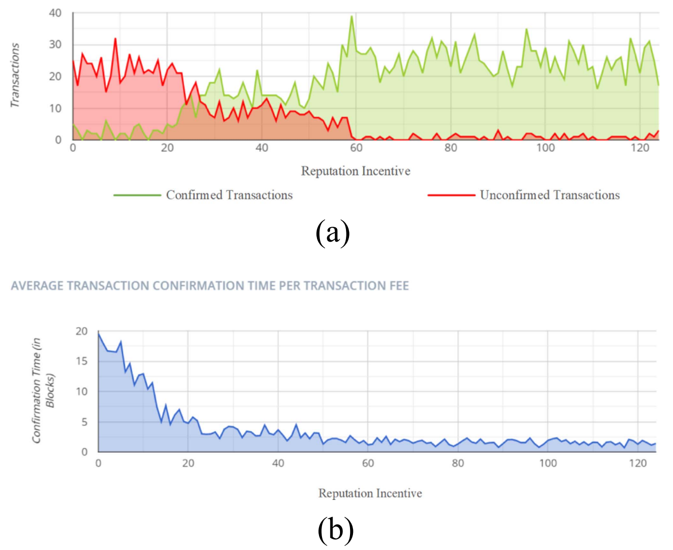


Fig. 14. The pending transactions of PBFT and improved PBFT (Ref. [16]).



(a)

(b)

Fig. 15. Pending transactions. (a) Confirmed and unconfirmed transactions with the increase of reputation score. (b) Confirmation time with the increase of credit score.

malicious EVs executing Sybil Attack, respectively. There are 2000 EVs participating the ETB. The ABFT stage of ETB makes it possible for no EVs in ETB to prevent the network from reaching consensus or tampering with data after consensus is reached. The ETB provides strong security, which can theoretically match Bitcoin's level, in addition to excellent performance, such as high throughput. The ETB eliminates the assumption that many consensus algorithms have a maximum tolerance for message latency, allowing certain messages to be lost or delayed indefinitely before being eventually validated as coming from honest EVs. The ETB could tolerate network communication failures and resist arbitrary loop breaking by malicious EVs, thus guaranteeing that V2V energy trading could operate well in the IoEV. The results show that our ETB system is not significantly affected by the dishonest behavior of EVs.

Utilizing the distributed nature of V2V energy trading, the Sybil node broadcasts its disguised nodes to the entire IoEV network to take over the network, deny responses, obstruct requests, etc. In the BFT scenario, the number of disguised nodes can control the whole network as long as the number of disguised nodes exceeds the N/3 limit. While there may only be one malicious EV in the actual IoEV. Our ETB system defends against Sybil Attack by two methods. The first method is to authenticate the EVs joining the IoEV through CA in Hyperledger Fabric. Since malicious EVs execute Sybil Attack by fabricating network IDs, the most direct method is to authenticate each EV that joins the ETB. In this way, the fake EVs cannot pass the authentication, thus solving the witch assault. The ETB is certified for EVs through a CA authority in Fabric, which is equivalent to a third-party trusted organization. The second method is that our ETB makes it more difficult to forge identities through reputation incentives. The ETB elects committees and the leader based on EV users' reputation scores, and assigns EVs weight through the reputation incentive, which prevents malicious EVs from fabricating multiple identities to increase their probability of being elected. The ETB will conduct committee re-elections on a regular basis so that power can be randomly distributed to all network EVs, making it considerably less likely that malicious nodes can perpetrate evil and acquire control of the V2V energy trading network.

C. Hyperledger Fabric

We analyze the superiority of BAC through simulation in subsection B. In this subsection, we have deployed the V2V ETB platform on Hyperledger Fabric and tested the performance of the ETB utilizing the Hyperledger Caliper.

Hyperledger is the first open-source project for the consortium blockchain. It employs a modular and universal structure with unique identity management and access control features that make it well suited for a wide range of industrial applications, including energy trading. Hyperledger is a distributed ledger platform designed to support enterprise-level applications and support pluggability and scalability. Transactions in the Hyperledger Fabric are executed on a channel (private blockchain), and each party must be authenticated and authorized to process transactions on that channel. The consensus mechanism in Hyperledger is pluggable, so we add BAC to the consensus module of Hyperledger. The V2V ETB is implemented in a desktop running 64-bit Ubuntu 16.04.6 LTS with 1.6-GHz Intel Core i5 Quad-CPU and 6 G RAM.

We utilize Hyperledger Caliper for our performance evaluation. Hyperledger Caliper is a blockchain performance evaluation tool that enables users to evaluate various blockchain schemes through preset application cases and retrieve performance test results such as success rate, throughput, and latency. We have programmed our adaptors to connect with the ETB using Fabric Client SDK (NodeJS version) to combine with our current Hyperledger Fabric profile management system. A benchmark layer sits on top of the adaption layer, implementing specified use-cases in the form of YAML configuration files.

We move the detailed analysis of the read and write performance and success rate to the Appendix L because of the page limit. Fig. L.1 illustrates our V2V ETB performance under a different number of workloads from 500tps to 2000tps. There are 2000 EVs generating proposals for our ETB system.

The ordering service is the most critical part of the consensus mechanism in Hyperledger. To reach a consensus, all transactions have to be ordered through the ordering service. Once the transaction is written to a block, its location in the ledger can be ensured. The primary in BAC also needs to sort transaction proposals. However, the ordering service has also become the performance bottleneck of blockchain networks. Since WRITE transactions take additional processing to sequence transactions chronologically, build a new block, and broadcast it to all EVs in the ETB, they have a lower throughput, a lower success rate, and a greater latency.

D. Security Analysis on V2V ETB

The V2V ETB defends against various cyber attacks to ensure energy trading security.

1) Defend Against DoS Attack: The BAC uses VRF for cryptographic lotteries in selecting the leader EV and committees to randomly elect the primary based on its reputation value without interaction. EV users are the only ones aware of their status as the leader and committee members. Malicious nodes do not know EVs' identities and therefore cannot bride honest EVs or launch DoS attacks against them. If an EV is selected, a string is generated to prove that it is the committee member or leader, and it includes this string in the message it sends. The malicious node is unaware of which EV user has been chosen until the user begins to participate in the BAC consensus. Each validator utilizes $\langle seed_r, \mu\rangle = VRF_{skv}(seed_{r-1}||r)$ to calculate the seed for round r, where skv is the private key of the validator, $seed_{r-1}$ is the random seed of the previous round. If the attacker takes complete control of the messaging link, the proposed blocks are removed, and the blank blocks are forced to be approved by the user, thus generating a random seed that will be used for subsequent elections. This ensures that the proposer and random seed are not compromised in advance, and that BAC is adaptively secure against DoS attacks against the leader EV, even when the EV is offline or even in instantaneous corruption mode.

2) Defend Against Sybil Attack: Two strategies are used by our ETB system to defend against Sybil Attack. The first technique involves using the CA in Hyperledger Fabric to authenticate the EVs joining the IoEV. The most straightforward approach is to authenticate each EV that joins the ETB as malicious EVs carry out Sybil Attack by creating network IDs. As a result, the Sybil Attack is prevented because the bogus EVs are unable to pass authentication. A CA authority in Fabric, comparable to a third-party trusted organization, certifies the ETB for EVs. Our ETB's reputation incentives make it harder for malicious EVs to create false identities, which is the second strategy. Assigning weight to EVs through the reputation incentive, the ETB elects committees and the leader based on the reputation scores of EV users. This prohibits malicious EVs from creating several identities to boost their chances of getting chosen. Regular committee re-elections will be held by the ETB to ensure that power is allocated randomly to all network EVs, significantly reducing the possibility of malevolent nodes carrying out nefarious deeds and seizing control of the V2V energy trading network.

3) Get Rid of a Centralized Intermediary: In the V2V ETB, distributed energy trading is conducted in a P2P manner without any trusted third-party institutions. The V2V energy trading incorporating blockchain technology utilizes distributed data storage, and all transactions are stored in the form of blocks at each node, thus enhancing the security and stability of V2V ETB. Even if a single node is compromised by a malicious node, it will not lead to the paralysis of the entire energy trading system.

4) Data Unforgeability and Immutability: Each block in the V2V ETB contains the data fingerprint (hash value) of all the data in the previous block, and the current block's hash value is calculated while the previous block's hash value is taken into account. In this way, the ETB creates a linking relationship between the blocks. Therefore, once the data in a block is changed, the hash values of all subsequent blocks will be changed. All nodes can detect data tampering and discard such invalid data. This ensures that the blockchain data is tamper-proof. Our V2V ETB is deployed in the Fabric consortium blockchain environment, and any node must be authenticated in the ETB before joining the system. The feature of the ETB consortium blockchain combined with digital signature guarantees that no adversary can pose as V2V ETB nodes to corrupt the blockchain network.

E. Theoretical Analysis of BAC and V2V ETB

1) Storage Cost: The storage space required for the blockchain ledger is increasing over time. The Bitcoin system, for example, has a storage cost of more than 50 GB per year, and after more than 10 years of operation, its storage overhead has become extremely large. This problem becomes more prominent in the resource-limited V2V ETB, and the increasing storage overhead has become a hindrance for nodes in the IoEV to join data sharing. In V2V ETB, only the committee EV nodes need to store the full information of the blockchain, whereas regular EV users are like light nodes in Bitcoin and simply keep the block header and the portion of the transaction related to themselves. Moreover, the V2V ETB system deployed in Hyperledger Fabric stores block proofs to prove to new EV users who join the shard that a block is already in the blockchain. The size of this block proof is 200 KBytes, for a 1 MByte block, it is about a 19% storage cost. The V2V energy trading is an Internet of Things (IoT) application, and the status of the EVs in the IoEV varies constantly. EV users are not always guaranteed to be online as they might join or exit the blockchain network anytime. Nodes in Bitcoin mine according to their preferences, while nodes in V2V ETB participate in committee elections according to their preferences. EV users that desire rewards will spend more time packing or verifying blocks, whereas regular users are like light nodes in Bitcoin and simply need to keep the portion of the transaction connected to themselves.

2) Communication Complexity: There are c CP nodes and n CS nodes in the BAC, c is a fixed constant value and $c \ll n$. The rounds of communication are c in the Block-request stage. The rounds of communication are cn in the CP-validate stage. The rounds of communication are $(c+1)n$ in the CS-consensus stage. So the total rounds of communication are $T2 = 2[c + cn + (c+1)n] = C_1 n + C_2$,where $C_1 = 4c + 2, C_2 = 2c$.

BAC 2.0 implements blockchain sharding technique and Hashgraph and utilizes a "witness-as-vote" strategy to sorting transaction history. When an EV user i observes a new event (such as a new transaction T), it packs T into a micro block $Block(i)$; in addition to the event T known to EV_i itself, the block constructed by EV_i needs to reference two earlier blocks, one of which is its previous block $Block(i-1)$, which EV_i itself generated, and the other is the most recent block $Block(j)$, which EV_i received from the other EVs. EV_i then adds a timestamp to $Block(i)$ and its signature to randomly propagate the $Block(i)$ to another EV user EV_j. In the future, EV_j could continue to spread the "message EV_i told me about a transaction T that happened at a certain time" to others by referring to the $Block(i)$ as a witness. Thus, the transaction T initiated by EV_i can spread rapidly among the EVs in the form of gossip, taking only about $log(c)$ times to reach all c participants. Each EV in BAC 2.0 has a chain, and each microblock in the chain refers a microblock in another chain. This is a unique structural element that sets BAC 2.0 apart from a typical DAG. BAC 2.0 actually defines the path that "events" take to propagate through the gossip network. EVs can check the locally stored graph to determine not only whether the majority of EVs have seen a micro block, but also the sequence in which each EV has seen various micro blocks. The main benefit of Hashgraph over the conventional BFT algorithm is that it only requires c EVs to transmit $clog(c)$ messages to complete a round of voting. Thus the message complexity of the ABFT phase of BAC 2.0 is $O(clog(c))$. The total message complexity of BAC 2.0 is then $T_3 = O(n + clog(c)) = O(n)$. To finish a round of voting using the conventional BFT method, each EV must send $N-1$ point-to-point messages.

VI. Conclusion and the Road Ahead

In this paper, we propose the V2V energy trading based on blockchain technology. The ETB performs by removing centralized third-party platforms. Therefore, the consensus mechanism between distributed EVs is crucial. We propose the BAC consensus algorithm instead of directly adopting the traditional consensus mechanisms such as PoW and PBFT to improve the ETB's performance and allow the system to continue to work correctly when software errors or Byzantine EVs are present. Moreover, our proposed BAC mechanism solves the problems that Hashgraph is completely decentralized, cannot resist network attacks, and does not support dynamic node addition and deletion. The BAC of ETB randomly elects the primary and the committee through the cryptographic approach based on EVs' reputation incentive, making it impossible for attackers to disrupt the V2V network by creating numerous fake identities. There is still much work to improve the V2V energy trading blockchain system and the BAC consensus mechanism. For future work, we will utilize machine learning to shard EVs in the V2V ETB network and design an incentive mechanism based on a game theory.

ACKNOWLEDGMENT

Our deepest gratitude goes to the anonymous reviewers for their careful work and thoughtful suggestions that have helped improve this paper substantially.

REFERENCES

[1] Y. Yao, X. Chen, L. Rao, X. Liu, and X. Zhou, "LORA: Loss differentiation rate adaptation scheme for vehicle-to-vehicle safety communications," *IEEE Trans. Veh. Technol.*, vol. 66, no. 3, pp. 2499–2512, Mar. 2017.

[2] H. Peng et al., "Resource allocation for cellular-based inter-vehicle communications in autonomous multiplatoons," *IEEE Trans. Veh. Technol.*, vol. 66, no. 12, pp. 11249–11263, Dec. 2017.

[3] F. R. Badal, P. Das, S. K. Sarker, and S. K. Das, "A survey on control issues in renewable energy integration and microgrid," *Protection Control Modern Power Syst.*, vol. 4, no. 1, pp. 1–27, 2019.

[4] S. Xia, F. Lin, Z. Chen, C. Tang, Y. Ma, and X. Yu, "A Bayesian game based vehicle-to-vehicle electricity trading scheme for blockchain-enabled internet of vehicles," *IEEE Trans. Veh. Technol.*, vol. 69, no. 7, pp. 6856–6868, Jul. 2020.

[5] M. H. Rehmani, M. Reisslein, A. Rachedi, M. Erol-Kantarci, and M. Radenkovic, "Integrating renewable energy resources into the smart grid: Recent developments in information and communication technologies," *IEEE Trans. Ind. Informat.*, vol. 14, no. 7, pp. 2814–2825, Jul. 2018.

[6] H. Peng, L. Liang, X. Shen, and G. Y. Li, "Vehicular communications: A network layer perspective," *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 1064–1078, Feb. 2019.

[7] F. Tang, Z. M. Fadlullah, N. Kato, F. Ono, and R. Miura, "AC-POCA: Anticoordination game based partially overlapping channels assignment in combined UAV and D2D-based networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 2, pp. 1672–1683, Feb. 2018.

[8] M. Pazos-Revilla, A. Alsharif, S. Gunukula, T. N. Guo, M. Mahmoud, and X. Shen, "Secure and privacy-preserving physical-layer-assisted scheme for EV dynamic charging system," *IEEE Trans. Veh. Technol.*, vol. 67, no. 4, pp. 3304–3318, Apr. 2018.

[9] S. N. Islam, Z. Baig, and S. Zeadally, "Physical layer security for the smart grid: Vulnerabilities, threats, and countermeasures," *IEEE Trans. Ind. Informat.*, vol. 15, no. 12, pp. 6522–6530, Dec. 2019.

[10] G. Sun, M. Dai, F. Zhang, H. Yu, X. Du, and M. Guizani, "Blockchain-enhanced high-confidence energy sharing in internet of electric vehicles," *IEEE Internet Things J.*, vol. 7, no. 9, pp. 7868–7882, Sep. 2020.

[11] S. Xie, Z. Zheng, W. Chen, J. Wu, H.-N. Dai, and M. Imran, "Blockchain for cloud exchange: A survey," *Comput. Elect. Eng.*, vol. 81, 2020, Art. no. 106526.

[12] M. Salimitari, M. Chatterjee, M. Yuksel, and E. Pasiliao, "Profit maximization for bitcoin pool mining: A prospect theoretic approach," in *Proc. IEEE 3rd Int. Conf. Collaboration Internet Comput.*, 2017, pp. 267–274.

[13] J. Kang, Z. Xiong, D. Niyato, D. Ye, D. I. Kim, and J. Zhao, "Toward secure blockchain-enabled internet of vehicles: Optimizing consensus management using reputation and contract theory," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 2906–2920, Mar. 2019.

[14] H. Huang, W. Kong, S. Zhou, Z. Zheng, and S. Guo, "A survey of state-of-the-art on blockchains: Theories, modelings, and tools," *ACM Comput. Surv.*, vol. 54, no. 2, pp. 1–42, 2021.

[15] Z. Zheng, J. Pan, and L. Cai, "Lightweight blockchain consensus protocols for vehicular social networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 6, pp. 5736–5748, Jun. 2020.

[16] H. N. Abishu, A. M. Seid, Y. H. Yacob, T. Ayall, G. Sun, and G. Liu, "Consensus mechanism for blockchain-enabled vehicle-to-vehicle energy trading in the internet of electric vehicles," *IEEE Trans. Veh. Technol.*, vol. 71, no. 1, pp. 946–960, Jan. 2022.

[17] L. Lamport, R. Shostak, and M. Pease, "The Byzantine generals problem," in *Concurrency: The Works of Leslie Lamport*. New York, NY, USA: Assoc. Comput. Mach., 2019, pp. 203–226.

[18] M. Castro et al., "Practical Byzantine fault tolerance," *OsDI*, vol. 99, no. 1999, pp. 173–186, 1999.

[19] R. Kotla, L. Alvisi, M. Dahlin, A. Clement, and E. Wong, "Zyzzyva: Speculative byzantine fault tolerance," in *Proc. 21st ACM SIGOPS Symp. Operating Syst. Princ.*, 2007, pp. 45–58.

[20] P. Daian, R. Pass, and E. Shi, "Snow white: Robustly reconfigurable consensus and applications to provably secure proof of stake," in *Proc. Int. Conf. Financial Cryptogr. Data Secur.*, 2019, pp. 23–41.

[21] P. Ekparinya, V. Gramoli, and G. Jourjon, "The attack of the clones against proof-of-authority," 2019, *arXiv:1902.10244*.

[22] Z. Su, Y. Wang, Q. Xu, M. Fei, Y.-C. Tian, and N. Zhang, "A secure charging scheme for electric vehicles with smart communities in energy blockchain," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4601–4613, Jun. 2018.

[23] L. Feng, H. Zhang, Y. Chen, and L. Lou, "Scalable dynamic multi-agent practical byzantine fault-tolerant consensus in permissioned blockchain," *Appl. Sci.*, vol. 8, no. 10, 2018, Art. no. 1919.

[24] Y. Yang, W. Yang, Z. Guo, and J. Zhu, "Research on consensus algorithm of multi energy interaction agents based on PBFT," in *Proc. IEEE Int. Conf. Inf. Technol. Big Data Artif. Intell.*, 2020, pp. 416–421.

[25] D. Wang, Z. Tao, J. Zhang, and A. A. Abouzeid, "RPL based routing for advanced metering infrastructure in smart grid," in *Proc. IEEE Int. Conf. Commun. Workshops*, 2010, pp. 1–6.

[26] N. Z. Aitzhan and D. Svetinovic, "Security and privacy in decentralized energy trading through multi-signatures, blockchain and anonymous messaging streams," *IEEE Trans. Dependable Secure Comput.*, vol. 15, no. 5, pp. 840–852, Sep./Oct. 2018.

[27] S. Micali, M. Rabin, and S. Vadhan, "Verifiable random functions," in *Proc. 40th Annu. Symp. Found. Comput. Sci.*, 1999, pp. 120–130.

[28] O. A. Saraereh, A. Ali, I. Khan, and K. Rabie, "Interference analysis for vehicle-to-vehicle communications at 28 GHz," *Electronics*, vol. 9, no. 2, 2020, Art. no. 262.

[29] J. Kang et al., "Blockchain for secure and efficient data sharing in vehicular edge computing and networks," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4660–4670, Jun. 2019.

[30] I. Hussain and C. Bingcai, "Cluster formation and cluster head selection approach for vehicle ad-hoc network (VANETs) using K-means and floyd-warshall technique," *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 12, pp. 11–15, 2017.

Yingsen Wang (Student Member, IEEE) received the B.S. degree in electronic science and technology in 2019 from the Taiyuan University of Technology, Taiyuan, China, where he is currently working toward the Ph.D. degree with the College of Information and Computer. His research interests include decentralized systems and blockchain technology.

Leiming Yuan received the B.S. degree from Central South University, Hunan, China, in 2019. He is currently working toward the master's degree with the College of Information and Computer, Taiyuan University of Technology, Taiyuan, China. His research interests include decentralized systems and blockchain technology.

Weihan Jiao received the B.S. degree in electrical engineering in 2021 from the Shandong University of Technology, Zibo, China, where he is currently working the master's degree with the School of Electrical and Electronic Engineering, North China Electric Power University, Beijing, China. His research interests include energy markets, energy trading, and energy management.

Yan Qiang (Graduate Student Member, IEEE) received the Ph.D. degree in computer application technology from the Taiyuan University of Technology, Taiyuan, China, in 2010. He is currently a Professor with the College of Information and Computer, Taiyuan University of Technology. He has authored or coauthored more than 70 professional papers. His research interests include medical image processing and Big Data.

Juanjuan Zhao received the Ph.D. degree in computer application technology from the Taiyuan University of Technology, Taiyuan, China, in 2010. She is currently a Professor with the School of Computer Science and Technology, Taiyuan University of Technology. Her research interests include medical image processing and the deep learning.

Qianqian Yang is currently a Lecturer and Senior Engineer with the Jinzhong College of Information, Jinzhong, China. Her main research interests include the big data technology and artificial intelligence. Her e-mail address is evie_yang@qq.com.

Keqin Li (Fellow, IEEE) is currently a SUNY Distinguished Professor of computer science with the State University of New York, New York City, NY, USA. He is also the National Distinguished Professor with Hunan University, Changsha, China. He has authored or coauthored more than 770 journal articles, book chapters, and refereed conference papers and holds nearly 60 patents announced or authorized by the Chinese National Intellectual Property Administration. His research interests include cloud computing, fog computing and mobile edge computing, energy-efficient computing and communication, and intelligent and soft computing. He was the recipient of the several Best Paper awards. He has chaired many international conferences. He is an Associate Editor for *ACM Computing Surveys* and *CCF Transactions on High Performance Computing*. He was with Editorial Boards of the IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, IEEE TRANSACTIONS ON COMPUTERS, IEEE TRANSACTIONS ON CLOUD COMPUTING, IEEE TRANSACTIONS ON SERVICES COMPUTING, and IEEE TRANSACTIONS ON SUSTAINABLE COMPUTING.