International Journal of Electric Power and Energy Studies
ISSN: 3006-2004 (Print), ISSN: 3006-0826 (Online) | Volume 4, Number 2, Year 2025
DOI: https://doi.org/10.62051/ijepes.v4n2.05
Journal homepage: https://wepub.org/index.php/IJEPES/index

G-PBFT Algorithm and its Application in Distributed Energy Trading

Zichao Xu, Mingquan Zhang*, Junxian Zhao

Department of Computer, North China Electric Power University, Baoding 071003, China
*Email: mqzhang@ncepu.edu.cn

ABSTRACT

Distributed energy trading demands high scalability, low latency, and high reliability from the underlying consensus mechanism. However, traditional algorithms represented by Practical Byzantine Fault Tolerance (PBFT), when applied to large-scale, geographically dispersed energy networks, face two core bottlenecks: first, the inherent $O(N^2)$ communication complexity results in massive network overhead and poor scalability; second, the assumption of node homogeneity ignores the heterogeneity of real-world nodes, impacting consensus efficiency and system robustness. To address these challenges, this paper proposes an improved consensus algorithm, G-PBFT (Geohash-based Practical Byzantine Fault Tolerance). To address the scalability bottleneck, the algorithm first employs a geo-aware and latency-optimized intelligent grouping mechanism to partition the large-scale network into multiple low-latency consensus groups. Furthermore, to handle node heterogeneity, the algorithm introduces a multi-dimensional reputation-based dynamic representative election mechanism. By quantitatively evaluating the comprehensive performance of nodes, it ensures the selection of optimal nodes to lead the consensus, naturally forming an efficient two-layer consensus architecture. Simulation results demonstrate that G-PBFT exhibits comprehensive advantages in scalability and robustness. Compared to standard PBFT and various mainstream improved algorithms, G-PBFT maintains a stable throughput of approximately 200 TPS and an average latency below 220ms in a heterogeneous wide-area network with up to 220 nodes. Additionally, comparative experiments in heterogeneous network environments prove that the proposed reputation-based election mechanism can effectively ensure the system's robustness and efficiency, avoiding the catastrophic performance collapse that may result from random election. In conclusion, G-PBFT provides an efficient, scalable, and robust consensus solution for large-scale, geographically dispersed BFT application scenarios.

KEYWORDS

G-PBFT; Consensus Algorithm; Byzantine Fault Tolerance; Distributed Energy Resources; P2P Energy Trading; Scalability; Reputation Mechanism.

1. INTRODUCTION

As the global energy structure transitions towards a "clean" and "low-carbon" environmental model, Distributed Energy Resources (DERs), such as photovoltaics, wind power, and energy storage devices, are being integrated into the power system at an unprecedented rate. In its "14th Five-Year Plan for a Modern Energy System"[1], China explicitly states the need to actively promote the construction of smart microgrids centered on the consumption of new energy to achieve synergy and complementarity with the main power grid. The "Guidelines for the Construction of National Data Infrastructure"[2], released on January 1, 2025, by China National Development and Reform Commission, the National Data Administration, and the Ministry of Industry and Information

Technology, also encourages the exploration of new models for direct green power supply and the orderly development of green electricity and green certificate trading. Against this macroeconomic backdrop, the peer-to-peer (P2P) energy trading model, which allows for direct transactions between energy producers and consumers, is gradually becoming a key means of enhancing energy utilization efficiency and promoting the consumption of green electricity, leveraging its advantages of decentralization and efficient supply-demand matching[3,[4],[5].

Blockchain, a distributed ledger technology, offers promising technical support for the implementation of P2P energy trading through its unique advantages in establishing trust, ensuring security, and enhancing transparency. Through smart contracts and consensus mechanisms, blockchain can automate and securely execute transaction protocols, guaranteeing the validity and non-repudiation of trades. In recent years, researchers have actively explored the application of blockchain in the energy sector. For instance, to address the high cost, low speed, and lack of privacy in continuous double auction blockchains, reference [6] proposed a privacy-preserving scheme for direct microgrid trading that combines a consortium blockchain with continuous double auctions. Reference [7] introduced a machine learning-based power flow distribution identification method to automatically integrate congestion information into the blockchain system, thereby reducing transaction costs for virtual power plants. Reference [8] utilized smart contracts to implement P2P transactions, ensuring the security and decentralization of energy trading among electric vehicles. These studies demonstrate the potential of blockchain in optimizing specific transaction links and ensuring security and privacy.

Although the aforementioned studies have validated the effectiveness of their schemes in small-scale or idealized network environments, they generally evade a critical issue: when the application scenario expands from a community microgrid to a real-world energy internet comprising hundreds or even thousands of geographically dispersed nodes, the underlying consensus mechanism faces a severe performance test. The inherent performance bottlenecks of the BFT consensus algorithms relied upon by these studies make it difficult for these innovations to be implemented on a large scale.

This highlights the core problem that this paper aims to address: in large-scale, geographically dispersed P2P energy trading scenarios, two fundamental challenges emerge at the consensus layer:

(1) **Severe scalability issues caused by communication bottlenecks:** The inherent geographical dispersion of distributed energy nodes leads to significant network latency discrepancies (low latency within the same region, high latency across regions). Traditional BFT algorithms, especially the standard PBFT[9], are fundamentally limited by severe communication bottlenecks arising from their $O(N^2)$ communication complexity. In large-scale scenarios such as distributed energy, as the number of nodes increases, the total volume of messages in the network grows quadratically, rapidly overwhelming network bandwidth and node processing capabilities. The direct consequences are a sharp increase in consensus latency and a steep drop in system throughput, ultimately resulting in extremely poor scalability, rendering the system incapable of meeting the demands of a large-scale network.

(2) **Limitations of node heterogeneity and static mechanisms:** Real-world energy nodes exhibit significant differences in hardware performance, network bandwidth, and historical reliability. Traditional PBFT algorithms simplistically assume that all nodes are homogeneous. Its round-robin primary node selection mechanism cannot proactively leverage high-quality nodes in the network, nor can it effectively avoid "problematic" nodes that are slow to respond due to high load or poor network conditions. This not only impacts consensus efficiency but also reduces the stability and robustness of the entire system.

To address the aforementioned challenges, this paper proposes an improved Practical Byzantine Fault Tolerance algorithm-G-PBFT (Geohash-based Practical Byzantine Fault Tolerance). The core innovations and contributions of G-PBFT are as follows:

(1) To address scalability and communication bottlenecks, a **geo-aware and latency-optimized intelligent grouping mechanism** is constructed. This mechanism utilizes the Geohash algorithm for the initial geographical partitioning of nodes. It innovatively introduces a Latency-Aware Active Group Adjustment (LAGA) mechanism, which allows nodes to dynamically optimize their group affiliation based on measured network latency. This intelligently divides a large-scale network into logical subgroups with high internal communication efficiency, fundamentally resolving the performance bottleneck caused by network-wide broadcasting.

(2) To solve the efficiency and robustness issues arising from node heterogeneity, a **multi-dimensional reputation-based dynamic representative election mechanism** is proposed. This mechanism comprehensively evaluates multiple factors, such as the real-time load and historical consensus success rate of nodes, to dynamically elect the most performant and reliable nodes as representatives to participate in the consensus process. By allowing only these selected representative nodes to participate in the computationally and communication-intensive global consensus, an efficient two-layer consensus architecture is naturally formed. This not only reduces the complexity of the global consensus from $O(N^2)$ to $O(M^2)$ (where $M << N$) but also significantly enhances the consensus efficiency and stability of the entire system.

2. RELATED WORK

2.1. Blockchain Technology and Distributed Energy Trading

Blockchain, originating from Satoshi Nakamoto's vision of Bitcoin as a peer-to-peer electronic cash system[10], is essentially a distributed ledger technology that builds trust and enables secure data sharing among multiple participants. Although initially applied to cryptocurrencies, blockchain technology has demonstrated its transformative potential in numerous fields, including supply chain finance, IoT data management, digital identity, and energy management [11],[12],[13],[14].

In the energy sector, particularly in the context of Distributed Energy Trading, the application of blockchain has garnered significant attention. With the increasing penetration of renewable energy generation, such as solar and wind power, and the widespread adoption of Distributed Energy Resources (DERs), including electric vehicles and user-side energy storage, the traditional centralized model of energy trading and management led by large utility companies is facing numerous challenges. The core features of blockchain offer unique value in addressing the pain points of the traditional model and empowering new P2P trading paradigms:

(1) **Establishing a Decentralized and Trustworthy Trading Environment:** The decentralized nature of blockchain enables energy producers (prosumers) and consumers to trade electricity directly. This reduces transaction costs, improves market matching efficiency, and stimulates user participation [15],[16],[17]. For example, the blockchain-based electricity trading platforms constructed in references [18] and [19] allow residents in community microgrids to directly trade surplus solar power, achieving efficient energy allocation.

(2) **Achieving Transaction Automation and Intelligence via Smart Contracts:** Smart contracts are a vital component of blockchain technology. They are self-executing program codes that automatically handle transactions when specific conditions are met. In distributed energy trading, smart contracts can be used to automatically match supply and demand, settle transactions, and execute contract terms, thereby reducing manual intervention and operational errors [20],[21].

(3) **Enhancing Market Trust and Regulatory Efficacy through Transparency and Immutability:** On the blockchain, all transaction records and critical grid status information (subject to appropriate permissions) can be securely and transparently shared and accessed by all authorized participants. Once recorded on the chain, data is difficult to alter. This not only facilitates supervision and auditing but also significantly enhances the level of trust among participants[22].

In summary, blockchain technology, with its unique technical attributes, presents a broad prospect for building a more efficient, transparent, secure, and user-centric distributed energy trading system. However, despite this promising outlook, the practical implementation of blockchain in large-scale, real-time distributed energy trading systems still needs to overcome key technical obstacles. One of the core challenges is the design of an efficient, scalable, and secure distributed consensus mechanism.

2.2. Distributed Consensus Algorithms and Their Improvements

Consensus algorithms are the cornerstone for building reliable and fault-tolerant distributed systems. Among the numerous consensus algorithms, Proof of Work (PoW) is ill-suited for energy trading scenarios that demand high efficiency and low cost, due to its immense energy consumption and long confirmation delays [23]. While Proof of Stake (PoS) and its variants significantly reduce energy consumption, they face potential centralization risks and have more complex security models [24].

In this context, deterministic BFT consensus algorithms, represented by Practical Byzantine Fault Tolerance (PBFT), achieve consensus through multiple rounds of voting communication among nodes, making them a mainstream choice for consortium chains and high-performance scenarios. PBFT employs a three-phase voting process (pre-prepare, prepare, commit) to achieve finality and can tolerate f malicious nodes among a total of $N \geq 3f + 1$ nodes. Despite its advantage of low latency, PBFT has two major limitations when applied to large-scale, wide-area network (WAN) environments: first, its $O(N^2)$ communication complexity leads to poor scalability and extreme sensitivity to high network latency; second, its assumption of node homogeneity prevents it from adapting to the heterogeneity of real-world networks. To address these issues, researchers have proposed various improvement strategies.

2.2.1. Improvements for Scalability Bottlenecks: Grouping or Layering

To address the scalability problem of PBFT, a mainstream research direction is to reduce the scale of nodes involved in global consensus through grouping or layered architectures. For example, the k-PBFT algorithm proposed in reference [25] innovatively uses k-means clustering for grouping; however, its protocol design still retains a network-wide broadcast step, leaving room for optimization in communication efficiency. Reference [26] constructs a primary-replica multi-chain architecture based on hypergraph theory. The RG-BFT algorithm proposed in reference [27] reduces communication volume by randomly electing a consensus committee. While this method can achieve fairness in general scenarios, its randomness prevents it from perceiving and avoiding low-performance nodes in the network, posing a robustness risk in heterogeneous environments. These works demonstrate the effectiveness of enhancing scalability by grouping or layering nodes to achieve consensus within smaller groups and then coordinating at a higher level through representatives.

However, most of these grouping or sharding schemes adopt generic, location-agnostic partitioning strategies (e.g., hash-based, random allocation), failing to fully leverage the natural geographical clustering of nodes in P2P energy networks to fundamentally optimize intra-group communication efficiency.

2.2.2. Improvements for Heterogeneity and Efficiency: Reputation Mechanisms

Another important research direction for optimizing consensus node selection is the introduction of reputation mechanisms. Electing more reliable leaders by evaluating the historical behavior or status of nodes can effectively enhance the stability and efficiency of the system in heterogeneous environments. For instance, reference [28] designed a reputation mechanism based on physical behaviors such as driving mileage for new energy vehicle energy trading. Reference [29] proposed an optimized PBFT scheme combined with a dynamic credit mechanism. However, most existing reputation models are relatively simplistic or use static weights, making it difficult to comprehensively and dynamically reflect a node's overall performance in terms of computing power,

network conditions, and historical behavior. In particular, research on combining these mechanisms with efficient, geo-aware grouping strategies is still insufficient.

In conclusion, although existing studies such as k-PBFT and RG-BFT have optimized PBFT in various dimensions, a comprehensive solution that can simultaneously address the communication bottlenecks and node heterogeneity issues in large-scale, geographically dispersed scenarios is still lacking. Therefore, designing a consensus mechanism that can both leverage geographical information for efficient grouping and elect optimal nodes through a dynamic reputation model is crucial for large-scale distributed energy trading scenarios. This is the core motivation for proposing the G-PBFT algorithm in this paper.

3. G-PBFT ALGORITHM DESIGN

To address the challenges of scalability, latency sensitivity, and heterogeneous node evaluation methods faced by traditional PBFT algorithms in the large-scale distributed energy trading scenarios introduced in the introduction, this chapter elaborates on our proposed G-PBFT (Geohash-based Practical Byzantine Fault Tolerance) algorithm. The G-PBFT algorithm reconstructs a large-scale, unordered, and heterogeneous network of nodes into an efficient and orderly two-layer consensus architecture through the synergistic operation of two core mechanisms. Figure 1 provides a complete illustration of this process and clarifies the relationships and functions of the key measures within the algorithm: first, the network scalability and communication bottleneck issues are resolved through a geo-aware and latency-optimized intelligent grouping mechanism (detailed in Section 2.1); subsequently, on this basis, the challenges of node heterogeneity are addressed through a multi-dimensional reputation-based dynamic representative election mechanism (detailed in Section 2.2), ultimately forming a stable and robust consensus system.

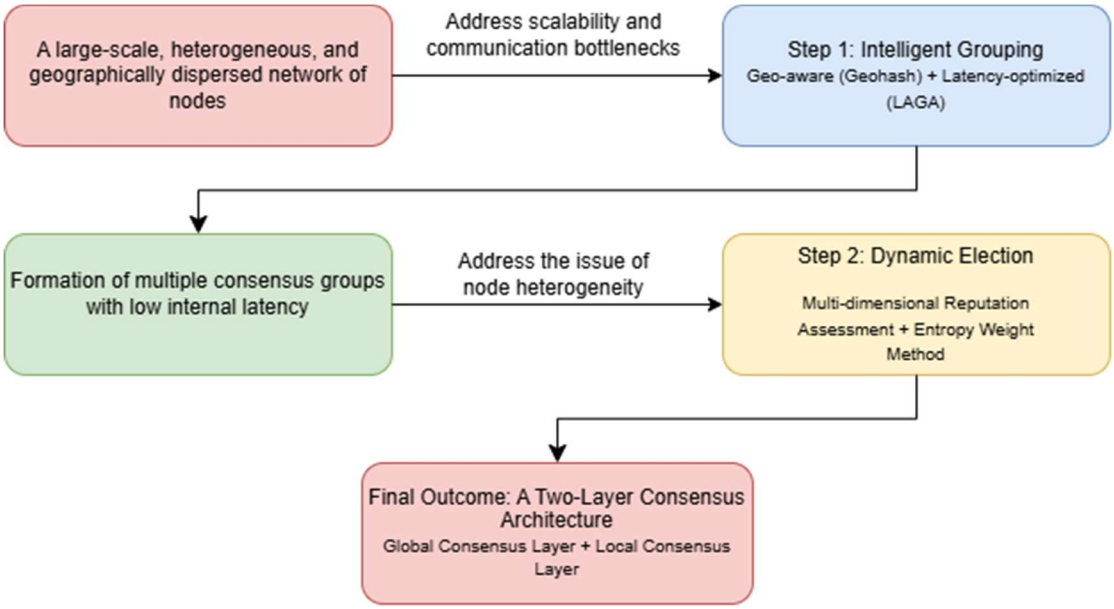

Figure 1. Overall Architecture Diagram

3.1. Geo-aware and Latency-optimized Intelligent Grouping Mechanism

To solve the scalability and communication bottleneck problems of traditional PBFT in geographically dispersed networks, the primary innovation of G-PBFT lies in its intelligent node grouping strategy. This strategy aims to partition a large-scale, geographically dispersed network into several consensus subgroups with high internal communication efficiency and clear external

boundaries, laying a solid foundation for subsequent high-efficiency consensus. This intelligent grouping mechanism is implemented in two phases: first, a preliminary geographical area partitioning is performed using Geohash, followed by fine-grained optimization through the introduction of a Latency-Aware Active Group Adjustment (LAGA) mechanism.

3.1.1. Phase 1: Preliminary Geographical Grouping based on Geohash

This phase utilizes the Geohash algorithm [30] to perform an initial, coarse-grained partitioning of nodes based on their geographical location. Geohash, as an efficient geocoding method, is well-suited as a preliminary partitioning tool for large-scale node networks due to its core features—particularly that "encodings with longer shared prefixes typically represent geographically proximate locations" and "the length of the encoding determines geographical precision."

1) **Global Precision Setting and Geohash Calculation:** The system pre-sets a global Geohash encoding precision, P. Each node N_i calculates its corresponding Geohash string GH_i based on its latitude and longitude coordinates (lat_i, lon_i).

2) **Preliminary Grouping:** All nodes with the same encoding GH_k are initially assigned to the same geographical group $Group_{geo\_k}$

3) **Preliminary Representative Election**: To support the latency probing in the next phase, a preliminary representative node L_{geo_k} is elected within each formed group $Group_{geo\_k}$ based on an initial reputation score. This preliminary representative is primarily used during the latency probing stage of the subsequent LAGA mechanism.

The results of Geohash encoding and preliminary grouping are illustrated in Figure 2.

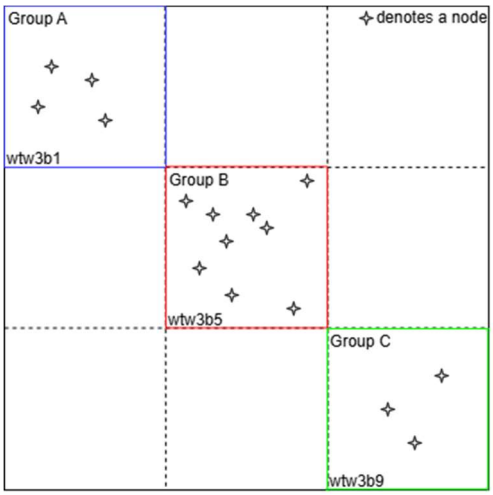

Figure 2. Geohash encoding & Initial grouping

3.1.2. Phase 2: Active Group Adjustment based on Measured Latency (LAGA)

While geographical grouping with Geohash is efficient, its fixed geometric grid partitioning may not perfectly align with complex real-world network topologies, potentially causing "boundary effects" (geographically close nodes with high network latency being incorrectly grouped) or "cross-segment effects" (nodes within the same Geohash cell actually belonging to different network service providers). To overcome these limitations, G-PBFT introduces the Latency-Aware Active Group Adjustment (LAGA) mechanism, which dynamically fine-tunes the initial groupings based on measured network latency.

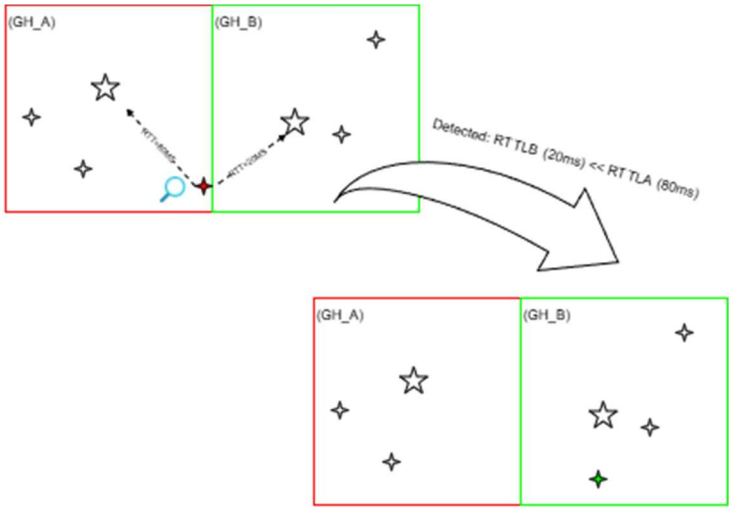

Figure 3. Dynamic node grouping with LAGA

As shown in Figure 3, the core process of the LAGA mechanism is as follows:

1) **Neighboring Representative Discovery:** Each node N_i discovers the representative nodes L_{adj_k} present in the 8 neighboring Geohash cells by querying the areas adjacent to its own Geohash code GH_i.

2) **Network Latency Probing:** Node n_i actively sends lightweight probe messages to its current preliminary group's representative L_{geo} and all discovered neighboring representatives L_{adj_k} to measure and record the actual round-trip times (RTTs), namely $RTT\left(N_i, L_{geo}\right)$ and $RTT\left(N_i, L_{adj_k}\right)$.

3) **Group Affiliation Optimization:** Node N_i compares all the probed RTT data. If it finds a neighboring preliminary representative L_{adj_k} that satisfies $RTT\left(N_i, L_{adj_k}\right) < RTT\left(N_i, L_{geo}\right) - \Delta_{RTT}$, where Δ_{RTT} is a pre-set minimum latency improvement threshold used to prevent frequent switching due to minor network fluctuations, node N_i will adjust its affiliation to the group of the neighboring representative that provides the lowest RTT.

3.1.3. Formation and Advantages of the Final Optimized Consensus Groups

After the two stages of preliminary geographical grouping and the subsequent LAGA dynamic adjustment, the final optimized consensus groups formed in the system are no longer mere static geographical blocks but are logical units that consider both geographical proximity and real-world network latency characteristics. The core advantages of this intelligent grouping mechanism are:

It effectively mitigates the "boundary effects" and mismatches with the actual network topology caused by relying solely on Geohash partitioning.

It ensures that members within the finally formed consensus groups enjoy low network communication latency, providing a solid foundation for G-PBFT's subsequent two-layer consensus mechanism.

Through more rational grouping, it is expected to further reduce the average consensus latency and enhance the system's tolerance to network fluctuations.

3.2. Dynamic Representative Election and Two-Layer Consensus based on Multi-dimensional Reputation

To address the node heterogeneity challenge presented in the introduction and to enhance consensus efficiency in large-scale networks, the core idea of G-PBFT is not to have all nodes participate in every consensus round. Instead, it employs a dynamic election mechanism to select a group of nodes with the best overall performance to act as "representatives" who lead the consensus process.

3.2.1. Necessity of Election and Node Roles

In the design of G-PBFT, nodes are assigned different roles, which are determined through dynamic elections to achieve optimal resource allocation.

Leader: Elected from within each optimized consensus group (formed in Section 2.1) through a reputation evaluation. The Leader is responsible for representing its group in the global consensus and synchronizing the global consensus results with its group members.

Follower: All other nodes in a group besides the Leader. They do not participate in the computationally and communicatively intensive global consensus voting; their role is limited to listening to, verifying, and executing the final consensus results from their Leader.

Primary Global Leader: The node with the highest reputation score among all elected Leaders. It is responsible for receiving client requests, ordering them, and initiating the global consensus process.

3.2.2. Multi-dimensional Reputation Evaluation System

To achieve an objective, fair, and flexible election of representative nodes, G-PBFT designs a multi-dimensional node reputation evaluation algorithm based on the Entropy Weight Method. This algorithm comprehensively considers three key dimensions: historical performance, response success rate, and load capacity. Each evaluation metric is designed as follows:

1) Historical Performance Score S_h

This score measures a node's long-term stability and reliability in consensus activities. This paper calculates the historical performance score using a weighted method. Considering the number of successful responses and failures in the recent k consensus rounds, the score is defined as:

$$S_h = \alpha \frac{N_{success}}{k} + \beta \frac{N_{fail}}{k} \quad (\alpha > 0, \beta < 0, |\alpha| > |\beta|) \tag{1}$$

where α and β are the weights for the success rate and failure rate, respectively, emphasizing success while tolerating a certain proportion of occasional errors.

2) Response Success Rate Score S_s

This score evaluates a node's network performance, representing its real-time responsiveness during the consensus process. It is defined as:

$$S_s = \frac{N_{ack}}{N_{req}} \tag{2}$$

where N_{ack} is the number of successful responses, and N_{req} is the total number of requests.

3) Load Capacity Score S_l

This score is specifically based on the node's CPU usage, memory usage, disk usage, and average load time. To convert these negative indicators (i.e., higher values mean worse performance) into a positive score, the calculation formula is defined as:

$$S_l = 1 - (w_{cpu}U_{cpu} + w_{mem}U_{mem} + w_{disk}U_{disk} + w_{load}T_{load}) \tag{3}$$

where $w_{cpu}, w_{mem}, w_{disk}, w_{load}$ are the pre-set weights for each load metric, with their sum being 1. According to this formula, a higher S_l score indicates that the node has low computational resource consumption and a light load, making it more suitable to serve as a representative node in the consensus process.

3.2.3. Dynamic Weight Allocation based on the Entropy Weight Method

Traditional methods of manually setting weights are susceptible to subjective bias and cannot dynamically adapt to the changing loads and states of the power system. This paper introduces the Entropy Weight Method to objectively and automatically determine the weights of the three evaluation metrics mentioned above. The Entropy Weight Method dynamically determines weights based on the degree of differentiation among indicators, ensuring the scientific and flexible nature of the evaluation.

The calculation steps for the Entropy Weight Method are as follows (assuming there are *m* nodes in a group and *j*=3 evaluation indicators):

1) **Construct the Original Evaluation Matrix:** Build an evaluation matrix $X = (x_{ij})_{m\times 3}$ containing m nodes and 3 indicators.

2) **Normalize the Indicators:** To eliminate the influence of different dimensional units, the matrix is normalized to obtain the standardized matrix $Y = (y_{ij})_{m\times 3}$:

$$y_{ij} = \frac{x_{ij} - \min(x_j)}{\max(x_j) - \min(x_j)} \tag{4}$$

3) **Calculate Information Entropy:** Calculate the information entropy E_j for the j-th indicator:

$$E_j = -k\sum_{i=1}^{m} p_{ij}\ln(p_{ij}) \quad ,\text{where}: p_{ij} = \frac{y_{ij}}{\sum_{i=1}^{m} y_{ij}}, k = \frac{1}{\ln m} \tag{5}$$

4) **Calculate the Weight of Each Indicator:** The final weight w_j for each indicator is calculated based on the information entropy:

$$w_j = \frac{1 - E_j}{\sum_{j=1}^{3}(1 - E_j)} \tag{6}$$

Finally, the comprehensive reputation score for node i is:

$$S_i = w_h S_{h,i} + w_s S_{s,i} + w_l S_{l,i} \tag{7}$$

where w_h, w_s, w_l correspond to the dynamic weights of the three scoring items calculated using the Entropy Weight Method.

3.2.4. Representative Election Process and Formation of the Two-Layer Consensus Architecture

Based on the reputation evaluation model described above, the election of representative nodes is conducted periodically. The process is as follows:

1) Each node updates and shares its latest evaluation metric data.

2) The system calculates the current weights for each metric using the Entropy Weight Method.

3) The comprehensive reputation score S_i is calculated for each node.

4) Within each group, the node with the highest comprehensive score is elected as the group's Leader.

5) Among all newly elected Leaders, the one with the highest comprehensive score is designated as the Primary Global Leader.

Through this election process, G-PBFT's "Two-Layer Consensus Architecture" is naturally formed. As shown in Figure 4, this architecture is divided into two logical layers:

Global Consensus Layer: Composed of all *M* elected representative nodes (Leaders). This layer is responsible for reaching a final, globally consistent BFT consensus on transactions submitted by clients and serves as the system's core decision-making layer.

Local Consensus Layer: Composed of the large number of *N-M* follower nodes within each group. This layer does not directly participate in the global consensus; its responsibilities are greatly simplified to only synchronizing and executing the results agreed upon by their respective Leaders in the global consensus layer.

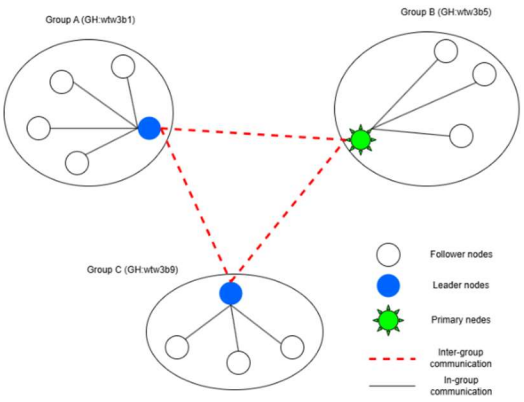

Figure 4. G-PBFT Two-layer consensus architecture

3.2.5. The Complete G-PBFT Consensus Process

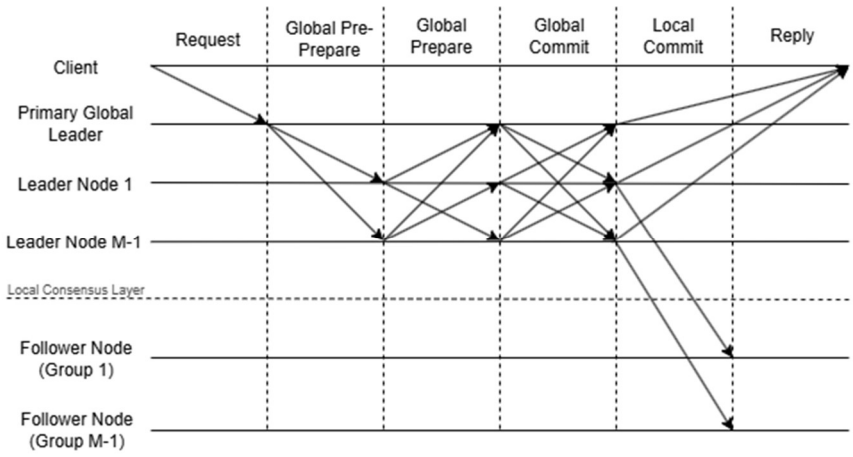

Figure 5. G-PBFT Consensus Process Illustration

Under G-PBFT's two-layer architecture, a complete consensus process is illustrated in Figure 5 and proceeds as follows:

(1) **Request:** The client sends a transaction request to the current Primary Global Leader.

(2) **Global Consensus Stage (Executed at the Global Consensus Layer):**

Global Pre-prepare: After validating the request, the Primary Global Leader broadcasts a GlobalPrePrepare message to all other M-1 Leaders.

Global Prepare: Each Leader validates the message, then broadcasts a GlobalPrepare message to each other and waits to collect $2f_M$ confirmations (where f_M is the maximum number of malicious nodes among the Leaders).

Global Commit: Once a node enters the "globally prepared" state, it broadcasts a GlobalCommit message to others. When a Leader collects $2f_M+1$ GlobalCommit messages, global consensus is considered reached.

(3) **Local Synchronization Stage (Executed at the Local Consensus Layer):**

Result Dissemination (Local Commit): After global consensus is reached, each Leader broadcasts a LocalCommit message, containing the final result and signed by itself, to all Followers within its group.

Local Execution: Followers verify that the LocalCommit message originates from their legitimate Leader and then directly accept the consensus result and execute the corresponding transaction without any additional voting or communication.

Reply: After executing the request locally, each Leader sends the result to the client. The client confirms the successful completion of the request after collecting f_M+1 identical replies from different Leaders.

Through this strictly layered communication mechanism that eliminates network-wide broadcasts, G-PBFT reduces the communication complexity of global consensus from $O(N^2)$ to $O(M^2)$, where M is typically much smaller than N. The communication and computational burden on the vast majority of nodes is significantly reduced, thereby dramatically improving the algorithm's scalability and performance in high-latency networks.

4. SIMULATION EXPERIMENTS AND ANALYSIS

To comprehensively and objectively evaluate the overall performance of the G-PBFT algorithm proposed in this paper, this chapter details a series of simulation experiments.

All tests are conducted in a simulated heterogeneous wide-area network that includes low-performance nodes. We perform a direct, horizontal comparison of G-PBFT against standard PBFT and two representative improved PBFT algorithms (k-PBFT and RG-BFT). The experiments are designed to systematically validate the superiority of G-PBFT over these mainstream solutions by varying the network scale and assessing two core dimensions: scalability and robustness in heterogeneous environments.

4.1. Experimental Environment and Comparison Algorithms

4.1.1. Experimental Platform and Parameter Settings

This study utilizes a modular, concurrent simulation platform built in the Go language. The platform employs Goroutines to simulate independent consensus nodes and Channels to simulate asynchronous message passing between them, enabling the efficient simulation of large-scale distributed networks. To ensure the accuracy and comparability of the experimental results, we designed the following experimental environment and parameter models.

Network Model: All experiments uniformly adopt a Wide-Area Network (WAN) model to simulate a geographically dispersed, real-world energy network environment. The communication latency between nodes is randomly distributed between 50ms and 150ms.

Node Model: All experiments uniformly adopt a heterogeneous model, where 30% of the nodes in the network are designated as "slow nodes" with slower processing speeds, introducing additional delay when handling each message.

This paper uses the following three core metrics to measure algorithm performance:

Throughput (TPS): The number of transactions the system can successfully complete and confirm per second, reflecting the system's maximum processing capacity.

Average Consensus Latency: The average time elapsed from when a client initiates a transaction to when it is finally confirmed by the system, reflecting the system's responsiveness.

Total Communication Overhead: The total number of messages transmitted across the network during the entire simulation, used to measure the algorithm's consumption of network resources.

The software and hardware configurations for the experiment are shown in Table 1:

Table 1. Experimental software and hardware environment configuration

Parameter	Specification
Operating System	Windows 11 Pro 24H2
CPU	Intel(R) Core(TM) i5-12400F
Memory	32.0 GB
Go Version	1.23.3

4.1.2. Comparison Algorithms

To fully validate the advantages of G-PBFT, we selected the following three algorithms as benchmarks for comparison against G-PBFT. They represent different technical routes, from no optimization to optimizations with different focuses:

Standard PBFT: The original Practical Byzantine Fault Tolerance algorithm, where all nodes participate in consensus. Its communication complexity is $O(N^2)$, serving as the minimum performance baseline.

k-PBFT (k-means based PBFT): The algorithm from reference [25], representing the improvement approach of "intelligent grouping + basic reputation." This algorithm uses k-means for clustering and grouping and introduces a basic reputation mechanism, but its communication protocol design includes redundant network-wide broadcasts.

RG-BFT (Random Grouping BFT): The algorithm from reference [27], representing the "random grouping" improvement approach. This algorithm reduces communication overhead by randomly selecting a consensus committee to enhance efficiency and fairness in general scenarios, but its randomness prevents it from avoiding low-performance nodes in heterogeneous networks.

G-PBFT (Our Algorithm): The comprehensive improvement scheme proposed in this paper, with its core being "geo-aware grouping + multi-dimensional dynamic reputation election + efficient layered communication protocol."

4.2. Comprehensive Performance Comparison and Evaluation

The experiments are designed to evaluate the comprehensive performance of each algorithm in an expanding and uncertain heterogeneous network. We incrementally increase the total number of nodes from 40 to 210, while the client sends requests at a constant rate of 200 TPS, and we record the core performance metrics.

4.2.1. Throughput Comparison

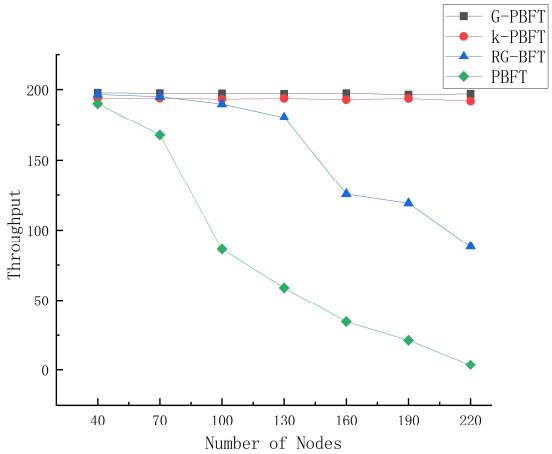

Figure 6. A Comparison of TPS for Varying Node Counts

As shown in Figure 6, G-PBFT and k-PBFT demonstrate the best scalability, thanks to their reliable grouping and reputation mechanisms. They both maintain a stable throughput close to 200 TPS across all node scales, with G-PBFT's average throughput of approximately 197 TPS being slightly better than k-PBFT's 193 TPS. Although RG-BFT and standard PBFT can also handle high throughput in most cases, their performance curves show significant inflection points and instability, reflecting their lack of robustness in heterogeneous networks.

4.2.2. Consensus Latency Comparison

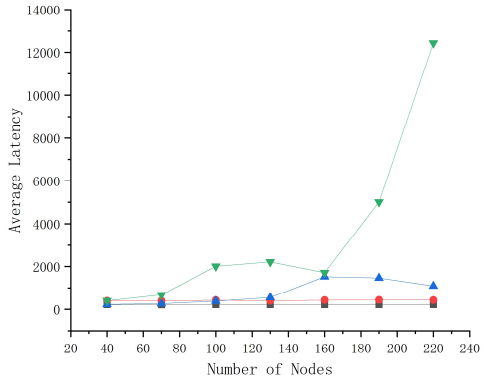

Figure 7. A Comparison of Consensus Latency for Varying Node Counts

As shown in Figure 7, in terms of consensus latency, G-PBFT has the lowest and most stable average latency, remaining at an extremely low level of approximately 214ms across all node scales, almost unaffected by the network's expansion. This is attributable to its efficient communication protocol and excellent grouping strategy. k-PBFT's average latency is also stable but is significantly higher than G-PBFT's, maintaining a level of about 400ms due to the redundant network-wide broadcasts in its protocol.

In contrast, the average latency of RG-BFT and standard PBFT soars exponentially as the number of nodes increases. Notably, at 220 nodes, the latency of standard PBFT reaches as high as 12,433ms (over 12 seconds), meaning most users experience unbearable transaction confirmation delays, and the system is on the verge of failure. This reflects its architectural inadequacy in handling large-scale challenges. The vast difference in latency profoundly reveals the advantages of G-PBFT in ensuring service quality and user experience.

4.2.3. Communication Overhead Comparison

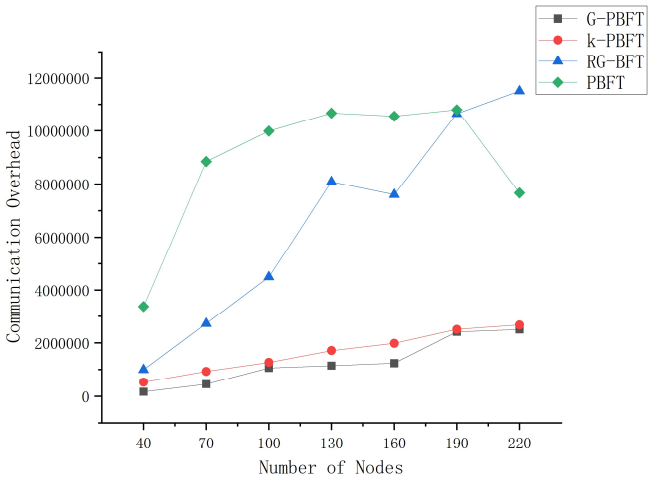

Figure 8. A Comparison of Communication Overhead for Varying Node Counts

As shown in Figure 8, in terms of communication efficiency, G-PBFT and k-PBFT perform the best, with their communication overhead showing a gentle linear increase with the number of nodes. In contrast, the overhead of standard PBFT and RG-BFT exhibits an explosive quadratic growth,

proving that their communication protocols do not fundamentally solve the message storm problem in large-scale networks. It is particularly noteworthy that after the node scale exceeds 190, the total communication overhead of PBFT begins to decrease. This does not contradict the $O(N^2)$ theory but is a clear signal that the system's performance has collapsed. Under the dual pressure of high latency and a large number of nodes, the network becomes severely congested, causing numerous transaction requests to fail in the initial stages of consensus, thus preventing them from entering the subsequent voting phases that generate a large volume of messages. Consequently, the total number of successfully completed transactions drops sharply, leading to a reduction in total communication overhead. This phenomenon serves as reverse proof of the severe scalability bottleneck of standard PBFT, whereas G-PBFT maintains high system efficiency at an extremely low communication cost through its innovative layered architecture.

4.3. Failure Mode Analysis

To ensure the readability of the charts and the clarity of trend comparisons, the performance curves for RG-BFT and standard PBFT shown in Figures 6, 7, and 8 represent their best results from multiple repeated experiments. However, the inherent instability exhibited by these two algorithms in heterogeneous networks is a direct reflection of their core design flaws and warrants in-depth analysis.

In multiple experiments, we observed that the throughput of the RG-BFT algorithm would occasionally plummet to less than 10 TPS, nearly paralyzing the system. The root cause lies in the fundamental conflict between its "random election" mechanism and the heterogeneous network. If one or more "slow nodes" are unfortunately selected into the core consensus committee, these slow nodes immediately become the performance bottleneck for the entire system, leading to a large number of consensus timeouts and failures. The charts display the "lucky" outcomes where this did not occur, but this does not mask the significant risk of a performance avalanche inherent in its design.

For standard PBFT, performance degradation is inevitable, culminating in a complete collapse at 220 nodes (with a throughput of 3.8 TPS). Its performance also fluctuates; in some experiments, the system collapsed prematurely when the node count reached 160. This exposes its dual flaws: first, the $O(N^2)$ communication complexity dictates that its performance will deterministically collapse as the scale increases; second, its simple round-robin primary node mechanism leads to performance instability. When a "slow node" happens to be the primary, it significantly accelerates the system's collapse.

4.4. Comprehensive Analysis and Conclusion

Based on the comprehensive experiments above, the G-PBFT algorithm achieves a leading edge in unified, realistic, heterogeneous network tests because its integrated design addresses the shortcomings of other algorithms:

Compared to standard PBFT, it fundamentally solves the $O(N^2)$ scalability bottleneck through grouping and layering, and it resolves instability in heterogeneous networks via its reputation mechanism.

Compared to RG-BFT, it overcomes the dual deficiencies of random grouping in network efficiency and robustness in heterogeneous environments by using intelligent grouping and reputation-based election, providing predictable and stable high performance.

Compared to k-PBFT, it achieves lower latency and overhead through a more efficient layered communication protocol and a grouping strategy better adapted to geographically distributed scenarios.

The experimental results strongly demonstrate that G-PBFT is a more comprehensively designed solution with no obvious weaknesses, capable of providing efficient, stable, and reliable consensus

support for scenarios such as large-scale, geographically dispersed, and heterogeneous distributed energy trading.

5. CONCLUSION

This paper addresses the core challenges faced by the standard PBFT consensus algorithm when applied to large-scale, geographically dispersed scenarios such as P2P distributed energy trading-namely, poor scalability, sensitivity to network latency, and an inability to adapt to node heterogeneity. To tackle these issues, an improved consensus algorithm, G-PBFT, is proposed.

To resolve the scalability bottleneck, the algorithm designs and implements a geo-aware and latency-optimized intelligent grouping mechanism. This mechanism utilizes Geohash for preliminary geographical partitioning and incorporates the latency-aware LAGA mechanism for dynamic fine-tuning, intelligently dividing the large-scale network into multiple logical subgroups with high internal communication efficiency. This fundamentally solves the performance bottleneck caused by network-wide broadcasts. Furthermore, to address the issue of node heterogeneity, a multi-dimensional reputation-based dynamic representative election mechanism is introduced. By quantitatively evaluating the comprehensive performance of nodes, this mechanism ensures the election of optimal nodes to lead the consensus, effectively enhancing the system's efficiency and robustness while naturally forming an efficient two-layer consensus architecture.

The experimental results demonstrate that, by leveraging its layered grouping and dynamic election mechanisms, G-PBFT significantly overcomes the scalability bottleneck of standard PBFT, maintaining high efficiency and stability even in large-scale, high-latency networks. In conclusion, G-PBFT can effectively guarantee the timeliness and system stability required for scenarios such as large-scale distributed energy trading.

Of course, there is still room for optimization in this study, including in the simulation experiments, the client confirmation mechanism, and the reputation and grouping strategies. Future work can focus on deployment in real-world environments, in-depth security analysis, and energy consumption evaluation to promote the practical application of G-PBFT.

REFERENCES

[1] National Development and Reform Commission, "'Fourteenth Five-Year' Modern Energy System Plan," [Online]. Available: https://www.gov.cn/zhengce/zhengceku/202410/content_6978315.htm.

[2] National Development and Reform Commission, National Data Administration, and Ministry of Industry and Information Technology, "Notice on the Issuance of the 'National Data Infrastructure Construction Guidelines'," [Online]. Available: https://www.gov.cn/zhengce/zhengceku/202501/content_6996487.htm. [Accessed: Mar. 7, 2025].

[3] K. E. Bassey, S. A. Rajput, and K. Oyewale, "Peer-to-peer energy trading: Innovations, regulatory challenges, and the future of decentralized energy systems," World Journal of Advanced Research and Reviews, vol. 24, pp. 172-186, 2024.

[4] H. Zhu, K. Ouahada, and A. M. Abu-Mahfouz, "Peer-to-peer energy trading in smart energy communities: A lyapunov-based energy control and trading system," IEEE Access, vol. 10, pp. 42916-42932, 2022.

[5] Y. Zhou and P. D. Lund, "Peer-to-peer energy sharing and trading of renewable energy in smart communities─trading pricing models, decision-making and agent-based collaboration," Renewable Energy, vol. 207, pp. 177-193, 2023.

[6] K. Fan, J. Liu, and M. Yang, "Privacy Protection of Power Transaction Based on Blockchain," Ordnance Industry Automation, vol. 44, no. 1, pp. 65-69, 109, 2025.

[7] C. Gao, Y. Chen, K. Shi, et al., "Distributed power trading among park microgrid-based virtual power plants based on power flow distribution identification," Distribution & Utilization, vol. 41, no. 8, pp. 112-119, 2024, doi: 10.19421/j.cnki.1006-6357.2024.08.011.

[8] A. Iqbal, A. S. Rajasekaran, G. S. Nikhil, and M. Azees, “A Secure and Decentralized Blockchain Based EV Energy Trading Model Using Smart Contract in V2G Network,” IEEE Access, vol. 9, pp. 75761-75777, 2021.

[9] B. Liskov, “From viewstamped replication to Byzantine fault tolerance,” in Replication: Theory and Practice, Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 121-149.

[10] S. Nakamoto, “Bitcoin: A peer-to-peer electronic cash system,” 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[11] X. Feng, L. Li, D. Liu, et al., “A review of access control research for blockchain data sharing,” Journal of Frontiers of Computer Science and Technology, pp. 1-27, 2025.

[12] Z. Zhen, “Research on green bank financing decision of agricultural supply chain based on blockchain,” Logistics Technology, vol. 44, no. 3, pp. 37-48, 2025.

[13] Q. Shi, “Research on the innovative development mode of cross-border e-commerce service platform based on blockchain technology,” China Journal of Commercevol. 34, no. 5, pp. 102-107, 2025, doi: 10.19699/j.cnki.issn2096-0298.2025.05.102.

[14] Y. Fan, Z. Zhang, T. Qin, et al., “Review of blockchain-based data sharing in Internet of Vehicles,” Application Research of Computers, pp. 1-15, 2025, doi: 10.19734/j.issn.1001-3695.2024.10.0450.

[15] T. Sawa, “Blockchain technology outline and its application to field of power and energy system,” Electrical Engineering in Japan, vol. 206, no. 2, pp. 11-15, 2019.

[16] S. Rahmadika, D. R. Ramdania, and M. Harika, “Security analysis on the decentralized energy trading system using blockchain technology,” Jurnal Online Informatika, vol. 3, no. 1, pp. 44-47, 2018.

[17] Y. Li, N. Li, and Y. Xia, “Research on a data desensitization algorithm of blockchain distributed energy transaction based on differential privacy,” in 2019 IEEE 8th International Conference on Advanced Power System Automation and Protection (APAP), 2019, pp. 980-985.

[18] Z. Jing, “A Distributed Energy Transaction Method Based on Blockchain,” in E3S Web of Conferences, vol. 267, EDP Sciences, 2021, p. 01007.

[19] X. Zhu, “Application of blockchain technology in energy internet market and transaction,” in IOP Conference Series: Materials Science and Engineering, vol. 592, no. 1, IOP Publishing, 2019, p. 012159.

[20] N. Saeed, F. Wen, and M. Z. Afzal, “Decentralized peer-to-peer energy trading in microgrids: Leveraging blockchain technology and smart contracts,” Energy Reports, vol. 12, pp. 1753-1764, 2024.

[21] J. Xie, X. Zhou, S. Wang, et al., “Transaction search engine of distributed electricity market trading platform based on blockchain technology,” Energy Science & Engineering, vol. 10, no. 2, pp. 439-457, 2022.

[22] J. Song, Y. Li, and W. Hu, “Research on Weakly Centralized Trading Mechanism of Distributed Energy Based on Smart Contract of Blockchain,” in 2021 China International Conference on Electricity Distribution (CICED), 2021, pp. 997-1001.

[23] I. F. T. Alyaseen, “Consensus algorithms blockchain: A comparative study,” International Journal on Perceptive and Cognitive Computing, vol. 5, no. 2, pp. 66-71, 2019.

[24] M. S. Ferdous, M. J. M. Chowdhury, M. A. Hoque, et al., “Blockchain consensus algorithms: A survey,” arXiv preprint arXiv:2001.07091, 2020.

[25] Y. Yang, L. Tang, and H. Wang, “Dynamic multi-organizational PBFT algorithm based on k-means,” Journal of Chongqing University, vol. 47, no. 7, pp. 125-139, 2024.

[26] X. Jing and Z. Liu, “Master-Slave Multi-Chain Consensus Mechanism of Consortium Blockchain Based on Hypergraph and MuSig2,” Acta Electronica Sinica, vol. 52, no. 3, pp. 803-813, 2024.

[27] Y. Song, G. Zheng, and X. Zhang, “RG-BFT: Random grouping based Byzantine fault-tolerant algorithm,” Computer Engineering and Design, vol. 45, no. 6, pp. 1661-1667, 2024, doi: 10.16208/j.issn1000-7024.2024.06.009.

[28] H. Zhang, J. Li, K. Liu, et al., “PoRT consensus mechanism for energy trading system of new energy vehicles,” Journal on Communications, vol. 46, no. 1, pp. 157-166, 2025

[29] Z. Liu, F. Wang, and H. Jia, “Optimization Scheme of PBFT Algorithm Combining Dynamic Credit Mechanism,” Computer Engineering, vol. 49, no. 2, pp. 191-198, 2023, doi: 10.19678/j.issn.1000-3428.0063464.

[30] G. Niemeyer, “Geohash,” [Online]. Available: http://geohash.org/. [Accessed: Jun. 2008].