A Privacy-Preserving Blockchain-Based Method to Optimize Energy Trading

Jian Ping, Zheng Yan, *Member, IEEE*, and Sijie Chen, *Member, IEEE*

***Abstract*—It is always desired for optimizing energy trading to disable manipulation and preserve individual privacy. These two features become increasingly appealing for an energy market where interest parties do not mutually trust each other, such as peer-to-peer energy trading. Traditional centralized or hierarchical optimization schemes are vulnerable to an untrusted coordinator who may dishonestly broadcast coordination results or be curious about individual privacy. Recent blockchain-based optimization schemes resist dishonesty but increase the risk of privacy leakage. This paper proposes a privacy-preserving blockchain-based method to optimize energy trading. In the proposed method, participants submit encrypted bids/offers based on a bid/offer encryption algorithm to preserve their privacy. A privacy-preserving-Byzantine-fault-tolerance-based coordination algorithm is proposed to ensure the correctness of trading results with considering dishonesty. Numerical results in a peer-to-peer energy trading case demonstrate the performance of our method on convergence, resisting dishonesty, preserving privacy, and scalability.**

***Index Terms*—Blockchain, privacy-preserving, consensus algorithm, energy trading, distributed optimization.**

I. Introduction

A. Motivation and Related Work

CENTRALIZED optimization has been widely used in energy trading. To do so, a central operator collects all participants' operational parameters and bidding curves and then yields an optimal result. However, centralized optimization is vulnerable to an untrusted center who may manipulate the trading result [1] or collect individual privacy [2]. As alternatives, distributed algorithms and blockchain-based algorithms have been extensively studied.

Manuscript received 10 April 2022; revised 14 July 2022; accepted 8 August 2022. Date of publication 11 August 2022; date of current version 20 February 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 52107115 and Grant U2166201, and in part by the China Postdoctoral Science Foundation under Grant 2020M681299. Paper no. TSG-00489-2022. *(Corresponding author: Sijie Chen.)*

Jian Ping is with the Key Laboratory of Control of Power Transmission and Conversion, Ministry of Education, and the College of Smart Energy, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: ppjj1994@sjtu.edu.cn).

Zheng Yan is with the Key Laboratory of Control of Power Transmission and Conversion, Ministry of Education, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: yanz@sjtu.edu.cn).

Sijie Chen is with the Key Laboratory of Control of Power Transmission and Conversion, Ministry of Education, Shanghai Jiao Tong University, Shanghai 200240, China, and also with Shanghai Non-Carbon Energy Conversion and Utilization Institute, Shanghai 200240, China (e-mail: sijie.chen@sjtu.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TSG.2022.3198165.

Digital Object Identifier 10.1109/TSG.2022.3198165

Distributed algorithms can be categorized as hierarchical schemes and fully distributed schemes. In hierarchical schemes, a central coordinator communicates with participants iteratively until reaching optimum [3], [4], [5]. However, hierarchical schemes are still vulnerable to an untrusted coordinator. On the one hand, a dishonest coordinator may manipulate price signals in the iteration process. Such dishonesty could bring illegal profits to the coordinator and some participants while lowering the welfare of others [6]. On the other hand, though participants only submit limited information in each iteration, a curious coordinator can still estimate the private information of participants by collecting their submissions [7]. In fully distributed schemes, participants directly exchange information with neighbors to reach optimum (e.g., [8], [9], [10]). These fully distributed schemes are valuable in scenarios where a coordinator is absent, but they may converge slower than hierarchical schemes [11]. More importantly, participants also have the motivation and ability to manipulate the exchanged information [12], [13] or collect the privacy of neighbors [14].

Recently, a blockchain-as-coordination-committee framework has been proposed in optimization-based energy trading, such as P2P energy trading [15], [16], [17], EV charging coordination [6], [18], and integrated energy system dispatching [19], [20]. In these studies, a set of pre-elected participants (named delegates) jointly serve as a coordination committee in a hierarchical scheme. A consensus algorithm, e.g., practical Byzantine fault tolerance (PBFT) [16] or proof-of-work (PoW) [6], determines a leader of the committee and allows other delegates (named followers) to oversee the decisions of the leader. This ensures that the signals broadcast by a committee cannot be manipulated by dishonest delegates if and only if the proportion of honest delegates exceeds a threshold. However, participants in a blockchain-based application have to disclose their information to the whole committee. This further increases the risk of privacy leakage [21].

Some other research focuses on preserving individual privacy in optimization. One potential solution is differential privacy, which preserves privacy by adding proper noise. Studies [22], [23], [24] apply differential privacy in optimal power flow (OPF) to preserve the load data or the parameters of networks. However, the added noise inevitably brings optimality losses. Other research studies on cryptographic methods. Study [25] proposes a fully distributed OPF algorithm based on partially homomorphic encryption. In [26], a fully distributed algorithm based on secret sharing preserves privacy in multiarea economic dispatch. However, these

studies assume that participants honestly follow the proposed algorithms, which brings limitations in practice. It is also hard to directly introduce these cryptographic algorithms into the blockchain-as-coordination-committee framework. On a blockchain, a delegate validates messages from other delegates by checking the consistency between the received data and its local data, yet a delegate in cryptographic algorithms receives different encrypted data. This brings difficulties in identifying manipulations.

B. Contributions

Though resisting manipulation and preserving individual privacy become increasingly appealing in energy trading, these two features are seemingly conflicting and are hard to co-exist. This paper proposes a manipulation-resistant and privacy-preserving method to optimize energy trading. The contributions of this paper are summarized as follows:

1) This paper demonstrates that an untrusted (dishonest-or-curious) coordinator in energy trading may hurt the interests of participants. A dishonest coordinator could bring illegal profits to some participants while decreasing the welfare of others by manipulating price signals. A curious coordinator could estimate individual privacy by collecting the submissions of participants. An illustrative example in a P2P energy trading scenario is employed to demonstrate the misbehaviors of an untrusted coordinator and their impacts on participants.

2) This paper proposes a privacy-preserving blockchain-based method to optimize energy trading. To the best of our knowledge, this is the first work that resists manipulation and preserves individual privacy in optimization-based energy trading without sacrificing convergence. The proposed method inherits the hierarchical structure of the blockchain-as-coordination-committee framework. In the local layer, a bid/offer encryption algorithm based on Shamir's secret sharing scheme is proposed to preserve individual privacy. In the upper layer, a privacy-preserving-BFT-based (PP-BFT-based) coordination algorithm is proposed to update the price signals without manipulation and privacy exposure.

3) This paper proves that the proposed method resists the dishonesty of delegates and preserves individual privacy as long as the proportion of untrusted delegates is less than $1/4$.

Compared with traditional distributed algorithms, the proposed method inherits both the merits of hierarchical and fully distributed schemes. The proposed method holds the same convergence ability as hierarchical schemes without needing a central coordinator. More importantly, the proposed method protects the interests of participants by resisting dishonest and curious behaviors.

Compared with the existing blockchain-as-coordination-committee research, the proposed method preserves individual privacy in blockchain-based optimization, which is one of the critical concerns in energy blockchain applications [27]. The performance on resisting dishonesty is only slightly lower than existing blockchains ($1/4$ versus $1/3$ or $1/2$).

C. Organization

The rest of the paper is organized as follows. Section II discusses the limitations of optimization-based energy trading methods. Section III proposes a privacy-preserving blockchain-based method to optimize energy trading. Section IV analyzes the performance of the proposed method. Section V presents simulation results. Section VI concludes the paper.

II. Limitations in Optimization-Based Energy Trading Methods

A. General Form of Optimization-Based Energy Trading Models

The general form of optimization-based energy trading models can be defined as (1)-(5). Constraint (2) and (3) represent the individual constraints. Constraint (4) and (5) represent global constraints in the market.

$$\min \sum_{i\in \boldsymbol{S}} f_i(\boldsymbol{X}_i) \tag{1}$$

$$\boldsymbol{g}_i(\boldsymbol{X}_i) = \boldsymbol{0},\ i \in \boldsymbol{S} \tag{2}$$

$$\boldsymbol{h}_i(\boldsymbol{X}_i) \le \boldsymbol{0},\ i \in \boldsymbol{S} \tag{3}$$

$$\sum_{i\in \boldsymbol{S}} \boldsymbol{a}_i(\boldsymbol{X}_i) = \boldsymbol{0} \tag{4}$$

$$\sum_{i\in \boldsymbol{S}} \boldsymbol{b}_i(\boldsymbol{X}_i) \le \boldsymbol{0} \tag{5}$$

where $\boldsymbol{X}_i$ denotes the J-dimensional bids/offers of participant i, $\boldsymbol{X}$ denotes the vector of $\boldsymbol{X}_i$, $\boldsymbol{S}$ is the set of participants.

B. Hierarchical Optimization With a Single Coordinator

The individual constraints and parameters in model (1)-(5) belong to different participants. Given that a central operator hardly collects all the information, a hierarchical scheme can be designed to solve the model. A widely used algorithm is Lagrangian relaxation (LR) [28], which decomposes the model into local problems (denoted by (6), (2), and (3)) and a master problem (denoted by (7) and (8)). Then, the original model can be solved iteratively.

On the local layer, a participant solves its local problem with given price signals (i.e., Lagrangian multipliers) and then submits its bids/offers to a coordinator. On the upper layer, a coordinator updates price signals according to bids/offers from all participants. The iteration process converges to optimum when the convergence criterion (9) is satisfied. For simplicity, this paper uses a sub-gradient algorithm to update price signals. Other updating algorithms which may have better convergence in a large-scale system, such as the quasi-Newton method [29], are also applicable for the proposed method.

$$\min f_i(\boldsymbol{X}_i) + \boldsymbol{\lambda}^{(\upsilon)\mathrm{T}} \boldsymbol{a}_i(\boldsymbol{X}_i) + \boldsymbol{\mu}^{(\upsilon)\mathrm{T}} \boldsymbol{b}_i(\boldsymbol{X}_i) \tag{6}$$

$$\boldsymbol{\lambda}^{(\upsilon+1)} = \boldsymbol{\lambda}^{(\upsilon)} + \frac{1}{\zeta + \eta\upsilon} \sum_{i\in \boldsymbol{S}} \boldsymbol{a}_i\left(\boldsymbol{X}_i^{(\upsilon)}\right) \tag{7}$$

$$\mu_j^{(\upsilon+1)} = \begin{cases} 0, \text{if } \mu_j^{(\upsilon)} = 0 \text{ and } b_{ij}\left(\boldsymbol{X}_i^{(\upsilon)}\right) \le 0 \\ \mu_j^{(\upsilon)} + \frac{1}{\zeta+\eta\upsilon} \sum_{i\in \boldsymbol{S}} b_{ij}\left(\boldsymbol{X}_i^{(\upsilon)}\right), \text{else} \end{cases} \tag{8}$$

$$\max\left\{ \frac{\left\|\boldsymbol{\lambda}^{(\upsilon+1)} - \boldsymbol{\lambda}^{(\upsilon)}\right\|}{\left\|\boldsymbol{\lambda}^{(\upsilon)}\right\|}, \frac{\left\|\boldsymbol{\mu}^{(\upsilon+1)} - \boldsymbol{\mu}^{(\upsilon)}\right\|}{\left\|\boldsymbol{\mu}^{(\upsilon)}\right\|} \right\} \le \varepsilon \tag{9}$$

where $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$ are the Lagrangian multipliers of constraint (4) and (5) (also known as the market prices), respectively, υ is

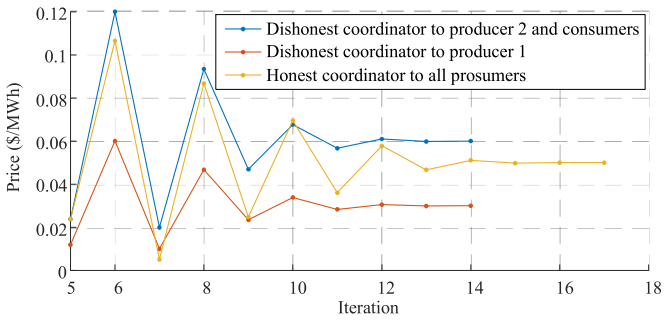


Fig. 1. Iteration process with a dishonest coordinator or an honest coordinator.

the iteration number, ζ, η are the parameters of the step size, $b_{ij}(\boldsymbol{X}_i^{(\upsilon)})$ is the jth local inequality constraints, ε is a small positive number.

C. Limitations of Hierarchical Schemes

The hierarchical scheme releases the computational burden of a central operator, but it remains several limitations:

1) Vulnerability to a dishonest coordinator. The hierarchical scheme requires the coordinator to obey the rule of price signal updating. A dishonest coordinator could manipulate price signals during the iteration process, which results in a non-optimal solution. The solution may hurt the interests of some participants while bringing illegal profits to others.

2) Vulnerability to a curious coordinator. A curious coordinator attempts to collect individual privacy by collecting the bids/offers of participants. Though a participant only needs to provide limited information in each iteration, the bids/offers submitted in all iterations may still disclose its gradient information or constraints.

An illustrative example in a P2P energy trading scenario is presented to demonstrate the above vulnerabilities. Assuming there are four prosumers in the market. Two of them are consumers who have the same utility function $U_i(P_i^{\mathrm{B}}) = -0.005(P_i^{\mathrm{B}})^2 + 0.15P_i^{\mathrm{B}}$ and the same load limits $2 \le P_i^{\mathrm{B}} \le 15$, where P_i^{B} denotes the demand quantity. Other two of them are producers who have the same cost function $C_i(P_i^{\mathrm{S}}) = 0.0025(P_i^{\mathrm{S}})^2$ and the same output limits $0.5 \le P_i^{\mathrm{S}} \le 12$, where P_i^{S} denotes the supply quantity. The market should meet the energy balance constraint, denoted by $\sum P_i^{\mathrm{B}} - \sum P_i^{\mathrm{S}} = 0$. The Lagrangian multiplier of the energy balance constraint also represents the market clearing price.

Assuming a dishonest coordinator colludes with producer 2. During the iteration process, the coordinator always sends lower price signals to producer 1 while sending correct price signals to other prosumers. As illustrated in Fig. 1, compared with in a market cleared by an honest coordinator, producer 1 would face a lower price while others face a higher price. As shown in Table I, the dishonesty brings illegal profits to the coordinator and producer 2, but lowers the profits of others.

A curious coordinator may collect bids/offers of prosumers at different prices. By doing so, the coordinator can form the supply/demand curves of prosumers and then estimate the cost/utility functions and limits. As illustrated in Fig. 2, the coordinator precisely estimates the personal information of consumer 1 and producer 1 according to their bids/offers.

TABLE I
PROFITS OF PROSUMERS AND COORDINATOR

	Honest coordinator ($)	Dishonest coordinator ($)
Producer 1	0.25	0.09
Producer 2	0.25	0.36
Consumer 1&2	0.5	0.40
Coordinator	0	0.18

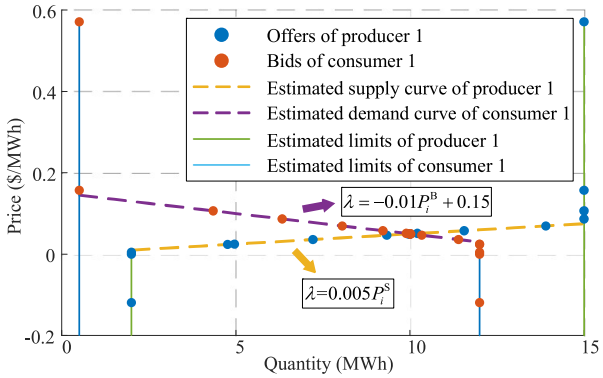


Fig. 2. Personal information estimated by a curious coordinator.

III. PRIVACY-PRESERVING BLOCKCHAIN-BASED METHOD

A. Design Goals

The privacy-preserving blockchain-based method should achieve the following design goals:

1) Resistance to untrusted delegates. Some delegates may be untrusted, i.e., cooperatively manipulate the trading results or collect individual privacy. The proposed method should resist these misbehaviors. Here we assume that there are N delegates and no more than f of them are untrusted.

2) Tolerance for offline participants. Participants may be offline temporally in some iterations due to communication failures. The proposed method should still converge to the optimum with considering offline participants. Here we assume that there are M participants and no more than d of them are offline in an iteration.

3) Effectiveness in a weakly synchronous communication network. In such a network, messages are guaranteed to be delivered after a certain time delay that is bounded, yet the bound is unknown to participants and delegates [30]. This setting is widely used in blockchain consensus algorithms (e.g., PBFT [31], RBFT [32], Zyzzyva [33]) and mimics the communication features in reality. The proposed method should be effective in such a network.

B. Structure

The proposed method has a hierarchical structure, as illustrated in Fig. 3. In the local layer, a participant optimizes its bids/offers by solving the local problem. A participant would generate pieces of encrypted bids/offers. Then, it submits the encrypted bids/offers to the upper layer.

TABLE II
DEFINITIONS OF MESSAGES IN THE PROPOSED ALGORITHM

Message	Information	Definition
M_{in}^{Secret}	$< \text{Secret}, [\boldsymbol{X}_i^{(\upsilon)}]_n, < \upsilon, n >^{\delta_i} >^{\delta_i}$	Secret message sent from participant i to delegate n
$M_{nk}^{\text{Pre-p}}$	$< \text{Pre-p}, \upsilon, \Omega^{(\upsilon)}, [\boldsymbol{a}_i]_n, [\boldsymbol{b}_i]_n >^{\delta_n}$	Pre-prepare message sent from delegate n to delegate k
M_{nk}^{Prepare}	$< \text{Prepare}, \upsilon, \Omega^{(\upsilon)}, [\boldsymbol{a}_i]_n, [\boldsymbol{b}_i]_n >^{\delta_n}$	Prepare message sent from delegate n to delegate k
M_{nk}^{Commit}	$< \text{Commit}, \upsilon >^{\delta_n}$	Commit message sent from delegate n to delegate k
M_{n}^{Conv}	$< \text{Conv}, \upsilon >^{\delta_n}$	Convergence message sent from delegate n
M_{n}^{Update}	$< \text{Update}, \lambda_j^{(\upsilon+1)}, \mu_j^{(\upsilon+1)}, \upsilon >^{\delta_n}$	Update message sent from delegate n
$M_{in}^{\text{L-C}}$	$< \text{L-C Request}, \upsilon >^{\delta_i}$	Leader-change request message sent from participant i to delegate n

Superscript δ_i denotes the message is signed by i, $[\cdot]_n$ denotes a piece of encrypted data held by delegate n, $\Omega^{(\upsilon)}$ contains $< \upsilon, n >^{\delta_i}$ from all received M_{in}^{Secret}, $[\boldsymbol{a}_i]_n$ denotes $[\sum_{i\in\boldsymbol{S}} \boldsymbol{a}_i(\boldsymbol{X}_i^{(\upsilon)})]_n$, $[\boldsymbol{b}_i]_n$ denotes $[\sum_{i\in\boldsymbol{S}} \boldsymbol{b}_i(\boldsymbol{X}_i^{(\upsilon)})]_n$.

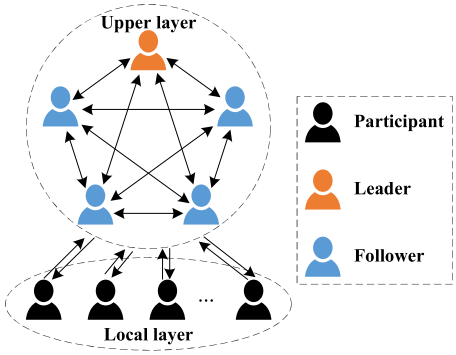


Fig. 3. Structure of the proposed method.

In the upper layer, a set of delegates serve as a coordination committee. They cooperatively update and broadcast price signals according to the encrypted data. Delegates act as leaders in rotation during the iteration process. Non-leader delegates are named followers. A PP-BFT coordination algorithm is proposed to resist the dishonesty of delegates.

C. Delegate Selection

Before the iteration process, the delegates should be pre-elected from all participants. Each participant could vote for a delegate. The top N participants in voting serve as delegates. In iteration υ, delegate l (where $l = \upsilon \bmod \text{N}$) is assigned as the leader. If delegate l behaves dishonestly, then the next delegate, i.e., delegate l' (where $l' = (\upsilon+1) \bmod \text{N}$), becomes the new leader in iteration υ.

D. Algorithm

Each iteration has five regular phases: Secret uploading, Pre-prepare, Prepare, Commit, and Reply. If the leader in iteration υ behaves dishonestly, an additional View-change phase is triggered. The algorithms of participants and delegates are detailed in this subsection. Table II defines messages in the proposed algorithm. Fig. 4 illustrates the flowchart of the proposed method in an iteration.

1) Bid/offer encryption algorithm for participants.

Algorithm 1 defines the participants' rules. *Briefly speaking, a participant generates* N *pieces of encrypted data and separately submits each piece of data to each delegate.*

Algorithm 1 Bid/Offer Encryption Algorithm for Participant C_i

1: //Secret uploading phase
2: Calculate $\boldsymbol{X}_i^{(\upsilon)}$ with equation (6), (2), and (3)
3: Construct polynomials $f_{ij}^{(\upsilon)}(\xi) = x_{ij}^{(\upsilon)} + \sum_{l=1}^{\text{f}} a_{jl}^{(\upsilon)} \xi^l$, where $a_{jl}^{(\upsilon)}$ are randomly generated, $x_{ij}^{(\upsilon)}$ is the jth element of $\boldsymbol{X}_i^{(\upsilon)}$.
4: **for** $n = 1 : \text{N}$ **do**
5: **for** $j = 1 : \text{J}$ **do**
6: Generate encrypted bids/offers per $[x_{ij}^{(\upsilon)}]_n = f_{ij}^{(\upsilon)}(n)$
7: **end for**
8: Submit M_{in}^{Secret} to delegate n
9: **end for**
10: //Reply phase
11: **if** Receive M_{ni}^{Update} from no less $\text{f}+1$ delegates which contained the same $\lambda_j^{(\upsilon+1)}, \mu_j^{(\upsilon+1)}$ during $\text{T}^{\max}$ since submitting M_{in}^{Secret} **then**
12: Step into iteration $\upsilon+1$ and return to Line 2
13: **else if** Receive M_{ni}^{Conv} from no less $\text{f}+1$ delegates during $\text{T}^{\max}$ since submitting M_{in}^{Secret} **then**
14: The optimal solution is derived
15: **else**
16: Send $M_{in}^{\text{L-C}}$ to all delegates
17: **end if**

In the Secret uploading phase, a participant firstly optimizes its bids/offers (Line 2). After that, it generates N pieces of encrypted bids/offers based on Shamir's Secret Sharing scheme [34] (Line 3-7). Each piece of encrypted bids/offers is packed into a Secret message and submitted to the corresponding delegate (Line 9).

A participant would wait for the feedback from delegates for $\text{T}^{\max}$ since submitting the Secret message. If a participant receives no less than $\text{f}+1$ Update message or Convergence message with the same detailed information in time, it would step into the next iteration (Line 12) or reach the optimum (Line 14). Otherwise, a participant would send a Leader-change request message to all delegates because the current leader may behave dishonestly (Line 16). The rest actions of participants and delegates are similar to the View-change protocol in PBFT [31], hence are omitted here.

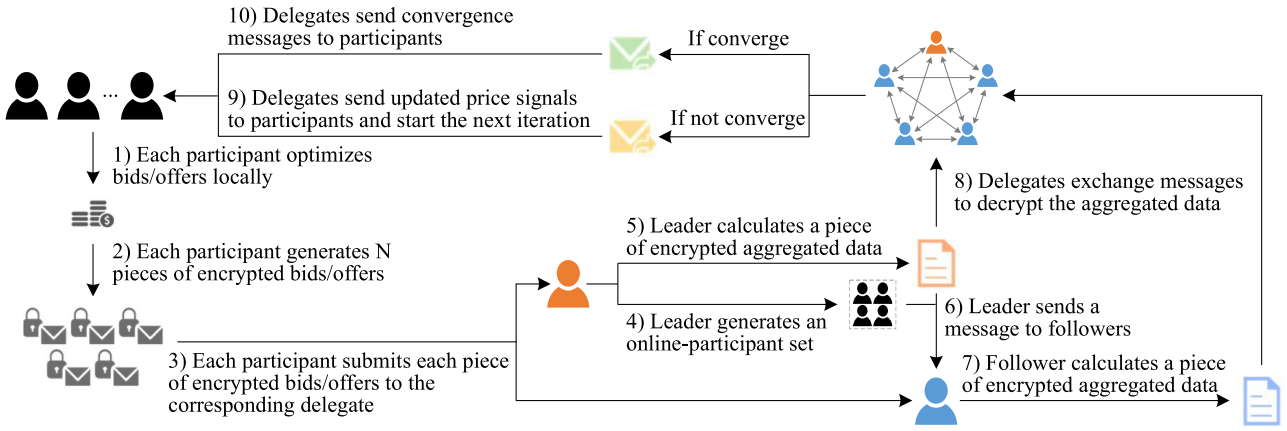


Fig. 4. Flowchart of the proposed method.

The bid/offer encryption algorithm has the following properties:

i) It is a (f+1, N) threshold scheme. That is, one can decrypt the actual bids/offers of a participant by using (10) only if it collects at least $\mathrm{f}+1$ pieces of the encrypted bids/offers.

$$x_{ij}^{(\upsilon)} = \sum_{n=1}^{\mathrm{f}+1} [x_{ij}^{(\upsilon)}]_n \prod_{m=1, m\neq n}^{\mathrm{f}+1} \frac{-m}{m-n} \tag{10}$$

where $x_{ij}^{(\upsilon)}$ is the jth element of $\boldsymbol{X}_i^{(\upsilon)}$, $[x_{ij}^{(\upsilon)}]_n$ denotes the corresponding encrypted bids/offers sent to delegate n.

ii) It is addictive homomorphic. That is, one can decrypt the sum of participants' bids/offers by using (11) only if it collects at least $\mathrm{f}+1$ pieces of the sum of participants' encrypted bids/offers.

$$x_{ij}^{(\upsilon)} + x_{i'j}^{(\upsilon)} = \sum_{n=1}^{\mathrm{f}+1} \left([x_{ij}^{(\upsilon)}]_n + [x_{i'j}^{(\upsilon)}]_n \right) \prod_{m=1, m\neq n}^{\mathrm{f}+1} \frac{-m}{m-n} \tag{11}$$

To the best of our knowledge, Shamir's Secret Sharing scheme is the simplest scheme with the above properties in the real number field. Our method is still effective with other schemes which have the same properties, such as Brickell scheme [35].

2) PP-BFT-based coordination algorithm for delegate D_n.

Algorithm 2 defines the rules of delegates. The algorithm inherits the philosophy of PBFT because PBFT performs better on throughput and computational cost than PoW, proof-of-stake (PoS), and delegated PoS (DPoS) [1]. *The major difference is that participants need to submit messages to all delegates and the validation rules of delegates are modified.*

In the Secret uploading phase, a delegate collects Secret messages from participants (Line 2).

In the Pre-prepare phase, when a leader receives Secret messages from no less than $\mathrm{M}-\mathrm{d}$ participants, it generates an online-participant set $\Omega^{(\upsilon)}$ as a proof of receiving Secret messages from mentioned participants (Line 5). After that, a leader calculates a piece of the encrypted aggregated data (i.e., left-hand sides in (4) and (5)) according to (12) and (13) (Line 6). The first equal marks in (12) and (13) hold because of the addictive homomorphic feature. To ensure convergence, for participants in $\Omega^{(\upsilon)}$, the encrypted bids/offers in the current iteration are used for calculation; for participants not in $\Omega^{(\upsilon)}$, the encrypted bids/offers in the last iteration are used, which are represented by the second equal marks in (12) and (13).

$$\begin{aligned}\left[\sum_{i\in\boldsymbol{S}} \boldsymbol{a}_i\left(\boldsymbol{X}_i^{(\upsilon)}\right)\right]_n &= \sum_{i\in\boldsymbol{S}} \boldsymbol{a}_i\left(\left[\boldsymbol{X}_i^{(\upsilon)}\right]_n\right) \\ &= \sum_{i\in\Omega^{(\upsilon)}} \boldsymbol{a}_i\left(\left[\boldsymbol{X}_i^{(\upsilon)}\right]_n\right) + \sum_{i\notin\Omega^{(\upsilon)}} \boldsymbol{a}_i\left(\left[\boldsymbol{X}_i^{(\upsilon-1)}\right]_n\right)\end{aligned} \tag{12}$$

Algorithm 2 PP-BFT-Based Coordination Algorithm for Delegate D_n

1: //Secret uploading phase
2: Collect M_{in}^{Secret} from participants
3: //Pre-prepare phase
4: **if** Delegate D_n is the leader **then**
5: When receiving M_{in}^{Secret} from no less than $\mathrm{M}-\mathrm{d}$ participants, generate $\Omega^{(\upsilon)}$
6: Calculate $[\boldsymbol{a}_i]_n$, $[\boldsymbol{b}_i]_n$
7: Send $M_{nk}^{\text{Pre-p}}$ to all followers
8: **end if**
9: //Prepare phase
10: **if** Delegate D_n is a follower **then**
11: When receiving $M_{kn}^{\text{Pre-p}}$, wait for collecting all M_{in}^{Secret} from participants mentioned in $M_{kn}^{\text{Pre-p}}$
12: Calculate $[\boldsymbol{a}_i]_n$, $[\boldsymbol{b}_i]_n$
13: Send M_{nk}^{Prepare} to other delegates
14: **end if**
15: //Commit phase
16: When receiving M_{kn}^{Prepare} with $\Omega^{(\upsilon)}$ from no less than 3f delegates, send M_{nk}^{Commit} to other delegates
17: //Reply phase
18: When receiving M_{kn}^{Commit} from no less than $3\mathrm{f}+1$ delegates, identify the correct results of $\sum_{i\in\mathrm{S}} \boldsymbol{a}_i(\boldsymbol{X}_i^{(\upsilon)})$ and $\sum_{i\in\mathrm{S}} \boldsymbol{b}_i(\boldsymbol{X}_i^{(\upsilon)})$ by executing Algorithm 3
19: Update $\lambda_j^{(\upsilon+1)}$ and $\mu_j^{(\upsilon+1)}$ by (7) and (8)
20: **if** Convergence criterion (9) is satisfied **then**
21: Send M_n^{Conv} to all participants
22: **else**
23: Send M_n^{Update} to all participants
24: **end if**

$$\left[\sum_{i\in\boldsymbol{S}}\boldsymbol{b}_i\left(\boldsymbol{X}_i^{(\upsilon)}\right)\right]_n=\sum_{i\in\boldsymbol{S}}\boldsymbol{b}_i\left(\left[\boldsymbol{X}_i^{(\upsilon)}\right]_n\right)$$
$$=\sum_{i\in\Omega^{(\upsilon)}}\boldsymbol{b}_i\left(\left[\boldsymbol{X}_i^{(\upsilon)}\right]_n\right)+\sum_{i\notin\Omega^{(\upsilon)}}\boldsymbol{b}_i\left(\left[\boldsymbol{X}_i^{(\upsilon-1)}\right]_n\right) \quad (13)$$

Then, a leader sends a Pre-prepare message to all followers (Line 7). The Pre-prepare message presents a proposal for an online-participant set and a piece of the encrypted aggregated data.

In the Prepare phase, when a follower receives a Pre-prepare message, it recognizes that participants in $\Omega^{(\upsilon)}$ are online. A follower waits for receiving all Secret messages from these participants (Line 11). After that, it calculates a piece of the encrypted aggregated data according to (12) and (13) (Line 12). Then, it sends a Prepare message to all delegates (Line 13). The Prepare message proves that the follower has also received Secret messages from participants in $\Omega^{(\upsilon)}$ and presents a proposal for a piece of the encrypted aggregated data.

In the Commit phase, when a delegate receives no less than 3f Prepare messages which have the same online-participant set as the Pre-prepare message, it sends a Commit message to other delegates (Line 16). The Commit message asserts that trusted delegates already have the same online-participant set and present proposals for a piece of the encrypted aggregated data.

In the Reply phase, when a delegate receives no less than $3\mathrm{f}+1$ Commit messages, it executes Algorithm 3 to derive the decrypted aggregated data (Line 18). *Briefly speaking, a delegate enumerates all possible aggregated data. The majority of these possible results are deemed as the correct aggregated data.* A delegate updates Lagrangian multipliers according to the correct aggregated data (Line 19). If the convergence criterion (9) is satisfied, it broadcasts a Convergence message to all participants (Line 21); otherwise, it broadcasts an Update message containing the updated Lagrangian multipliers to all participants (Line 23).

Algorithm 3 Aggregated Data Decryption Algorithm

1: Let $\boldsymbol{D}^{(\upsilon)}$ denote the set of delegates which sent Commit messages
2: Generate all subsets of $\boldsymbol{D}^{(\upsilon)}$: $\Gamma_1^{(\upsilon)},\ldots,\Gamma_u^{(\upsilon)}$, where each subset has $\mathrm{f}+1$ delegates. Obviously, $u=\binom{3\mathrm{f}+1}{\mathrm{f}+1}$
3: **for** $l=1:u$ **do**
4: Calculate $\boldsymbol{s}_l^{(\upsilon)}=\{\sum_{i\in\boldsymbol{S}}\boldsymbol{a}_i(\boldsymbol{X}_i^{(\upsilon)}),\sum_{i\in\boldsymbol{S}}\boldsymbol{b}_i(\boldsymbol{X}_i^{(\upsilon)})\}$ according to $\{[\sum_{i\in\boldsymbol{S}}\boldsymbol{a}_i(\boldsymbol{X}_i^{(\upsilon)})]_n,[\sum_{i\in\boldsymbol{S}}\boldsymbol{b}_i(\boldsymbol{X}_i^{(\upsilon)})]_n\}$ provided by delegates in $\Gamma_l^{(\upsilon)}$
5: **end for**
6: Output the major result in $\{\boldsymbol{s}_1^{(\upsilon)},\ldots,\boldsymbol{s}_u^{(\upsilon)}\}$ as the correct aggregated data

IV. Performance Analysis

A. Resistance to Dishonest Behavior

Theorem 1: If the proportion of untrusted delegates is less than 1/4, i.e., $4\mathrm{f}+1\leq\mathrm{N}$, the proposed method ensures the correctness of the trading results.

Proof: 1) The pre-prepare phase and the prepare phase guarantee that all trusted delegates agree on the same online-participant set. More precisely, if a trusted delegate believes $\Omega^{(\upsilon)}$ (i.e., it receives no less than 3f Prepare messages which contain $\Omega^{(\upsilon)}$), none of the other trusted delegates would believe $\Omega^{'(\upsilon)}$. This is because other delegates cannot receive 3f Prepare messages which contain $\Omega^{'(\upsilon)}$. Therefore, all trusted delegates would only broadcast pieces of encrypted aggregated data according to the same participant set.

2) The Reply phase guarantees the correctness and the consensus of the updated Lagrangian multipliers. More precisely, when a delegate receives no less than $3\mathrm{f}+1$ Commit messages, it means that there are at least $2\mathrm{f}+1$ Commit messages sent by trusted delegates. Hence, a trusted delegate can derive correct aggregated data for at least $\binom{2\mathrm{f}+1}{\mathrm{f}+1}$ times in Algorithm 3. Untrusted delegates may manipulate their pieces of encrypted aggregated data to let delegates decrypt incorrect aggregated data. However, even if f of $3\mathrm{f}+1$ Commit messages are sent by untrusted delegates, a trusted delegate would derive the incorrect aggregated data for at most $\binom{2\mathrm{f}}{\mathrm{f}+1}$ times [36]. Hence, a trusted delegate always outputs the correct aggregated data in Algorithm 3. When a participant receives no less than $\mathrm{f}+1$ Update message or Convergence message which contains the same information, it can trust the message because it has been verified by at least one trusted delegate.

3) The Commit phase and the View-change phase resist an untrusted leader. The philosophy is the same as PBFT [31] and is omitted in this paper.

To sum up, untrusted delegates cannot manipulate the Lagrangian multipliers. This naturally ensures the correctness of the trading results.

B. Resistance to Curious Behavior

Theorem 2: When at least two participants are online in each iteration, i.e., $\mathrm{M}-\mathrm{d}\geq 2$, f untrusted delegates cannot obtain participants' privacy even through collaboration.

Proof: 1) In the Secret uploading phase, untrusted delegates collect no more than f pieces of encrypted bids/offers from a participant. According to the first property of the bid/offer encryption algorithm, it is not enough to decrypt the actual bids/offers of a participant.

2) In the Pre-prepare phase, a leader generates an online-participant set which contains no less than $\mathrm{M}-\mathrm{d}$ participants. Hence, the encrypted aggregated data in the Pre-prepare phase and the Prepare phase include at least two participants. The bids/offers of a participant are still unknown to untrusted delegates.

3) In and after the Prepare phase, untrusted delegates only collect the information of aggregated data, whereas the encrypted bids/offers of a participant would not be transmitted among delegates.

To sum up, untrusted delegates cannot obtain participants' privacy during the iteration process.

TABLE III
COMPARISON OF OPTIMIZATION-BASED ENERGY TRADING METHODS

	LR	PBFT	Proposed
Communication complexity	$O(1)$	$O(\mathrm{N}^2)$	$O(\mathrm{N}^2)$
Dishonesty tolerance	No	$\mathrm{f} \leq (\mathrm{N}-1)/3$	$\mathrm{f} \leq (\mathrm{N}-1)/4$
Preserving privacy	No	No	Yes
Examples	[2]–[5]	[16]	This paper

C. Comparison

To better clarify the proposed method's merits and shortages, Table III compares different optimization-based energy trading methods.

Compared with the traditional LR method, the blockchain-based methods (the PBFT-based and the proposed method) have higher communication complexity in the upper layer because of the information exchange among delegates. However, the blockchain-based methods resist the dishonesty of delegates. Compared with the PBFT-based method, though the proposed method performs slightly worse on dishonesty tolerance, it effectively preserves privacy.

V. Simulation Results

The proposed method is tested in a P2P energy trading scenario, which is formulated as (14)-(20). The objective of the model is to maximize the total welfare of participants. Constraints (15) and (16) represent the utility functions of consumers and the cost functions of producers, respectively. Constraints (17) and (18) represent the power limits of participants. Constraint (19) is the power balance constraint. Constraint (20) represents the power flow limit. Obviously, constraints (15)-(18) are local constraints while (19) and (20) are global constraints.

$$\max \sum_{i \in \boldsymbol{S}^{\mathrm{B}}} U_i\left(P_i^{\mathrm{B}}\right) - \sum_{j \in \boldsymbol{S}^{\mathrm{S}}} C_j\left(P_j^{\mathrm{S}}\right) \tag{14}$$

$$U_i\left(P_i^{\mathrm{B}}\right) = -\frac{1}{2}\theta_i^{\mathrm{B}}\left(P_i^{\mathrm{B}}\right)^2 + \beta_i^{\mathrm{B}} P_i^{\mathrm{B}}, \forall i \in \boldsymbol{S}^{\mathrm{B}} \tag{15}$$

$$C_j\left(P_j^{\mathrm{S}}\right) = a_j^{\mathrm{S}}\left(P_j^{\mathrm{S}}\right)^2 + b_j^{\mathrm{S}} P_j^{\mathrm{S}}, \forall j \in \boldsymbol{S}^{\mathrm{S}} \tag{16}$$

$$P_i^{\mathrm{B,min}} \leq P_i^{\mathrm{B}} \leq P_i^{\mathrm{B,max}}, \forall i \in \boldsymbol{S}^{\mathrm{B}} \tag{17}$$

$$P_j^{\mathrm{S,min}} \leq P_j^{\mathrm{S}} \leq P_j^{\mathrm{S,max}}, \forall j \in \boldsymbol{S}^{\mathrm{S}} \tag{18}$$

$$\sum_{j \in \boldsymbol{S}^{\mathrm{S}}} P_j^{\mathrm{S}} - \sum_{i \in \boldsymbol{S}^{\mathrm{B}}} P_i^{\mathrm{B}} = 0 : \lambda^{\mathrm{E}} \tag{19}$$

$$-L_l^{\max} \leq \sum_{j \in \boldsymbol{S}^{\mathrm{S}}} \alpha_{jl} P_j^{\mathrm{S}} - \sum_{i \in \boldsymbol{S}^{\mathrm{B}}} \alpha_{il} P_i^{\mathrm{B}} \leq L_l^{\max}, \forall l \in \boldsymbol{S}^{\mathrm{L}} : \mu_l^-, \mu_l^+ \tag{20}$$

where $P_i^{\mathrm{B}}/P_j^{\mathrm{S}}$ are the amounts of the energy bought by consumer i/sold by producer j, $U_i(P_i^{\mathrm{B}})$ is the utility function of consumer i, with corresponding coefficients $\theta_i^{\mathrm{B}}, \beta_i^{\mathrm{B}}$, $C_j(P_j^{\mathrm{S}})$ is the cost function of producer j with corresponding coefficients $a_j^{\mathrm{S}}, b_j^{\mathrm{S}}$, $\boldsymbol{S}^{\mathrm{B}}/\boldsymbol{S}^{\mathrm{S}}$ are the sets of consumers/producers, $P_i^{\mathrm{B,min}}/P_i^{\mathrm{B,max}}$ are the minimum/maximum demands of consumer i, $P_j^{\mathrm{S,min}}/P_j^{\mathrm{S,max}}$ are the minimum/maximum outputs of producer j, α_{jl} is the power transfer distribution factor from participant j to line l, $\boldsymbol{S}^{\mathrm{L}}$ is the set of lines, λ^{E} is the Lagrangian multiplier of the power balance constraint (19), μ_l^-/μ_l^+ are the Lagrangian multipliers of power flow limit constraints (20) for line l.

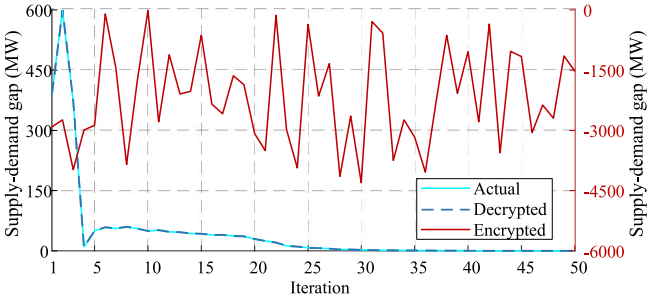


Fig. 5. Supply-demand gap during the iteration process.

The clearing price of a participant can be calculated by (21).

$$\pi_i = \lambda^{\mathrm{E}} - \sum_{l \in \boldsymbol{S}^{\mathrm{L}}} \alpha_{il}\left(\mu_l^+ - \mu_l^-\right) \tag{21}$$

where π_i is the clearing price of participant i.

To demonstrate the performance on convergence, preserving privacy, resisting dishonesty, and scalability, the proposed method is compared with the PBFT-based method and the traditional LR-based method on an IEEE 9-bus system [37] and an IEEE 118-bus system [38].

A. Case 1: IEEE 9-Bus System

This case demonstrates the performance of the proposed method on convergence, preserving privacy, and resisting dishonesty. The system contains three producers and six consumers [37]. It is assumed that at most two participants are offline in each iteration. In the PP-BFT-based method and the PBFT-based method, five participants are pre-elected as delegates. Delegate 1 is assumed to be untrusted, which intends to manipulate the trading results and collect privacy.

1) Convergence: Fig. 5 illustrates the supply-demand gap during the iteration process in the PP-BFT-based method. In the Reply phase, a delegate can derive the correct gap according to Prepare messages received from other delegates. Hence a trusted delegate can correctly update the price signal. The proposed method reaches the optimum after fifty iterations, which is the same as the number of iterations in the PBFT-based and LR-based methods.

2) Performance on preserving privacy: Fig. 6 compares the submissions of producer 2 in different energy trading methods. For simplicity, the submissions to trusted delegates are omitted. In the PP-BFT-based method, the submissions received by delegate 1 are always oscillating and different from the actual energy supply of producer 2. In contrast, delegate 1 in the PBFT-based method or the coordinator in the LR-based method can collect the actual offers of producer 2 during the iteration process.

Fig. 7 shows the estimated supply curves of producer 2. Under the PBFT-based method and the LR-based method, the

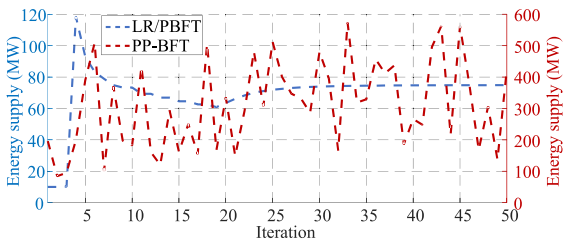


Fig. 6. Submissions of producer 2 during the iteration process.

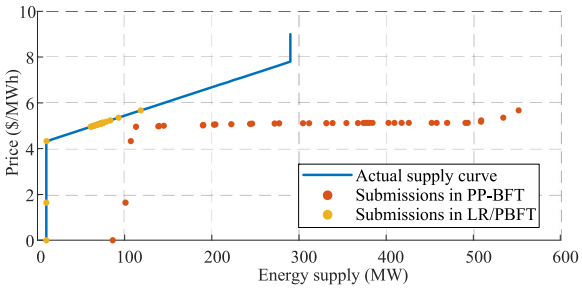


Fig. 7. Actual supply curve and submissions of producer 2.

TABLE IV
TRADING RESULTS WITH CONSIDERING DISHONESTY

	LR		PBFT/PP-BFT	
	λ^{E} ($/MWh)	Profit ($)	λ^{E} ($/MWh)	Profit ($)
Producer 1	5.52	333.17	5.13	258.84
Other producers	4.97	121.40	5.13	152.28
Consumers	5.52	106.14	5.13	253.28
Coordinator	/	97.52	/	0

lower limit of the output and the cost function are successfully estimated according to the submissions of producer 2. However, under the PP-BFT-based method, delegate 1 fails to obtain the personal information about producer 2.

3) Performance on resisting dishonesty: Table IV compares the trading results under different methods if delegate 1 or the coordinator manipulates λ^{E}. The power flow constraint (20) is not bounded in the case hence μ_l^-/μ_l^+ are omitted here. According to Table IV, an untrusted coordinator in the LR-based method brings illegal profits to the coordinator and producer 1 but lowers the profits of others. In the PBFT-based/PP-BFT-based methods, delegate 1 fails to manipulate the trading results because prosumers would trust messages from other trusted delegates.

Moreover, in the PP-BFT-based method, delegate 1 may manipulate the received pieces of encrypted aggregated data. Fig. 8 illustrates the enumeration results with considering this dishonesty, where the node size reflects the number of a result in the enumeration. It can be observed that the major results in iterations are still the same as the actual supply-demand gap. Hence, delegate 1 fails to disturb the decryption process.

B. Case 2: IEEE 118-Bus System

This case analyzes the scalability of the proposed method. The system contains fifty-four producers and ninety-one consumers [38]. The loads of consumers are 1.1 times than the original values to mimic a congestion scenario. It is assumed that at most five participants are offline in each iteration. As shown in Fig. 9, the algorithm converges after 110 iterations. The following experiment is performed in MATLAB R2020b on a Dell desktop computer with an Intel i7-10700 CPU and 16-GB memory. The communication time between two peers (including prosumers, delegates, and the coordinator) is assumed to obey an exponential distribution with an expected value $\Delta \mathrm{T} = 0.02\mathrm{s}$ [39].

Table V compares the computational time of different methods with different numbers of delegates. i) The PP-BFT-based method has higher computational time than the other two methods, which is mainly because of the additional communication time. The additional computational time reflects the computational cost of resisting dishonesty and preserving privacy. ii) The computational time of the PP-BFT-based method can still meet the requirement of day-ahead or hour-ahead P2P energy trading. (iii) With the increasing number of delegates, the proposed method can tolerate more untrusted delegates, while the corresponding computational time just slightly increases.

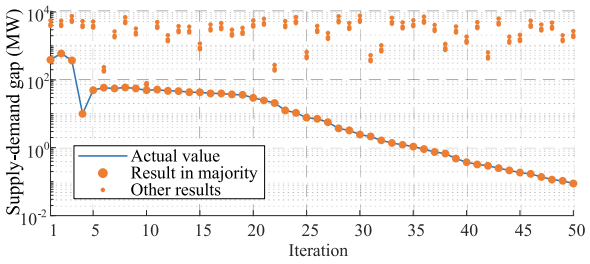


Fig. 8. Enumeration results influenced by untrusted delegates.

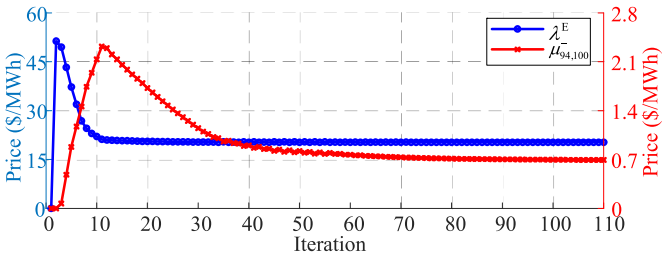


Fig. 9. Iteration process of the proposed method in case 2.

TABLE V
COMPUTATIONAL TIME OF DIFFERENT METHODS

	LR	PBFT	PP-BFT	PBFT	PP-BFT
Number of delegates	/	5	5	21	21
Tolerated number of dishonest delegates	/	1	1	6	5
Preserving privacy	No	No	Yes	No	Yes
Local layer computation (s)	14.05	13.63	14.51	13.95	13.77
Upper layer computation (s)	0.05	0.01	0.01	0.07	0.10
Communication (s)	24.64	49.16	67.83	52.13	74.56
Total (s)	38.74	62.80	82.35	66.15	88.43

VI. Conclusion

This paper focuses on the trust issues in optimization-based energy trading. Unlike existing methods which are vulnerable to dishonest or curious behaviors, the proposed method resists both misbehaviors without sacrificing convergence. Numerical results in P2P energy trading cases demonstrate the performance on resisting dishonesty, preserving privacy, and scalability.

For engineers, this work can be applied in various scenarios such as P2P energy trading, integrated energy system operation, and virtual power plant formation. For researchers, one can develop a blockchain platform integrated with the proposed method; one can explore privacy-preserving energy trading methods with lower communication complexity; one can develop privacy-preserving energy trading methods which can solve mixed-integer optimization problems.

References

[1] Y. Jiang, K. Zhou, X. Lu, and S. Yang, "Electricity trading pricing among prosumers with game theory-based model in energy blockchain environment," *Appl. Energy*, vol. 271, Aug. 2020, Art. no. 115239.

[2] Z. Li, W. Wu, B. Zhang, and B. Wang, "Decentralized multi-area dynamic economic dispatch using modified generalized benders decomposition," *IEEE Trans. Power Syst.*, vol. 31, no. 1, pp. 526–538, Jan. 2016.

[3] Q. Hu, Z. Zhu, S. Bu, K. W. Chan, and F. Li, "A multi-market nanogrid P2P energy and ancillary service trading paradigm: Mechanisms and implementations," *Appl. Energy*, vol. 293, Jul. 2021, Art. no. 116938.

[4] C. Shao, X. Wang, M. Shahidehpour, X. Wang, and B. Wang, "Partial decomposition for distributed electric vehicle charging control considering electric power grid congestion," *IEEE Trans. Smart Grid*, vol. 8, no. 1, pp. 75–83, Jan. 2017.

[5] A. Hassan, S. Acharya, M. Chertkov, D. Deka, and Y. Dvorkin, "A hierarchical approach to multienergy demand response: From electricity to multienergy applications," *Proc. IEEE*, vol. 108, no. 9, pp. 1457–1474, Sep. 2020.

[6] J. Ping, Z. Yan, S. Chen, L. Yao, and M. Qian, "Coordinating EV charging via blockchain," *J. Mod. Power Syst. Clean Energy*, vol. 8, no. 3, pp. 573–581, May 2020.

[7] C. Zhang and Y. Wang, "Enabling privacy-preservation in decentralized optimization," *IEEE Trans. Control Netw. Syst.*, vol. 6, no. 2, pp. 679–689, Jun. 2019.

[8] T. Xu, W. Wu, W. Zheng, H. Sun, and L. Wang, "Fully distributed quasi-Newton multi-area dynamic economic dispatch method for active distribution networks," *IEEE Trans. Power Syst.*, vol. 33, no. 4, pp. 4253–4263, Jul. 2018.

[9] E. Sorin, L. Bobo, and P. Pinson, "Consensus-based approach to peer-to-peer electricity markets with product differentiation," *IEEE Trans. Power Syst.*, vol. 34, no. 2, pp. 994–1004, Mar. 2019.

[10] Y. Wang, L. Wu, and S. Wang, "A fully-decentralized consensus-based ADMM approach for DC-OPF with demand response," *IEEE Trans. Smart Grid*, vol. 8, no. 6, pp. 2637–2647, Nov. 2017.

[11] A. Kargarian *et al.*, "Toward distributed/decentralized DC optimal power flow implementation in future electric power systems," *IEEE Trans. Smart Grid*, vol. 9, no. 4, pp. 2574–2594, Jul. 2018.

[12] C. Zhao, J. He, P. Cheng, and J. Chen, "Analysis of consensus-based distributed economic dispatch under stealthy attacks," *IEEE Trans. Ind. Electron.*, vol. 64, no. 6, pp. 5107–5117, Jun. 2017.

[13] S. Chen, L. Zhang, Z. Yan, and Z. Shen, "A distributed and robust security-constrained economic dispatch algorithm based on blockchain," *IEEE Trans. Power Syst.*, vol. 37, no. 1, pp. 691–700, Jan. 2022.

[14] C. Zhang, M. Ahmad, and Y. Wang, "ADMM based privacy-preserving decentralized optimization," *IEEE Trans. Inf. Forensics Security*, vol. 14, pp. 565–580, 2019.

[15] K. Zhou, J. Chong, X. Lu, and S. Yang, "Credit-based peer-to-peer electricity trading in energy blockchain environment," *IEEE Trans. Smart Grid*, vol. 13, no. 1, pp. 678–687, Jan. 2022.

[16] Q. Yang and H. Wang, "Blockchain-empowered socially optimal transactive energy system: Framework and implementation," *IEEE Trans. Ind. Informat.*, vol. 17, no. 5, pp. 3122–3132, May 2021.

[17] H. Haggi and W. Sun, "Multi-round double auction-enabled peer-to-peer energy exchange in active distribution networks," *IEEE Trans. Smart Grid*, vol. 12, no. 5, pp. 4403–4414, Sep. 2021.

[18] J. Ping, Z. Yan, and S. Chen, "A two-stage autonomous EV charging coordination method enabled by blockchain," *J. Mod. Power Syst. Clean Energy*, vol. 9, no. 1, pp. 104–113, Jan. 2021.

[19] S. Chen *et al.*, "A trusted energy trading framework by marrying blockchain and optimization," *Adv. Appl. Energy*, vol. 2, May 2021, Art. no. 100029.

[20] S. Chen *et al.*, "A blockchain consensus mechanism that uses proof of solution to optimize energy dispatch and trading," *Nat. Energy*, vol. 7, pp. 495–502, Jun. 2022.

[21] T. Wang, H. Hua, Z. Wei, and J. Cao, "Challenges of blockchain in new generation energy systems and future outlooks," *Int. J. Electr. Power Energy Syst.*, vol. 135, Aug. 2022, Art. no. 107499.

[22] T. W. Mak, F. Fioretto, L. Shi, and P. Van Hentenryck, "Privacy-preserving power system obfuscation: A bilevel optimization approach," *IEEE Trans. Power Syst.*, vol. 35, no. 2, pp. 1627–1637, Mar. 2020.

[23] F. Fioretto, T. W. K. Mak, and P. V. Hentenryck, "Differential privacy for power grid obfuscation," *IEEE Trans. Smart Grid*, vol. 11, no. 2, pp. 1356–1366, Mar. 2020.

[24] V. Dvorkin, F. Fioretto, P. Van Hentenryck, P. Pinson, and J. Kazempour, "Differentially private optimal power flow for distribution grids," *IEEE Trans. Power Syst.*, vol. 36, no. 3, pp. 2186–2196, May 2021.

[25] T. Wu, C. Zhao, and Y.-J. A. Zhang, "Privacy-preserving distributed optimal power flow with partially homomorphic encryption," *IEEE Trans. Smart Grid*, vol. 12, no. 5, pp. 4506–4521, Sep. 2021.

[26] N. Tian, Q. Guo, H. Sun, and X. Zhou. "Fully Privacy-Preserving Distributed Optimization Based on Secret Sharing." TechRxiv Preprint. 2021. [Online]. Available: https://doi.org/10.36227/techrxiv.15087774.v1

[27] M. Andoni *et al.*, "Blockchain technology in the energy sector: A systematic review of challenges and opportunities," *Renew. Sustain. Energy Rev.*, vol. 100, pp. 143–174, Feb. 2019.

[28] A. J. Conejo, E. Castillo, R. Mínguez, and R. García-Bertrand, *Decomposition Techniques in Mathematical Programming: Engineering and Science Applications*. Berlin, Germany: Springer, 2006.

[29] Z. Li, W. Wu, B. Zhang, H. Sun, and Q. Guo, "Dynamic economic dispatch using Lagrangian relaxation with multiplier updates based on a quasi-Newton method," *IEEE Trans. Power Syst.*, vol. 28, no. 4, pp. 4516–4527, Nov. 2013.

[30] C. Dwork, N. Lynch, and L. Stockmeyer, "Consensus in the presence of partial synchrony," *J. ACM*, vol. 35, no. 2, pp. 288–323, Apr. 1988.

[31] M. Castro and B. Liskov, "Practical Byzantine fault tolerance," in *Proc. OSDI*, New Orleans, LA, USA, 1999, pp. 173–186.

[32] P. L. Aublin, S. B. Mokhtar, and V. Quema, "RBFT: Redundant Byzantine fault tolerance," in *Proc. IEEE 33rd Int. Conf. Distrib. Comput. Syst.*, Philadelphia, PA, USA, 2013, pp. 297–306.

[33] R. Kotla, L. Alvisi, M. Dahlin, A. Clement, and E. Wong, "Zyzzyva: Speculative Byzantine fault tolerance," *ACM Trans. Comput. Syst.*, vol. 27, no. 4, p. 39, Nov. 2008.

[34] A. Shamir, "How to share a secret," *Commun. ACM*, vol. 22, no. 11, pp. 612–613, Nov. 1979.

[35] E. F. Brickell, "Some ideal secret sharing schemes," in *Proc. Workshop Theory Appl. Cryptograph. Techn.*, 1989, pp. 468–475.

[36] L. Harn and C. Lin, "Detection and identification of cheaters in (t, n) secret sharing scheme," *Des. Codes Cryptogr.*, vol. 52, no. 1, pp. 15–24, Jan. 2009.

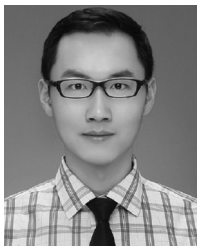
[37] A. Paudel, L. P. M. I. Sampath, J. Yang, and H. B. Gooi, "Peer-to-peer energy trading in smart grid considering power losses and network fees," *IEEE Trans. Smart Grid*, vol. 11, no. 6, pp. 4727–4737, Nov. 2020.

[38] H. Ye, "Surrogate affine approximation based co-optimization of transactive flexibility, uncertainty, and energy," *IEEE Trans. Power Syst.*, vol. 33, no. 4, pp. 4084–4096, Jul. 2018.

[39] J. Mišić, V. B. Mišić, X. Chang, and H. Qushtom, "PBFT-based ordering service for IoT domains," in *Proc. IEEE 92nd Veh. Technol. Conf.*, Victoria, BC, Canada, 2020, pp. 1–6.

Jian Ping received the B.E. and Ph.D. degrees from Shanghai Jiao Tong University, Shanghai, China, in 2015 and 2020, respectively, where he is currently a Postdoctoral Fellow. His research interests include energy blockchain, transactive energy systems, and the electricity market.

Zheng Yan (Member, IEEE) received the B.S. degree in electrical engineering from Shanghai Jiao Tong University, Shanghai, China, in 1984, and the M.S. and Ph.D. degrees in electrical engineering from Tsinghua University, Beijing, China, in 1987 and 1991, respectively. He is currently a Professor of Electrical Engineering with Shanghai Jiao Tong University. His research interests include the application of optimization theory to power systems, power markets, and dynamic security assessment.



Sijie Chen (Member, IEEE) received the B.E. and Ph.D. degrees from Tsinghua University, Beijing, in 2009 and 2014, respectively. He was an Assistant Research Professor with the Department of Electrical Engineering and Computer Science, Washington State University from 2014 to 2016. He is currently a Tenure Track Associate Professor of Electrical Engineering, Shanghai Jiao Tong University, Shanghai, China. His research interests include energy blockchain, demand response, transactive energy system, and the electricity market.