Privacy-Preserving Energy Trading Using Blockchain and Zero Knowledge Proof

Dongkun Hou, Jie Zhang, Sida Huang, Zitian Peng, Jieming Ma, and Xiaohui Zhu
School of Advanced Technology, *Xi'an Jiaotong-Liverpool University*, Suzhou, China

{Dongkun.Hou20, Sida.Huang16, Zitian.Peng20}@student.xjtlu.edu.cn
{Jie.Zhang01, Jieming.Ma, Xiaohui Zhu}@xjtlu.edu.cn

2022 IEEE International Conference on Blockchain (Blockchain) | 978-1-6654-6104-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/BLOCKCHAIN55522.2022.00064

***Abstract*—Microgrid research and construction effectively use distributed energy resources to stimulate clean energy development. Peer-to-peer (P2P) energy trading in microgrids helps create a fully competitive and autonomous energy trading market. Blockchain technology is employed to realize a P2P energy trading framework. However, while public blockchain is generally transparent, energy trading information is sensitive and thus requires a privacy-preserving mechanism. In this paper, we design a privacy-preserving energy trading mechanism by using blockchain and zero-knowledge proofs. A user uploads a commitment to the blockchain instead of the original bid amount, and the zero-knowledge proof of commitment is uploaded into the blockchain. Other participants can verify the correctness of energy trading in each auction match. Our experiments show the design is feasible and efficient in Ethereum although there are more gas consumptions.**

***Index Terms*—Energy Trading, Double Auction, Blockchain, Smart Contract, Energy Data Privacy, Zero Knowledge Proof**

I. Introduction

Traditional energy trading mainly adopts a centralized and optimal decision-making approach for resource allocation, which has shortcomings in terms of high cost, fragility to attack, and poor protection of consumer privacy. Under the current globalized trading model, energy trading demands many third-party administrators to maintain trading credits and organize consumer participation, leading to higher transaction costs and inefficient management.

Blockchain technology with fair, transparent, and decentralized characteristics has a very wide application prospect in distributed energy trading. The participants in the blockchain-based trading market are distributed and peer-to-peer (P2P). The energy transaction can be recorded and checked in the blockchain, and the rule energy trading is executed automatically by smart contract without the participation of trusted third-party institutions. There is an increasing number of studies on employing blockchain for energy trading, and some platforms have been realized and applied in the real-world energy market.

In blockchain-based energy trading, users conduct truly decentralized energy transactions without the requirement for intermediaries. For example, users are able to act as sellers when their solar panels generate excess energy production to facilitate the integration of renewable energy sources. P2P trading provides high flexibility to cope with energy deficiency and network congestion.

In recent years, many market mechanisms for P2P trading platforms rely on the double auction design [1]. Consumers and producers are able to publish their orders separately, and the market is in a continuous clearing state. [2] presented a three-layer blockchain-based energy trading platform and realized a full smart contract of the double auction process in the energy market. [3] proposed an energy trading mechanism based on the blockchain and double auction to achieve the coordination and complementarity of the energy microgrid system and promote economic benefit by using the inner layer autonomous scheduling model and analyzing the price acceptance probability indicator. [4] developed a double auction energy trading system that enables prosumers and auctioneers to adjust the bid amount to maximize benefits based on Stackelberg equilibrium.

The transparency of the blockchain gives the energy market great openness and robustness in a distributed situation, but in energy trading, individual energy trading data is sensitive, and therefore a privacy protection solution is required. [5] developed a blockchain-based privacy-preserving energy trading system to protect users' account characteristics by introducing dummy accounts. [6] and [7] designed the privacy-protection scheme of blockchain-based microgrids transaction and the energy information is hidden by threshold secret sharing technology, but the hidden information may be leaked if the number of malicious participants exceeds a threshold value. [8] proposed a decentralized application of privacy-preserving demand response programs to hide the amount of energy consumption in the blockchain; however, the design cannot support the P2P energy trading among the users' nodes.

The existing blockchain-based energy trading methods have the drawbacks of the privacy leakage of energy information or the lack of P2P trading. Therefore, the main contribution of this paper is to design a privacy-preserving energy trading

This research is supported by the National Natural Science Foundation of China under (Grant No. 62002296), the Natural Science Foundation of Jiangsu Province under (Grant No. BK20200250), the Suzhou Science and Technology Project-Key Industrial Technology Innovation (Grant No. SYG202006, SYG202122), the Future Network Scientific Research Fund Project (Grant No. FNSRFP-2021-YB-41), the XJTLU Key Programme Special Fund under (Grant No. KSF-E-54, KSF-E-65), the XJTLU Research Development Fund (Grant No. RDF-17-02-04), the XJTLU AI University Research Centre and Jiangsu Provincial Data Science and Cognitive Computational Engineering Research Centre at XJTLU.

978-1-6654-6104-7/22/$31.00 ©2022 IEEE
DOI 10.1109/Blockchain55522.2022.00064

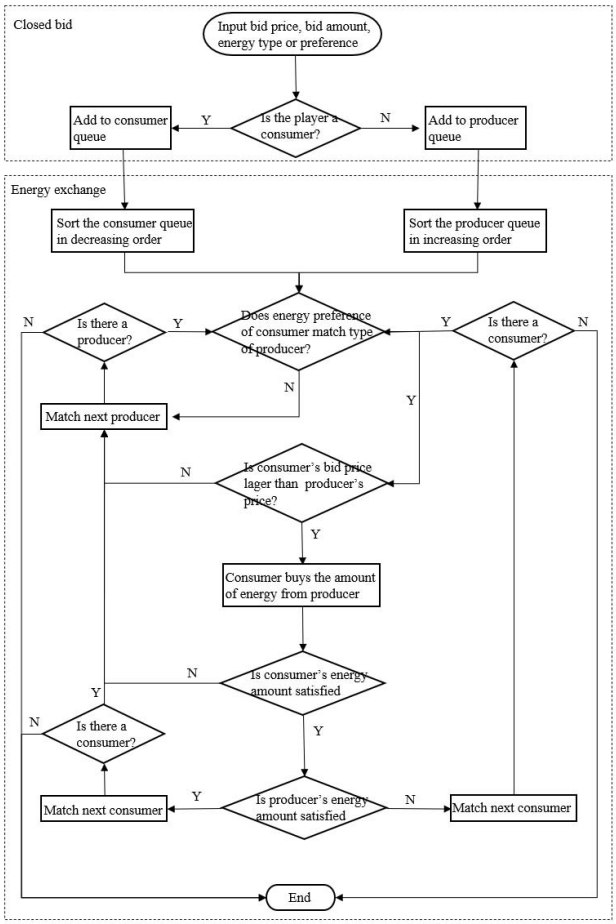


Fig. 1. Process of double auction in energy trading.

method based on the blockchain and double auction mechanism. Zero-knowledge proof is employed to hide the amount of energy demand and energy production in P2P energy trading. Other participants are able to verify the correctness of the bid amount of double auction by a smart contract of zero-knowledge proof in the blockchain.

The remaining of this paper is organized as follows. Section II presents the preliminary knowledge about energy trading and zero-knowledge proof. Our system model is summarized in Section III. Section IV shows the experimental results of the design. Finally, Section V concludes this paper.

II. Preliminary

This section introduces the double auction mechanism in energy trading, and the concepts of zero-knowledge proof and Pedersen commitment.

A. Double Auction in Energy Trading

Producers sell energy to many consumers; moreover, consumers can buy energy from many producers in the energy market. A double auction is a suitable method for the energy market involving multiple sellers and buyers, where all participants are decision-makers and participate in the auction. Here, a two-stage double auction mechanism for P2P energy trading is presented in Fig. 1 and described as follows.

1) Closed bidding stage: Each producer P_i and consumer C_j upload the bid amount b_a, bid price, and energy type e_t for producer or energy preference e_p for consumer. An energy trading auctioneer collects this information and constructs a producer queue Q_p by each producer P_i and a consumer queue Q_c by each consumer C_j.

2) Energy exchange stage: The producer queue Q_p and the consumer queue Q_c are sorted in increasing order and decreasing order, respectively. So, the first producer P_1 has the highest bid price B_p and the bid amount V_p in the Q_p, and the first consumer C_1 has the lowest bid price B_c and the bid amount V_c in Q_c. The auctioneer firstly matches the energy preference of the consumer and the type of producer. If $B_c > B_p$, the C_j buys the actual bid amount V_{act} from P_i at the clearing price B_{act}.

$$V_{act} = min(V_c, V_p)$$

$$B_{act} = (B_c + B_p)/2$$

If the bid amount is satisfied, the P_i or C_j is removed from the corresponding queue; otherwise, P_i or C_j repeats the energy exchange stage.

B. Zero-Knowledge Proof

Zero-knowledge proof [9] is a mechanism in which a prover uses an encrypted commitment scheme to prove the correctness of commitment to a verifier without revealing any information except for the correctness. Zero-knowledge proof should guarantee three properties as follows.

- Completeness: The verifier always accepts the prover's proof if the prover shows that the proof is true.
- Soundness: The verifier always rejects the prover's proof if the prover's proof is false.
- Zero-knowledge: The verifier learns nothing and only knows that the prover's proof is true if the prover can show the prover is true.

In this framework, a non-interactive Zero-knowledge proof method called Zero-Knowledge Succinct Non-Interactive Argument of Knowledge (zkSNARKs) [10] is employed. ZkSNARKs consists of three main steps including $setup$, $Proof$, and $Verify$. Each step is described as follows.

1) $(pk, vk) \leftarrow Setup(C, 1^\lambda)$: Given a security parameter λ and a program C, a pair of keys (pk, vk) is generated, where pk is a proofing key and vk is a verification key correspondingly.
2) $\pi \leftarrow Proof(pk, x, a)$: Given the proofing key pk, a public input x of program C, and a private auxiliary input a of program C, a knowledge proof π is generated.
3) $b \leftarrow Verify(vk, x, \pi)$: Given the verification key vk, the generated proof π, and the public input x of program C, the proof result is generated. The proof is true if $b = 1$; otherwise, the proof is false.

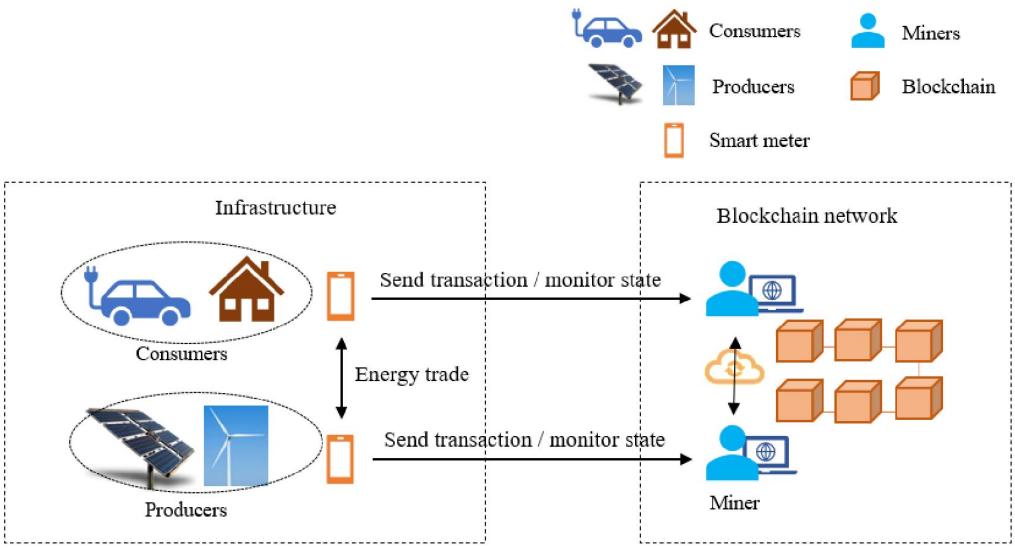


Fig. 2. System overview.

C. Pedersen Commitment

Pedersen commitment [11] is a homomorphic commitment protocol that satisfies perfect hiding and computational binding. In our framework, Pedersen commitment based on ECC is employed. The Commitment $C(m, r)$ is generated by the following equation.

$$C(m, r) = m \times G + r \times H$$

where G and H are the points of order of a larger prime; m and r are a secret message and a random number.

Homomorphism of Pedersen commitment means that if $C_1(m_1, r_1)$ and $C_2(m_2, r_2)$ are generated by the following equation respectively, $C_3(m_3, r3)$ is equal $C_1 + C_2$ where $r_3 = r_1 + r_2$ and $m_3 = m_1 + m_2$.

$$C_1(m_1, r_1) = m_1 \times G + r_1 \times H$$

$$C_2(m_2, r_2) = m_2 \times G + r_2 \times H$$

The main purpose of the commitment scheme is to provide an information swapping scheme for both parties to avoid information leakage by hiding key values. Pedersen commitment can be used to confirm that both parties have the same value by not revealing the relevant information.

III. System Model

The section illustrates our system overview and the design of the smart contract.

A. System Overview

The system overview is shown in Fig. 2. It includes two modules: infrastructure and blockchain network.

There are two types of entities in the infrastructure module including producers and consumers. Producers are equipped with solar panels or wind cells to produce their own energy. Consumers, such as smart cars or homes, need to expend energy. Moreover, each participant equips a smart meter with high onboard computing resources and networking facilities to communicate with other smart meters. Moreover, the smart meter is also responsible to publish information to the blockchain, to monitor the state of the smart contract on the blockchain, and to send or receive energy based on the state of the smart contract if the submitted trading is done. For simplicity, we ignore the implementation of smart meters and energy storage mechanisms.

The blockchain network is composed of miners nodes and users nodes. Users nodes, including consumers and producers, send the bid information to the blockchain network. Miners nodes are responsible for the verification of each transaction from the users node, such as transaction signification, information format, the state of the smart contract, etc. As mentioned in Section II-A, energy trading employs a double auction mechanism including a closed biding stage and energy exchange stage. The smart contract is designed to execute the double auction mechanism automatically on the blockchain instead of the auctioneer. The bid information and results of the double auction are stored on the blockchain if the verification is successful. There are many blockchain platforms that can be employed as the underlying technologies, such as Ethereum and Hyperledger.

B. Design of Smart Contract

The process of double auction is realized by the smart contract. In the auction scenario, there are three participants: the producer, the consumer, and the smart contract. Smart contracts provide the functionality to support the above stages of action. The producer or consumer as an auctioneer initializes the state of the smart contract and publishes it to the

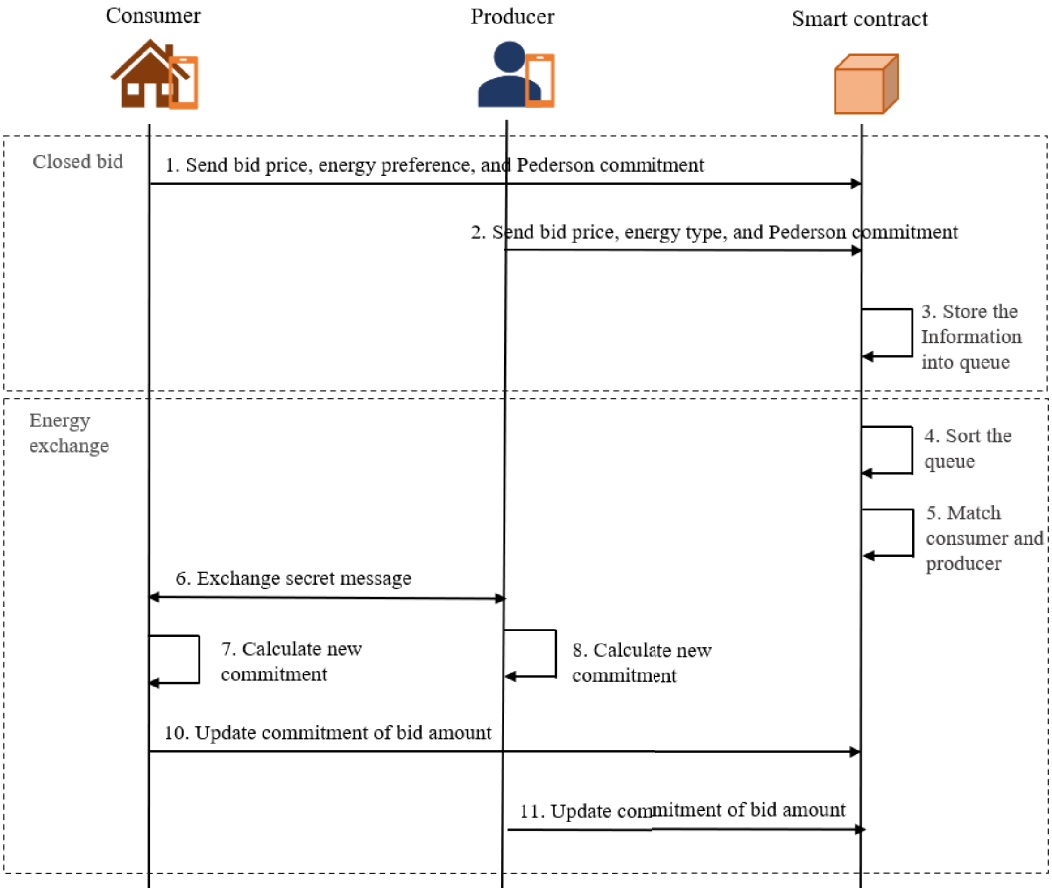


Fig. 3. Process of our smart contract.

blockchain for a double auction of energy trading. Fig. 3 shows the process of our smart contract.

1) Closed bidding stage: Before the end time of the closed bidding stage, all energy producers or consumers are allowed to participate in energy trading. They first generate the Pedersen commitment $H(p, r)$ using the bid amount a and a random number r. The producers or consumers pack their energy bid price, energy preference or type, and Pedersen commitment of bid amount into a transaction and then sent the transaction to the blockchain. If the transaction is verified, the information is stored in a map $Users[msg.sender]$ where $msg.sender$ is the sender's address. If the sender is the producer, the information is saved in queue Q_p; otherwise, the information is saved in queue Q_c.

Algorithm 1 Closed bidding stage

Input: bid price, energy preference or type, and Pedersen commitment of bid amount
Output: Queue Q_c and Q_p

1: $Users[msg.sender] \leftarrow$ bid price, energy preference or type, and Pedersen commitment of bid amount
2: **if** $Users[msg.sender]$ is producer **then**
3: $\quad Q_p \leftarrow Users[msg.sender]$
4: **else**
5: $\quad Q_C \leftarrow Users[msg.sender]$
6: **end if**
7: Return Q_c and Q_p

2) Energy exchange stage: The auctioneer firstly sends a transaction to execute the bubble sorting algorithm of the smart contract. The producer queue Q_p and the consumer queue Q_c are sorted in increasing order and decreasing order in the smart contract, respectively. So, the first producer P_1 has the highest bid price in the Q_p, and the first consumer C_1 has the lowest bid price in Q_c. Then, the auctioneer sends a transaction to execute the match algorithm. If C_1's energy preference and P_1's energy type are matched and the consumer's bid price is greater than the producer's bid price, they exchange energy with each other, and the Pay function is executed automatically if the consumer's balance is sufficient; otherwise, the payment is canceled. In addition, after each energy exchange, the user with the surplus bid amount needs to update a new commitment of bid amount and a new random number. Then, producers and consumers continue matching

until the transaction ends.

Algorithm 2 Energy exchange stage

Input: Queue Q_c and Q_p

for Each $C_i \in Q_c$ **do**
  for Each $P_j \in Q_p$ **do**
    if $C_i[energy_preference] == P_j[energy_type]$ **then**
      if $C_i[bid_price] > P_j[bid_price]$ **then**
        $Pay(C_i[bid_price], P_j[bid_price])$
        $UpdateCommit(H(x))$
      end if
    end if
  end for
end for

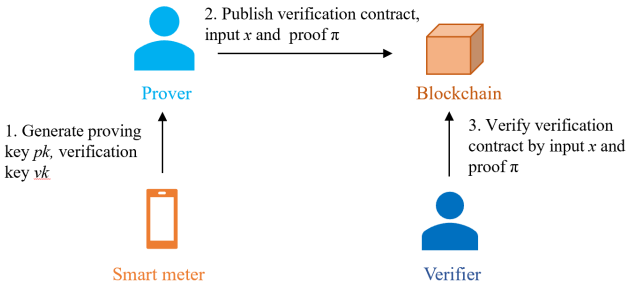


Fig. 4. Verification of zero-knowledge proof.

C. Design of Zero-Knowledge Proof

Pedersen commitment based on ECC is employed to hide the bid amount in the energy trading. All participants can verify the correctness of the commitment by zkSNARKs in the blockchain.

The consumers and producers download ECC parameters from the smart contract including two based points H and G of ECC. They calculate the commitment $H(V, r)$ by the following equation.

$$H(V, r) = V \times G + r \times H$$

In the closed bidding stage, all users submit their commitment with a hidden bid amount into the smart contract. After matching successfully, the matched consumer and the matched producer exchange secret message V_c, V_p, and r_c, r_p by a trusted channel. The producer and the consumer calculate the commitment by the received secret message respectively, and then compare it with the commitment of blockchain. If two commitments are same, actual bid amount V_{act} is calculated by $V_{act} = min(V_c, V_p)$. The user with the rest bid amount updates the new commitment of bid amount V' and a new random number r', where $V' = V - V_{act}$ and $r' = r - r_{received}$. The new commitment is calculated by the following equation.

$$H(V', r') = V' \times G + r' \times H$$

In order to verify the correctness of actual bid amount, zkSNARK is employed in the framework. Fig. 4 describes the process of commitment verification. The smart meter with the rest bid amount generates a proof key pk and a verification key vk. The proof is calculate by the following equation.

$$\pi \leftarrow Proof(pk, timestamp, (V, V_{act}, V', r, r_{received}, r'))$$

where the $timestamp$ is public input; the bid amount V, V_{act}, V' and the random numbers $r, r_{received}, r'$ are private input.

A smart contract of zero-knowledge proof of commitment is generated and transmitted to the blockchain. The proof π and public input $timestamp$ with the commitment are sent to the smart contract. The user's state in the smart contract is updated if the transaction is confirmed. Other participants have the ability to verify the correctness of zero-knowledge proof by the following equation.

$$true/false \leftarrow Verify(vk, timestamp, \pi)$$

Algorithm 3 Verification of commitment

Input: the bid amount V, V_{act}, V', and the random numbers $r, r_{received}, r'$, timestamp.
Output: true/false

$H_1 = (V \times G + r \times H) - (V_{act} \times G + r_{received} \times H)$
$H_2 = V' \times G + r' \times H$
if $H_1 == H_2$ **then**
  Return true
else
  Return false
end if

IV. Experiment

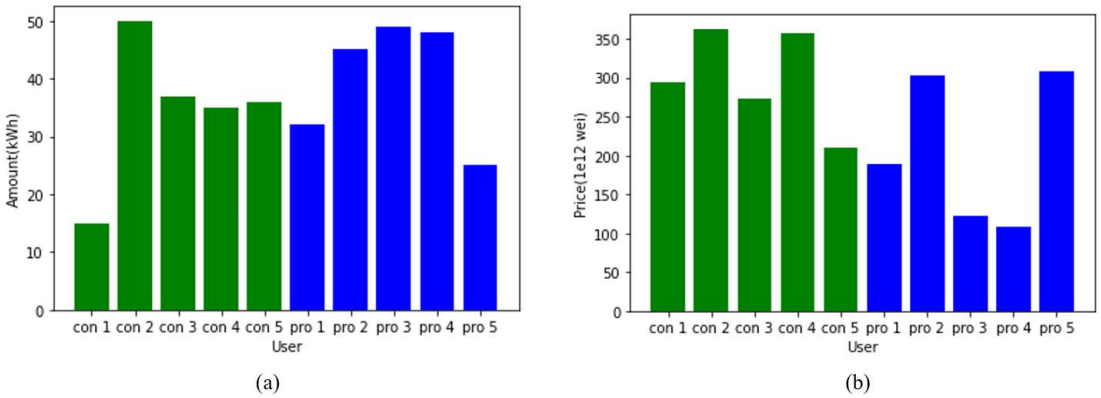
A. Experimental Setup

In the experimental phase, we collect 10 participants' energy demand and energy produced in one day, including 5 consumers and 5 producers. Fig. 5a shows the amount of energy demand and energy produced by the consumers and the producers, respectively. Fig. 5b shows the bid price of the consumers and the producers. Each user's smart meter is considered as a blockchain node. We use *Solidity* language to realize the smart contract and utilize ZoKrates to realize zkSNARKs. The smart contract is deployed in a private Ethereum.

B. Energy Market Analysis

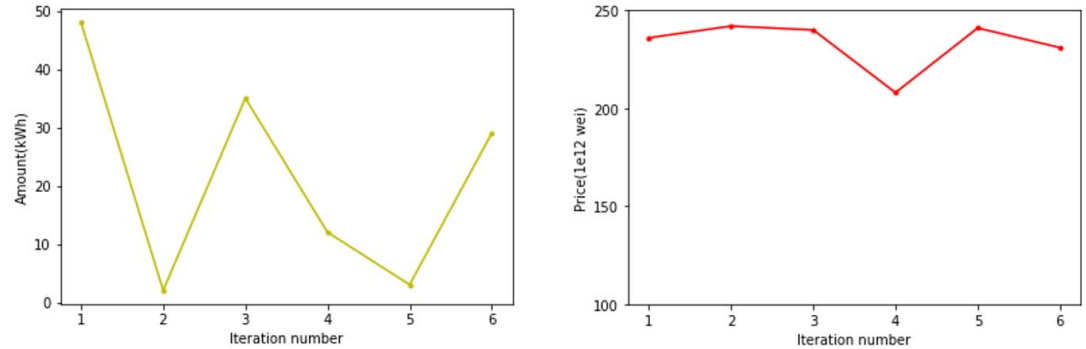
In our simulation, we assume all users have the same energy preference or energy type. By using the double auction mechanism, the smart contract first sorts the consumer array and producer array based on the bid price in decreasing order and increasing order, respectively. Secondly, the smart contract matches the energy type and bid price of consumer and producer by Algorithm 2.

Fig. 6a shows the actual bid amount in each energy match. It is noted that the first match sells all energy amounts of



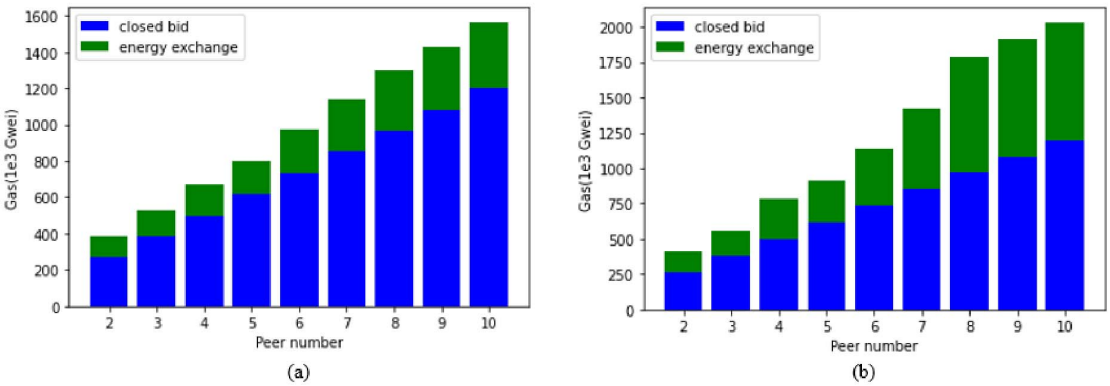
(a) (b)

Fig. 5. Individual bid amount and bid price in one day.



(a) (b)

Fig. 6. Amount and clearing price of trading in each match.



(a) (b)

Fig. 7. Gas consumption of double auction.

producer 2 but consumer 2 has energy deficiency. Consumer 2 supplies the energy deficiency in the second match. The following match has a similar rule-based on Section II-A. Fig. 6b illustrates the clearing price in each energy match based on Section II-A.

C. Blockchain Performance Analysis

Fig. 7 shows the change between gas consumption and the number of users. According to the rule of adding a producer first, the number of users gradually increases. Fig. 7a illustrates the gas consumption of the original double auction. It can be observed that the gas consumption of the closed bidding and energy exchange stage increases with the growth of the number of users, because each user uploads the bid information to the blockchain. In a special situation, the gas consumption of energy exchange is similar. For instance, producer 5 has the highest bid price among all producers, so the smart contract ignores the producer in match and the producer cannot sell any energy in this auction.

Fig. 7b illustrates the gas consumption of our privacy-preserving double auction. Similar to the original double auction, the gas consumption of closed bidding increases with the growth of the number of users, and the gas consumption of energy exchange increases slowly. Moreover, Fig. 8 describes the number of calling smart contracts by an auctioneer in the energy exchange stage. It is found that the gas consumption of energy exchange is similar in Fig. 7b if the number of calling smart contracts is the same in Fig. 8, since the new user does not affect the result and the number of match, for example, the new producer has a higher bid price or the new consumer has a lower bid price.

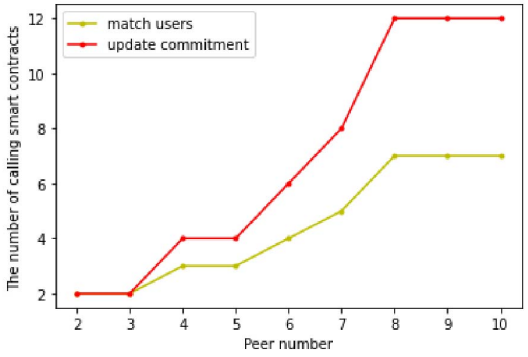


Fig. 8. The number of calling smart contracts in energy exchange stage.

Fig. 9 shows the gas consumption of verification of zero-knowledge proof. In each match, an updated commitment with a hidden energy amount is uploaded to the blockchain, and a smart contract of zero-knowledge proof is published. Other participants are able to verify the commitment with hidden energy amount by the smart contract in each match.

V. Conclusion

In this paper, a privacy-preserving energy trading mechanism based on blockchain and the double auction is proposed.

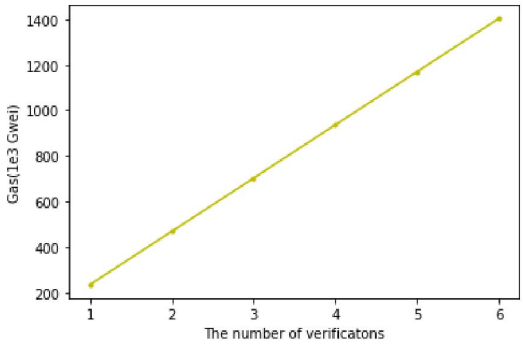


Fig. 9. Gas consumption of commitment verification.

The Pedersen commitment instead of the original energy bid amount is uploaded to the blockchain. All participants can verify the correctness of the actual bid amount by zkSNARKs in the blockchain. The feasibility and efficiency of the proposed mechanism are demonstrated through an experimental evaluation in Ethereum.

References

[1] J. Wang, Q. Wang, N. Zhou, and Y. Chi, "A novel electricity transaction mode of microgrids based on blockchain and continuous double auction," *Energies*, vol. 10, no. 12, p. 1971, 2017.

[2] D. Han, C. Zhang, J. Ping, and Z. Yan, "Smart contract architecture for decentralized energy trading and management based on blockchains," *Energy*, vol. 199, p. 117417, 2020.

[3] Z. Wang, X. Yu, Y. Mu, and H. Jia, "A distributed peer-to-peer energy transaction method for diversified prosumers in urban community microgrid system," *Applied Energy*, vol. 260, p. 114327, 2020.

[4] H. T. Doan, J. Cho, and D. Kim, "Peer-to-peer energy trading in smart grid through blockchain: A double auction-based game theoretic approach," *Ieee Access*, vol. 9, pp. 49 206–49 218, 2021.

[5] K. Gai, Y. Wu, L. Zhu, M. Qiu, and M. Shen, "Privacy-preserving energy trading using consortium blockchain in smart grid," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 6, pp. 3548–3558, 2019.

[6] S. Zhang, M. Pu, B. Wang, and B. Dong, "A privacy protection scheme of microgrid direct electricity transaction based on consortium blockchain and continuous double auction," *IEEE access*, vol. 7, pp. 151 746–151 753, 2019.

[7] L. Liu, M. Du, and X. Ma, "Blockchain-based fair and secure electronic double auction protocol," *IEEE Intelligent Systems*, vol. 35, no. 3, pp. 31–40, 2020.

[8] C. D. Pop, M. Antal, T. Cioara, I. Anghel, and I. Salomie, "Blockchain and demand response: Zero-knowledge proofs for energy transactions privacy," *Sensors*, vol. 20, no. 19, p. 5678, 2020.

[9] S. Goldwasser, S. Micali, and C. Rackoff, "The knowledge complexity of interactive proof systems," *SIAM Journal on computing*, vol. 18, no. 1, pp. 186–208, 1989.

[10] N. Bitansky, R. Canetti, A. Chiesa, and E. Tromer, "From extractable collision resistance to succinct non-interactive arguments of knowledge, and back again," in *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, 2012, pp. 326–349.

[11] T. P. Pedersen, "Non-interactive and information-theoretic secure verifiable secret sharing," in *Annual international cryptology conference*. Springer, 1991, pp. 129–140.