frontiers | Frontiers in Energy Research

TYPE Original Research
PUBLISHED 23 June 2023
DOI 10.3389/fenrg.2023.1183117

OPEN ACCESS

EDITED BY
Xiaoyu Cao,
Xi'an Jiaotong University, China

REVIEWED BY
Chunyu Chen,
China University of Mining and Technology, China
Shaoyan Li,
North China Electric Power University, China

*CORRESPONDENCE
Wang Guoshi,
yizawen9794256259@163.com

RECEIVED 09 March 2023
ACCEPTED 09 June 2023
PUBLISHED 23 June 2023

CITATION
Guoshi W, Hairong Z, Ying L, Qing Y, Dazhi Z, Xiuli L, Yan W and Wei G (2023), A novel management scheme for power grid demand response based on blockchain.
Front. Energy Res. 11:1183117.
doi: 10.3389/fenrg.2023.1183117

A novel management scheme for power grid demand response based on blockchain

Wang Guoshi[1]*, Zhang Hairong[1], Liu Ying[2], Yan Qing[1], Zhu Dazhi[1], Li Xiuli[1], Wang Yan[1] and Guo Wei[1]

[1]Information and Communication Branch of Hainan Power Grid Co., Ltd., Haikou, China, [2]Hainan Power Grid Co., Ltd., Haikou, China

The smart grid optimizes traditional power grids and provides more intelligent services for end users and utilities. However, due to the large number of unsafe devices in the smart grid, there are corresponding risks when implementing relevant solutions on these devices. Therefore, in this article, a security demand response management scheme based on blockchain is proposed to safely make energy trading decisions, thereby realizing the management of the overall load of users in the power grid. In this scheme, the miner node is the verifier. These nodes are responsible for verifying energy transactions in the smart grid (SG) and adding corresponding blocks to the blockchain. Successful energy transactions only occur in blocks in the blockchain. Here, the proposed method is validated through experiments on electric vehicles. The experimental results demonstrate the effectiveness of the proposed scheme in demand response management.

KEYWORDS

smart grid, blockchain, demand response management, network security, efficient

1 Introduction

With the rapid development and application of Distributed Energy Resources (DER), such as Rooftop Photovoltaics (PV) and Electric Vehicles (EVs), there is a new demand for coordinated management schemes for these distributed energy resources. Optimizing the management of distributed energy can enhance the flexibility of end users, reduce power costs, and provide key services to grid operators, achieving a match between demand and local renewable energy supply (Maharjan et al., 2017; Khan et al., 2018; Jindal et al., 2020; Zhang et al., 2021).

The development of the smart grid (SG) has led to incredible growth in the use of smart grid information and its related communication technology (ICT), and end users can obtain various related services within a specified time and range (Maharjan et al., 2017; Khan et al., 2018; Zhang et al., 2021). Smart energy trading, as one of the core components of smart grids, is closely related to users' daily lives, involving various entities such as energy consumers (smart devices, SD) and service providers (power grids), and is achieved through daily energy operations. In addition, the popularity of EVs has also increased the load on the power grid as they require charging (Jindal et al., 2020). These entities form an energy network in smart cities, and it is necessary to optimize the management of energy resources to sustainably maintain energy supplies in smart cities. Therefore, energy management in smart cities has become an important task. However, due to limited power generation resources, it is necessary to manage the overall load situation of various departments in smart cities through

energy trading (Wang et al., 2018; Li Q. et al., 2020; Gao et al., 2021). This can be achieved by using ICT and a cloud-based communication backend to manage the energy demand response of end users in residential, commercial, industrial, and transportation sectors (Fazio et al., 2016; Khoshkbarforoushha et al., 2017). Managing users' energy needs in this way ensures that their energy utilization rate meets their needs. For example, a household that has excess energy at a certain moment can trade with EVs that require energy. Likewise, when EVs have excess energy, they can trade energy with industries that require more energy to operate.

Due to various energy management vulnerabilities in urban communication networks, energy trading can, however, be vulnerable to attacks. Therefore, to address these attacks and provide security and privacy for entities participating in energy transactions, a secure energy management scheme is needed that can ensure the overall security and privacy of users, even if the network is at risk (Cheng et al., 2022).

Blockchain technology ensures security constraints in a decentralized manner through distributed ledger systems (An et al., 2020; Cheng et al., 2020; Xu, 2020; Cheng et al., 2021), making it very difficult to disrupt this system as the disruption requires the destruction of all nodes (i.e., miner nodes), which are responsible for maintaining the security of the entire system. Therefore, blockchain can be successfully applied as part of security measures in smart cities to ensure the security of energy transaction requests. To date, some researchers have conducted relevant research. For example, Puthal (Puthal et al., 2018) proposed a blockchain-based energy trading solution that utilizes energy coins to achieve secure energy trading in intelligent transportation systems (ITS). Aujla (Aujla et al., 2018) proposed a localized peer-to-peer (P2P) electricity trading model for local electricity trading between plug-in hybrid vehicles (PHEVs) in smart grids. Wilson (2013) proposed a blockchain-based edge as a service framework for supporting secure energy trading in V2G environments of Software Defined Networks (SDNs). Kaur et al. (2018) proposed a new EV charging scheme for decentralized blockchain smart grid systems, aiming to minimize the level of power fluctuations in the grid and the overall charging cost for EV users. However, these methods all have some common problems, such as the inability to select miner nodes, the inability to select wallets, and a single application scenario. Therefore, this article proposes a scheme for applying blockchain to energy trading (Smart Secure Grid, SSG) to ensure the integrity and authenticity of energy trading requests. Once the transaction is authenticated, the energy transaction request will be used to manage the demand response, and after each successful transaction, the energy coin will be traded. In summary, the main contributions of this work are three-fold:

1. A miner node selection scheme is proposed based on the power capacity and processing capacity of intelligent devices in the smart grid.
2. A blockchain-based creation and verification scheme is proposed, which adds entries to the blockchain to ensure the security of energy transactions.
3. An energy transaction scheme for demand response management is designed to deal with energy transaction requests from different sources.

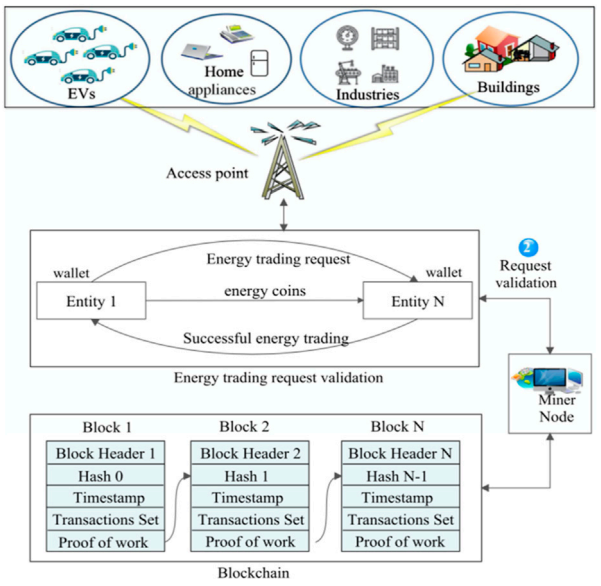

FIGURE 1
Architecture of the proposed method based on market structure.

2 The proposed method

Here, we carefully discuss the general scenario where the SG ecosystem (SGE) is utilized for energy transactions. The proposed system consists of different entities such as residential, industrial, construction, and EVs. To maximize their benefits, their energy resources need to be shared with each other. As shown in Figure 1, these entities connect by using communication infrastructure based on ICT, while information about energy transactions between entities is delivered through access points. The details of an energy transaction among two various entities (seller and buyer) are stated as follows. First, energy-seeking entities initiate energy transaction requests in the SG ecosystem. Next, the system sends the requests to the access point. The verification requests are further sent by these access points to the miner nodes. The validation of energy transaction requests and relevant users' privacy protection is performed by the miner nodes. The entity with superfluous energy will start trading with the energy-seeking entity after the request is validated by the miner node. The buyer transfers its energy coins to the seller at the value agreed by both parties once the energy trading succeeds. Note that in the system proposed in this paper, the initial value of energy coins owned by each participant is consistent; that is, the total amount of energy coins is constant. Figure 1 shows that two types of functional nodes (miner node and ordinary node) exist in our proposed system. Each node will be described in detail next.

2.1 Miner nodes

Miner nodes (MNs) are employed to authenticate, authorize, and audit energy-related transactions in the smart grid ecosystem. Each MN has its own limited storage capacity, where transaction

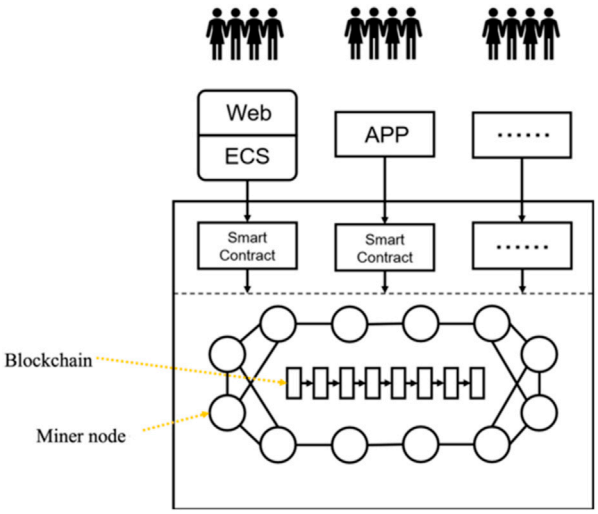

FIGURE 2
Blockchain schematic diagram.

data are temporarily stored before being added to the blockchain. The MN maintains a ledger for storing information related to received data blocks. The data block also includes block headers, hash values, timestamps, and transaction sets. For the block to be added to the blockchain, the MN calculates the proof of work based on the received information. If it matches the hash value received in the block, the MN will authenticate the block. It then sends information about the block to other MNs, and if all miner nodes agree on the authenticity of the block, the block is added to the blockchain. This step is repeated for each block received by an MN until its memory capacity is full, and then it sends the data to a centralized location with a large memory capacity.

The MN is employed to authenticate, authorize, and audit each energy-related trading in the SGE. The transaction data are temporarily stored in each MN's limited storage area before they are added to the blockchain. A special ledger is maintained by the MN for storing information related to the data blocks received in the proposed system. Specifically, the complete data block also commonly includes a discriminative block header, a proof of work (PoW), a transaction set, one timestamp, and one hash value. The MNs calculate a PoW based on the received information for the newly added block. Next, once the calculated PoW matches the hash value received by the data block, the MN will authenticate the aforementioned new data block and send the block's information to the other MNs. The new block will be added to the blockchain if all MNs reach a consensus on the authenticity of the block. These steps are repeated for each block received by a single MN until their memory capacity is exhausted, and then the data are sent to a centralized location that has a large memory capacity.

2.2 Ordinary nodes

In the smart grid ecosystem, nodes that are not miner nodes are referred to as ordinary nodes. The ordinary node maintains a limited-capacity ledger for storing its transaction logs. These ordinary nodes also have a wallet for energy coin payments between each other. The blockchain mechanism provides privacy protection for related entities by using encryption primitives to verify transactions between these modes. The blockchain is regarded as a chain of continuous data blocks that includes nodes' related information. These nodes wish to execute transactions in a safe manner without relying on a centralized credible third-party system. In this system, each node has its own ledger to maintain transaction history. The ledgers of miner nodes are usually larger than the ledgers of ordinary nodes. The MNs are responsible for authenticating and authorizing transactions that users wish to execute in a secure manner.

The blocks in the blockchain contain a set of values used for authenticating transactions. These values include transaction blocks and hash value sets. The block header also includes the hash value of the previous block, which is used for verifying and authorizing transactions. It also contains sender information related to the transaction. The hash value is sent as part of the block to verify the identity of the sender. The ordinary nodes calculate this value and append it to blocks that are sent to miner nodes. The timestamp is the real-time information utilized by the miner node to calculate the PoW. The trading set or content consists of instructions based on

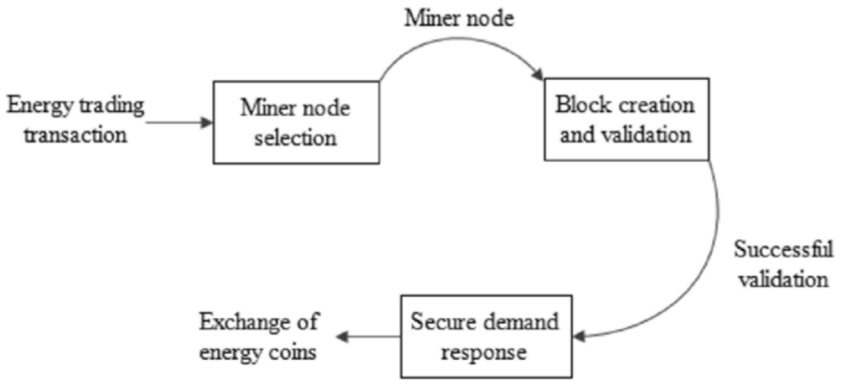

FIGURE 3
Workflow of the SSG method.

which actual energy transactions are carried out. For example, the trading set can include the price of energy trading and the quantity of energy traded.

The PoW is not sent with the transaction, instead, it will be calculated by the miner node and used to add new blocks to the blockchain. This involves solving complex mathematical problems (similar to calculating hash values) and then matching the solution with the hash values received from ordinary nodes. If these two values match, then the block (and the transactions within it) is considered real and processed. In summary, the working steps of the blockchain mechanism are as follows:

1. Entity (such as Entity 1) sends energy transaction requests in the network.
2. The request is passed to the miner node for identity verification. If the calculated hash value matches the received hash value, the miner node adds the transaction to the blockchain.
3. If the request is verified to be true, the miner node sends energy trading requests to other entities. For entities that agree to participate in the transaction (such as entity N), the miner node notifies entity 1 and conducts actual energy transactions. If multiple entities are interested in energy trading, Entity 1 decides which entity to engage in energy trading with.
4. Blockchain is used to transfer energy coins from entity 1 to entity N. Entity 1's wallet is deducted based on the agreed amount. Then, the miner node uses a similar method to verify the wallet address of entity N.

The schematic diagram of the entire blockchain is shown in Figure 2.

3 The principle of the SSG

This section introduces the SSG scheme. First, the miner nodes are chosen to verify blocks created during the process of energy trading. Once the block verification is complete, the secure energy transactions are carried out according to the energy needs of the different entities. Figure 3 shows the complete algorithm process, and we will discuss the relevant stages in the following sections.

3.1 The selection of a miner node

The method of selecting an MN makes all entities in the SSG system miner nodes. Some previous studies have proposed MN selection methods (Chaudhary et al., 2019). For example, smart meters are utilized as miner nodes to access and store energy data for smart homes, or EVs are used as miner nodes to facilitate energy transactions in a smart transportation department, etc. The selection of a miner node depends on the types of available nodes and applications. Therefore, each MN selection method is unique to the problem. However, in this scheme, the miner nodes in our proposed method are chosen from various entities including family residences, industrial facilities, and buildings via analysis of the data associated with the entity's processing capacity or electricity capacity. Algorithm 1 gives the MN selection process in detail.

```
Input: 𝕊, 𝔼, 𝕀, 𝔹
Output: MNs
procedure SELECTION_of_MN (𝕊, 𝔼, 𝕀, 𝔹)
/*Nodes in 𝕊 (𝕊 represents the smart home) */
Set threshold = τ_s
for (s = 1; s ≤ size(𝕊); ++s) do
  Get the power capacity P_c^s in each s
  if P_c^s > τ_s then
 Put s in 𝕃
  end if
end for
/* Nodes in 𝔹 (𝔹 represents the building) */
Set threshold = τ_b
for (b = 1; b ≤ size(𝔹); ++b) do
  Obtain each b's power capacity P_c^b
  if (P_c^b > τ_b) then
 Put b in 𝕃
  end if
end for
/* Nodes in 𝕀 ((𝕀 represents the industry)*/
Set threshold = τ_i
for (i = 1; i ≤ size(𝕀); i++) do
  Sort nodes in descending order based on processing
  power
  Obtain μ_i, the processing power
  j ← i
  while (j > 0 and 𝕀_{j-1} < 𝕀_j) do
 temp ← 𝕀_{j-1}
 𝕀_{j-1} ← 𝕀_j j ← j − 1
  end while
  if (μ_i > τ_i) then
 Put i in list 𝕃
  end if
end for
Select the top 50% nodes of 𝕀 and place them into list 𝕃
/* Nodes in 𝔼 (𝔼 represents EV) */
Set threshold = τ_e
for (e = 1; e ≤ size(𝔼); ++e) do
  Obtain each e's processing power p^e
  Ask e if it is possible to serve and store the protocol
  time in T (t^e e)
  Calculate μ_e = p^e · t^e
  if (μ_e > τ_e) then
 Put e in 𝕃
  end if
end for
Calculate the amount of MNs MN = γ · size(𝕃). Among them,
where γ is the ratio of miner nodes to ordinary nodes.
Randomly select MN from list 𝕃 and set it as a miner node
end procedure
```

Algorithm 1. Miner node selection.

First, the number of buildings, industries, smart homes (SHs), and EVs are taken as input for the algorithm and the output will be the

TABLE 1 Meaning of the symbols.

Symbol	Meaning
ID_E	Identity of entity E
rn	A random number between 0 and 2^{32}
H_E	Hash of entity of E
T_s	Transaction set
W_E	Wallet address of E
H_{Root}	Hash of Merkle tree's root
H_{Prev}	Previous value's hash
BH_E	Block header of E
M_E	Message bit of E
MD_E	Message digest of M_E
PoW_E	Proof of work of E
T_L	Transactions in the blockchain
H_{new}	E's hash calculated at M
H_{result}	H_{new}'s message digest
PoW_M	PoW generated at M
SHA-1	Hash function

selected MN. The algorithm utilizes different standards in order to select MNs from all existing entities. It is necessary to calculate the power capacity of SHs. SHs with power capacity greater than a threshold τ_s are put into a list $\mathbb{L}$ for selecting a possible MN. Likewise, buildings with a power capacity greater than τ_b are put into the created list, i.e., $\mathbb{L}$. Notably, the thresholds of buildings and dwellings should be various (i.e., $\tau_b > \tau_s$) because buildings have more load demands than single dwellings. For industry, it is necessary to consider their processing power when choosing MNs, as it is always smaller than the processing power of a building or an SH. We then order the loads of industrial facilities in descending order according to their processing capacity and add the top 50% into the $\mathbb{L}$.In addition, the processing power and protocol time of EVs is important as they are the only moving entities in the SG ecosystem. The algorithm defines protocol time as the time when EVs agree to act as miner nodes. Therefore, their utility is calculated based on their processing power and protocol time, and EVs whose utility is greater than τ_e are added to the list $\mathbb{L}$.

The ratio of MNs to ordinary nodes (γ, whose value is 20%) is taken into account when selecting MNs from the list $\mathbb{L}$. Another advantage of this algorithm is that even though an entity attempts to maximize its processing capability and power capacity to evolve as an MN, there is no guarantee that it will eventually become one. Therefore, each node will maintain its original nature, and some entities will be selected as MNs from the SHs or EVs by the algorithm so that they can be added to the overall system when verifying user requests. Furthermore, the values of τ_s, τ_b, τ_i, and τ_e change after periodic time intervals to exclude and include other accessible entities in the process of selecting an MN. Then, the system performs the algorithm again and selects a new MN according to the altered thresholds. Notably, the MNs are randomly selected, meaning it is difficult for adversaries to imitate this random selection process. This robust selection process prevents adversaries from locating a miner node and attempting to operate it.

3.2 The creation and validation of a block

After selecting the miner node, a block must be created and verified before it can be added to the blockchain. Therefore, the PoW of an ordinary node (such as entity E) is sent to the miner node. First, the first MN that resolves the PoW bootstraps the verification procedure by sending this PoW to the remaining nodes for review. The leading node adds blocks to the blockchain if the other results are consistent. Conversely, the blocks are abandoned, and the trade is invalid if the results mismatch. Algorithm 2 depicts the process of creating and validating the block. Table 1 explains the meanings of the symbols that appear in Algorithm 2.

Entity: (E)
Input: ID_E, rn
Output: PoW
$H_E = H(ID_E \| rn)$
$T_E = [ID_E || H_E || T_s]$
Calculate Wallet address:
$W_E = H(ID_E \| nonce)$
$H_{Root} = H\{[H(T_{E1}) + H(T_{E2})] +$
$[H(T_{E3}) + H(T_{E4})] + [H(T_{E(n-1)}) + H(T_{En})]\}$
Create Block Header:
$BH_E = [H_{Prev} || H_{Root} || T_s || t_s || rn]$
$M_E = [BH_E \| Pad]$
$M_{D_E} = SHA - 1[M_E]$
$PoW_E = (W_E \| M_{D_E})$
$< T_E, W_E, ts, PoW_E > \rightarrow (SSL/TLS)$
Miner node: (MN)
Input: PoW_E, T_E, W_E, ts
Output: valid/Invalid transaction
$T_L = [T_{E1} || T_{E2} || T_{E3} \| \ldots \| T_{En}]$
$H_{Root} = H\{[H(T_{E1}) + H(T_{E2})]$
$+ [H(T_{E3}) + H(T_{E4})] + [H(T_{E(n-1)}) + H(T_{En})]\}$
Blockchain Validation
$H_{Prev} = extract(T_L)$
$BH_E = [H_{Prev} || H_{Root} || T_s || t_s || rn]$
$H_{new} = [BH_E \| Pad]$
$H_{result} = SHA - 1[H_{new}]$
$PoW_H = [W_E \| H_{result}]$
if $(PoW_H == PoW_E)$
return (valid)
else
return(invalid)
Send results for auditing to other miner nodes.
If consensus is reached, then accept the block
$< valid/invalid > \leftarrow (SSL/TLS)$

Algorithm 2. The process of block creation and validation process between an ordinary node (E) and a miner node (MN).

We explain entity E's complete process of block creation as follows:

(a) The hash value H_E is calculated by using the value of the random number (rn) and E's identity (ID_E). Then, the algorithm appends the computed value to the transaction (T_E) combined with ID_E and the transaction set (T_S).
(b) E's wallet address W_E can be computed by adding a value of nonce and calculating the $ID_E \parallel$ nonce's hash. To enhance the wallet address' complexity, we add a 32-bit nonce, making it difficult for attackers to crack. The transfer of energy coins after an energy transaction succeeds on the basis of this wallet address.
(c) Through merging the right and left sub-hash index pairs, we can compute the root's hash in the Merkle hash tree, which is calculated together with the former T_E's hash value toward all entities (An, 2017). We obtain E's block header by computing this hash value.
(d) The information of T_E, T_S, timestamp (ts), and the previous hash of rn are included in the block header (BH_E). Adding padding bits into the block header can generate fixed-length message bits (M_E).
(e) A 160-bit final hash value is generated by calculating a message digest (MD) MD_E for M_E with SHA-1. We append the computed value to E's wallet *id*, called E's proof of work (PoW_E). PoW_E is then utilized to match the PoW generated by the miner nodes to validate the trade successfully.

The process of verifying blocks (identifying whether they are real or tampered with) is performed by the miner node (MN). To this end, the miner node's inputs are the PoW provided by E and other values given in Algorithm 2. Next, we discuss the miner node's process of validating received blocks as follows.

(a) First, the real trade received is computed by M, and a combined blockchain transaction T_L is generated.
(b) Next, the root node's hash value of the Merkle hash tree is computed by M. We can extract the previous hash function (H_{Prev}) from the blockchain, and compute entity E's block header via utilizing the newly computed values of H_{Prev} and H_{Root}.
(c) Appending padding bits can generate a new fixed-length message. To provide an MD value (160-bit), the SHA-1 algorithm takes these padding bits as input. The provided value is actually a PoW (PoW_H) computed by E at M.
(d) The PoW_H's calculated value is compared with the received PoW_E value, and the blocks are valid once they match. Conversely, it will be abandoned if they do not match.
(e) The other MNs must agree on the block value and update it to put a block into an existing blockchain. To do this, the first miner node that solves the PoW will take the lead and broadcast its results to other miner nodes. If all of them reach an agreement, the algorithm adds the blocks to the blockchain and notifies all miner nodes. Otherwise, the block will be abandoned.

3.3 Management of safe demand response

This section describes how to trade energy from one entity to another. To trade energy, an EV will go to a buyer/seller (for example, an industry, home, or building) and then plug itself into the place to sell or buy energy. An energy transaction between one static entity and a mobile one occurs if it has been verified, as described in the previous section. Therefore, according to the energy demand of static entities, EVs trade energy from one place to another, as described in Aujla et al. (2018). There are two situations in this case. EVs are considered buyers if the static entities have additional energy, and EVs are considered energy providers when static entities need energy. We discuss these two cases in the following.

3.3.1 Electric vehicles act as buyers

EVs act as energy buyers if an entity, such as X, has excess energy. Given the X's energy demand E_x^{dmd}, and its available energy E_x^{avl}, X's excess energy E_x^{exc} can be calculated by Eq. 1:

$$E_x^{exc} = E_x^{avl} - E_x^{dmd}. \tag{1}$$

The corresponding state-of-charge (SoC) level and an EV's rated energy capacity (REC) determine the maximum amount of energy it can purchase. Therefore, given i-*th* EV's maximum SoC level $SoC_i{}^{max}$ and available SoC SoC_i^{avl}, the charged SoC from X is given by Eq. 2:

$$SoC_i^{chr} = SoC_i{}^{max} - SoC_i^{prs}. \tag{2}$$

The excess energy given by X to the i^{th} EV (E_i^{gvn}) is calculated by Eq. 3.

$$E_i^{gvn} = \left(SoC_i{}^{max} - SoC_i^{avl}\right)E_i^{rate}, \tag{3}$$

where E_i^{rate} denotes the i-*th* EV's REC.

Because the EV consumes energy in the process of driving from its position to X's position, this energy will also be supplemented from X. The consumed energy can be calculated by Eq. 4.

$$E_i^{trvl} = \frac{D^{(x\to y)}}{D^{max}}E_i^{rate}, \tag{4}$$

where $D^{(x\to y)}$ denotes the distance (calculated by GPS) from the EV's position to X's position, and D^{max} represents the EV's maximum distance it can travel when its battery has been fully charged. Notably, the EV's battery capacity determines D^{max}, and the EV manufacturer has preset its value. In addition, this type of energy will also be charged from X, so we can update Eqs 3–5 from now on:

$$E_i^{gvn} = \left(SoC_i{}^{max} - SoC_i^{avl}\right)E_i^{rate} + E_i^{trvl}. \tag{5}$$

If entity X sells energy at a price of p_x, then the value of the energy coins ($P(x \leftarrow ev)$) that EV needs to pay to X is calculated by Eq. 6.

$$P(x \leftarrow ev) = E_i^{gvn} \times p_x \tag{6}$$

After the energy is sold to the EV, X's energy is updated by Eq. 7.

$$E_x^{upd} = E_x^{exc} - E_i^{gvn} \tag{7}$$

It is possible that the value of E_i^{gvn} is greater than 0 even if E_x^{upd} is sold to the i-th EV. In this case, X can approach other EVs and trade energy in a similar way until E_x^{upd} becomes 0. Moreover, the EV may not be able to fully charge its battery, and we can update the SoC and related energy values with Eqs 8, 9.

$$\mathrm{SoC}_i^{upd} = \mathrm{SoC}_i^{prs} + \mathrm{SoC}_i^{chr}, \tag{8}$$

$$E_x^{upd} = \mathrm{SoC}_i^{upd} E_i^{rate}, \tag{9}$$

where SoC_i^{upd} denotes the updated SoC level for the i-th EV after successful energy trading.

3.3.2 Electric vehicles act as providers

An EV can act as an energy supplier to provide a static entity X with the required resources. The energy required for X, that is, E_x^{req}, is calculated by Eq. 10.

$$E_x^{req} = E_x^{dmd} - E_i^{avl} \tag{10}$$

Because the required energy comes from EVs,

$$E_x^{req} \leq \sum_{i=1}^{n} E_i^{gvn}, \tag{11}$$

where E_i^{gvn} is the energy given to entity X by the i-*th* EV, and n represents the number of this type of EV.

After E_i^{gvn} is given to entity X, the available energy of the i-*th* EV can be calculated by Eq. 12:

$$E_i^{upd} = E_i^{avl} - E_i^{gvn} - E_i^{trvl}, \tag{12}$$

where E_i^{trvl} is calculated by Eq. 4. Only when the conditions of Eq. 13 are met can the energy of EV be successfully transmitted to X.

$$E_i^{upd} > E_i^{thr}, \tag{13}$$

where E_i^{thr} is the threshold energy that should always be maintained. EVs use E_i^{thr} to commute elsewhere and minimize the losses of battery degradation. The value of this energy is preset by the owner of the EV.

Another condition of a successful trade is that the EV owner's announced price must be agreed to by X before the transaction begins. If the EV charges the price p_{ev}, entity X must pay $P(ev \leftarrow x)$, which can be calculated by Eq. 14:

$$P(ev \leftarrow x) = E_i^{gvn} \times p_{ev}. \tag{14}$$

If the accessible energy is still smaller than X's load demand (LD), X will continue to exchange energy with other vehicles until it satisfies Eq. 11.

As previously described, once an energy transaction occurs, energy coins will be paid by the buyer to the seller (their wallets are updated synchronously) based on the blockchain. Figure 4 provides an example of such a transaction. Specifically, four different entities from various sectors, such as construction, commerce, industry, and residential, hope to perform energy trades with EVs to process their load requirements. In addition, different boxes denote the accessible energy of these entities. The dashed lines denote various entities' LDs, and the solid black lines represent their total available energy. Therefore, as shown in Figure 4, EVs move to various entities, charging or discharging their batteries according to the energy needs of these entities and the energy available in the aforementioned EVs. Additionally, the blockchain transfers energy coins between various entities.

4 Experimental results and analysis

We discuss and analyze the experimental results in the following section.

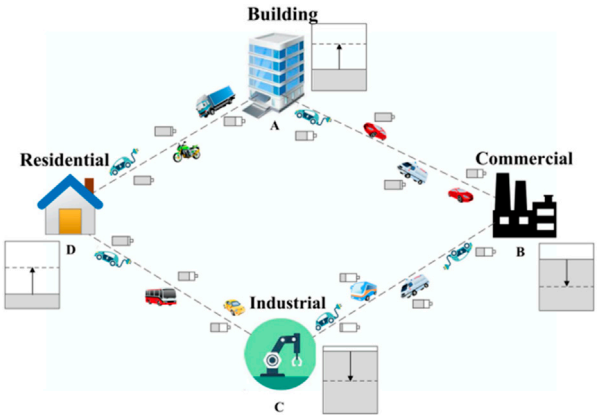

FIGURE 4
Energy transactions between different entities.

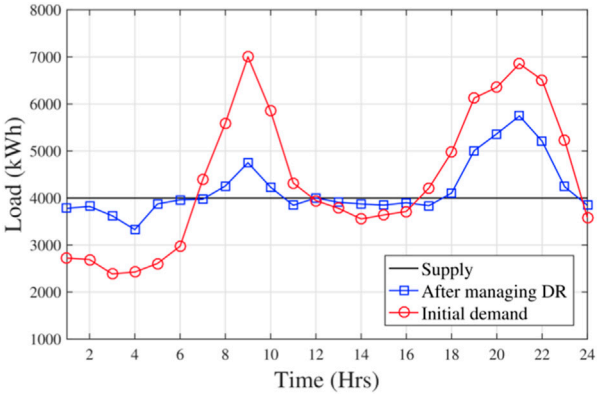

FIGURE 5
Impact of demand response management on total LD.

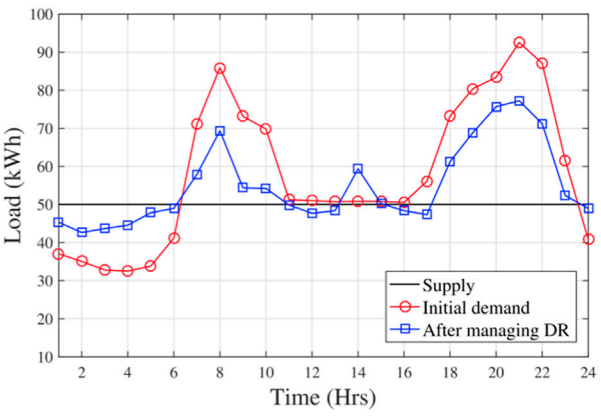

FIGURE 6
Impact of commercial buildings' demand response management.

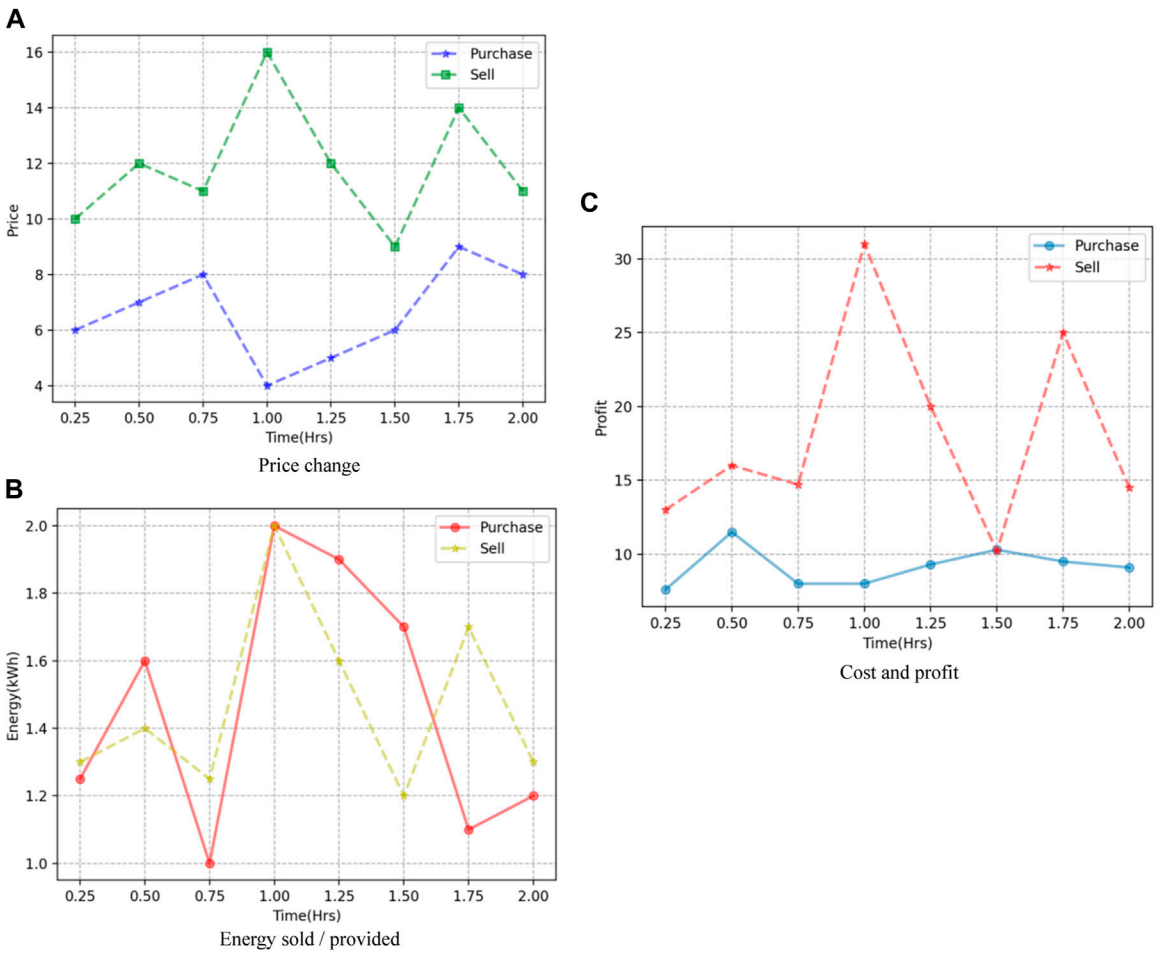

FIGURE 7
Comparison of energy consumers and providers. **(A)** Price change. **(B)** Energy sold/provided. **(C)** Cost and profit.

4.1 The analysis of algorithm complexity

In this section, we evaluate the performance of the proposed method based on its communicational and computational costs.

4.1.1 Communicational cost

Given the two different entities named E and M, we can calculate the communication cost of the transmitted message bits as follows:

(1) On entity €: If the bits of ID_E are selected as 128, the random number, transaction set, and real timestamp are each 32, along with the 160 bits hash function. The bits of BH_E occupy 32*3 + 160*3 = 416 bits, while T_E is 128 + 160 + 32 = 316 bits. Here, adding 96 bits of padding makes it a 512-bit message, and SHA-1 gives it a 160-bit output after calculating MD. The final PoW consists of the aforementioned 160-bit MD and a 160-bit wallet address (including a nonce of 32 bits along with a 128-bit ID). Therefore, PoW_E needs 320 bits. The total cost of T_E, W_E, ts, *and* PoW_E communicating with the miner node is equal to 160 + 316 + 320 + 32 = 828 bits.
(2) On the miner node (MN): This node M can extract H_{Prev} (160 bits), while H_{Root}'s value occupies the same number of bits. In addition, the number of bits for the generated PoW's value occupies 320 bits. The verification operation occupies one bit. Therefore, it takes 320 + 1 = 321 bits to finish the communication from PoW_H to other miner nodes, in addition to transmitting the eventual result to entity E.

4.1.2 Computational cost

When the blocks are created and validated, the involved computational operation mainly includes the add operation, hashing, and appending operation. The computational times for them need 1.0 *ms*, 2.7 *ms*, and 0.3 *ms*, respectively. Therefore, we can calculate the computational cost as follows:

(1) On entity (E): E conducts 10 appends, n/2 add operations and four hash operations. Thus, given the value n = 100, the computational cost is 10 × 0.3 + 50 × 1 + 4 × 2.7 = 63.8 ms.
(2) On the miner node (MN): Here, several operations must be utilized in the process of verification: n/2 add, seven append, and three hash operations. Therefore, the overall computational time to validate a block is 50 × 1 + 4 × 2.7 + 7 × 0.3 = 62.9 *ms*.

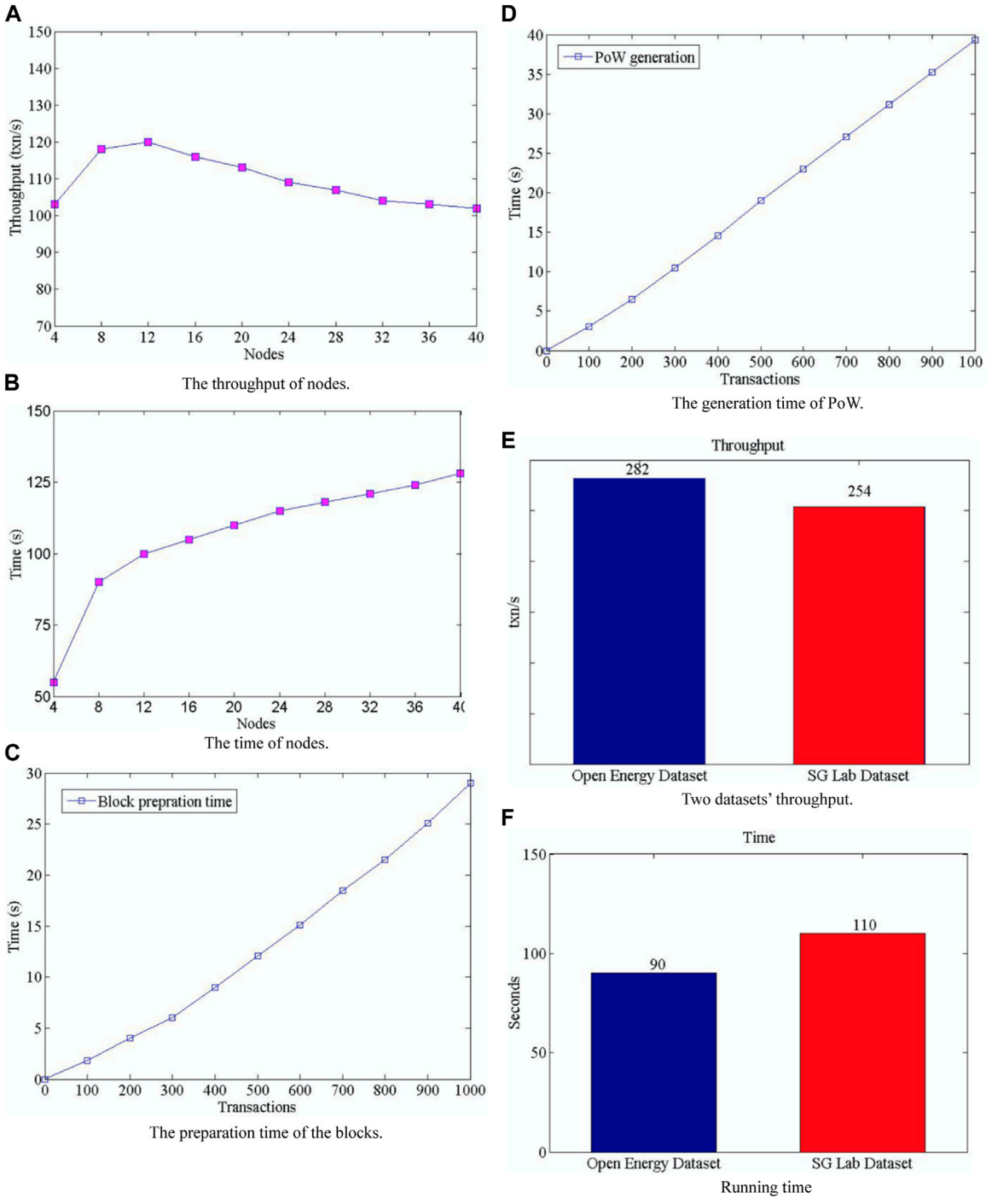

FIGURE 8
Safety assessment. **(A)** Node throughput. **(B)** Node time. **(C)** Block preparation time. **(D)** PoW generation time. **(E)** Throughput of the two datasets. **(F)** Running time.

4.2 The numerical results of the demand response process

Based on entity data from US Open Energy Information (Wilson, 2013), this paper simulates a scenario with 10 industrial buildings, 50 residences, and 30 commercial buildings. Subsequently, we use this scenario to test the proposed method for energy transactions. At the same time, in this scenario, 100 EVs (with energy capacity from 12 kWh to 36 kWh) are added, and their data are randomly assigned to simulate the real situation.

The entities' demand responses need to be managed by the 4 MW fixed energy supply of the SG test scenario. From

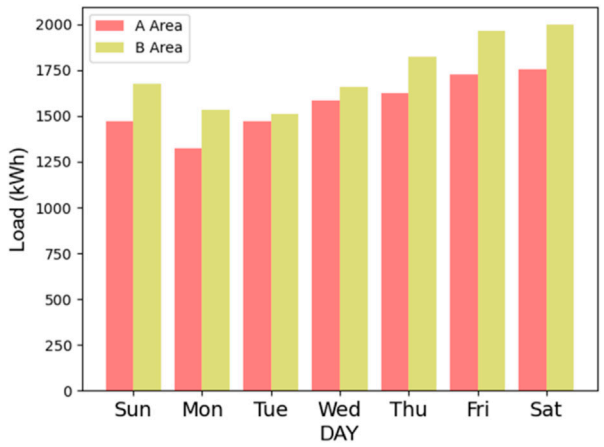

FIGURE 9
Comparison of actual deployment results.

Figure 5, we can observe the total initial LD and the LD after managing the demand response and the supplied energy. We can infer from Figure 5 that the EV largely manages the LD of entities involved in transactions in some time gaps, which can help reduce their dependence on SG. Figure 7 shows an example of a negative LD of a building of commerce that provides 50 kWh of fixed energy. The building buys energy close to the EV from 0000 and 0600 hours and sells its energy to the EV from 0600 to 1100 hours and from 1600 to 2300 hours while this EV owns available energy after managing LD. EVs that need to purchase energy from an entity can be regarded as energy buyers. Figure 6 shows an example of a building for commercial selling its available energy to several EVs between 0800 and 1000 hours.

For an EV, the price charged by the EV is shown in Figure 7A, and the energy sold to the EV is shown in Figure 7B. The cost of an EV purchasing energy from commercial buildings is shown in Figure 7C. Likewise, when the LD management of an entity needs the energy of an EV, the EV serves as an energy supplier. For example, a building may require energy to manage its load from 0400 to 0600 hours in the case shown in Figure 6. The price charged by the EV for selling energy is shown in Figure 7A, and the related energy sold by an EV is shown in Figure 7B. The profitability of EVs is shown in Figure 7C.

4.3 Safety assessment

We evaluate the security mechanism based on blockchain with several various metrics, that is, computational time, throughput, PoW generation time, and block preparation time, against changes in terms of the number of trades and nodes. Furthermore, we validate the security scheme on two datasets [one dataset from the literature (Kaur et al., 2018; Kumar et al., 2019), and the OEN (opening energy information) (Wilson, 2013)]. Specifically, from Figure 8A we can see the change in throughput as the total number of nodes is increasing. First, the quantity of trades every second is trending up and then further down without a major change. From Figure 8B, we can observe the consumed computation time as the number of nodes increases. It can be seen from the figure that after twelve nodes, the sharp increase trend of the initial level slows. The block preparation and proof-of-work generation times are also calculated. Figure 8C depicts the time of block preparation that appears to scale almost linearly with the increase in the number of trades. Interestingly, an analogous tendency of the proof-of-work generation curve with the increasing number of transactions can be seen in Figure 8D. Figures 8E, F show the higher throughput and lower computation time of the OEN dataset compared with that of the SG Lab dataset. The reason for this phenomenon lies in that compared with the data randomly generated in literature (Kaur et al., 2018; Kumar et al., 2019), the relevant datasets are structured.

4.4 Safety assessment

To better evaluate performance, we selected two approximate electricity consumption areas in Hangzhou, Zhejiang Province, China for comparison. The two areas consist of three commercial complexes and four residential areas with a total of 68 residential buildings. The method described in this article is deployed in Area A, while the traditional method is used as the control group in Area B. Each area has more than 120 EVs. We calculated a duration of 1 week, and the results are outlined below.

Figure 9 shows that compared to the traditional scheme, the method proposed in this article can reduce energy consumption by up to 18% and has good performance. At the same time, combined with energy sales, the method proposed in this paper has good economic benefits.

5 Conclusion

This paper proposes a demand response management method for smart grid ecosystem energy transaction security based on blockchain. The entities responsible for verifying energy trading are selected by the proposed method in the energy market. We can draw the following conclusions: 1) this scheme is capable of creating and validating blockchain blocks. If the validity of the data block is verified, energy transactions will be carried out. 2) The advantage of this method lies in the fact that even though a competitor participates in an energy transaction, the transaction cannot be tampered with because the transaction is added to the chain when all miner nodes have verified the transaction. 3) The experimental results show that the total communication and computational burden of this method are not high and that this method can effectively process the demand response of SGE.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material; further inquiries can be directed to the corresponding author.

Author contributions

WG, ZH, and LY contributed to the conception and design of the study. WG, LY, and ZD organized the database. YQ, LX, and GW performed the statistical analysis. ZH, WY, and YQ wrote the first draft of the manuscript. WY and LX wrote sections of the manuscript. All authors contributed to the article and approved the submitted version.

Funding

This research was funded by the National Natural Science Foundation of China (grant numbers 61862021 and 61662019).

Conflict of interest

Authors WG, ZH, YQ, ZD, LX, WY, and GW were employed by the Information and Communication Branch of Hainan Power Grid Co., Ltd. Author LY was employed by Hainan Power Grid Co., Ltd.

Publisher's note

References

An, D., Yang, Q., Li, D., Yu, W., Zhao, W., and Yan, C. B. (2020). Where Am I parking: Incentive online parking-space sharing mechanism with privacy protection. *IEEE Trans. Automation Sci. Eng.* 19 (1), 143–162. doi:10.1109/tase.2020.3024835

An, Q. (2017). *Research and applications on the key techniques of decentralized transaction based on blockchain*. Shanghai, China: Donghua University.

Aujla, G., Jindal, A., and Kumar, N. (2018). EVaaS: Electric vehicle-as-a-service for energy trading in SDN-enabled smart transportation system. *Comput. Netw.* 143 (10), 247–262. doi:10.1016/j.comnet.2018.07.008

Chaudhary, R., Jindal, A., Aujla, G., Aggarwal, S., Kumar, N., and Choo, K. K. R. (2019). Best: Blockchain-based secure energy trading in SDN-enabled intelligent transportation system. *Comput. Secur.* 85 (8), 288–299. doi:10.1016/j.cose.2019.05.006

Cheng, L., Chen, Y., and Liu, G. (2022). 2PnS-EG: A general two-population n-strategy evolutionary game for strategic long-term bidding in a deregulated market under different market clearing mechanisms. *Int. J. Electr. Power & Energy Syst.* 142, 108182. doi:10.1016/j.ijepes.2022.108182

Cheng, L., Liu, G., Huang, H., Wang, X., Chen, Y., Zhang, J., et al. (2020). Equilibrium analysis of general N-population multi-strategy games for generation-side long-term bidding: An evolutionary game perspective. *J. Clean. Prod.* 276, 124123. doi:10.1016/j.jclepro.2020.124123

Cheng, L., Yin, L., Wang, J., Shen, T., Chen, Y., Liu, G., et al. (2021). Behavioral decision-making in power demand-side response management: A multi-population evolutionary game dynamics perspective. *Int. J. Electr. Power & Energy Syst.* 129, 106743. doi:10.1016/j.ijepes.2020.106743

Fazio, M., Celesti, A., Ranjan, R., Liu, C., Chen, L., and Villari, M. (2016). Open issues in scheduling microservices in the cloud. *IEEE Cloud Comput.* 3 (5), 81–88. doi:10.1109/mcc.2016.112

Gao, Z., Peng, L., and Li, B. (2021). Research on the collaborative support of emerging new ICTs for power system automation. *Power Syst. Prot. Control* 49 (7), 160–166. doi:10.19783/j.cnki.pspc.200662

Jindal, A., Aujla, G., and Kumar, N. (2019). Survivor: A blockchain based edge-as-a-service framework for secure energy trading in SDN-enabled vehicle-to-grid environment. *Comput. Netw.* 153 (4), 36–48. doi:10.1016/j.comnet.2019.02.002

Jindal, A., Kumar, N., and Singh, M. (2020). Internet of energy-based demand response management scheme for smart homes and PHEVs using SVM. *Future Gener. Comput. Syst.* 108, 1058–1068. doi:10.1016/j.future.2018.04.003

Kang, J., Rong, Y., Huang, X., Maharjan, S., Zhang, Y., and Hossain, E. (2017). Enabling localized peer-to-peer electricity trading among plug-in hybrid electric vehicles using consortium blockchains. *IEEE Trans. Industrial Inf.* 13 (6), 3154–3164. doi:10.1109/tii.2017.2709784

Kaur, N., Aujla, G., Kumar, N., Zomaya, A. Y., Perera, C., and Ranjan, R. (2018). Tensor-based big data management scheme for dimensionality reduction problem in smart grid systems: SDN perspective. *IEEE Trans. Knowl. Data Eng.* 30 (10), 1985–1998. doi:10.1109/tkde.2018.2809747

Khan, O., Shah, M., Din, I., Kim, B. S., Khattak, H. A., and Rodrigues, J. J. P. C. (2018). Leveraging named data networking for fragmented networks in smart metropolitan cities. *IEEE Access* 6, 75899–75911. doi:10.1109/access.2018.2882811

Khoshkbarforoushha, A., Ranjan, R., Gaire, R., Abbasnejad, E., Wang, L., and Zomaya, A. Y. (2017). Distribution based workload modelling of continuous queries in clouds. *IEEE Trans. Emerg. Top. Comput.* 5 (1), 120–133. doi:10.1109/tetc.2016.2597546

Kumar, N., Aujla, G., Das, A., and Conti, M. (2019). ECCAuth: A secure authentication protocol for demand response management in a smart grid system. *IEEE Trans. Industrial Inf.* 15 (12), 6572–6582. doi:10.1109/tii.2019.2922697

Kurt, M., Yilmaz, Y., and Wang, X. (2020). Secure distributed dynamic state estimation in wide-area smart grids. *IEEE Trans. Inf. Forensics Secur.* 15, 800–815. doi:10.1109/tifs.2019.2928207

Li, Q., Wang, L., Zhang, Y., Liu, C. W., Yuan, Y. J., Lin, R., et al. (2020a). Anti-EGFR therapy in metastatic colorectal cancer: Mechanisms and potential regimens of drug resistance. *Power Syst. Prot. Control* 48 (19), 179–191. doi:10.1093/gastro/goaa026

Li, S., Wang, G., and Zhuang, L. (2020b). Reverse real-time model detection method for blockchain smart contract security. *J. Chin. Comput. Syst.* 41 (10), 2030–2035. doi:10.3969/j.issn.1000-1220.2020.10.003

Liu, C., Chai, K., and Zhang, X. (2018). Adaptive blockchain-based electric vehicle participation scheme in smart grid platform. *IEEE Access* 6, 25657–25665. doi:10.1109/ACCESS.2018.2835309

Liu, W., Wang, B., and Cao, Z. (2020). Design of shared charging pile platform of electric vehicle based on blockchain technology. *Comput. Eng. Des.* 41 (09), 2690–2696. doi:10.16208/j.issn1000-7024.2020.09.043

Maharjan, S., Zhang, Y., Gjessing, S., and Tsang, D. H. K. (2017). User-centric demand response management in the smart grid with multiple providers. *IEEE Trans. Emerg. Top. Comput.* 5 (4), 494–505. doi:10.1109/tetc.2014.2335541

Puthal, D., Malik, N., Mohanty, S., Kougianos, E., and Das, G. (2018). Everything you wanted to know about the blockchain: Its promise, components, processes, and problems. *IEEE Consum. Electron. Mag.* 7 (4), 6–14. doi:10.1109/mce.2018.2816299

Wang, K., Li, H., Maharjan, S., Zhang, Y., and Guo, S. (2018). Green energy scheduling for demand side management in the smart grid. *IEEE Trans. Green Commun. Netw.* 2 (2), 596–611. doi:10.1109/tgcn.2018.2797533

Wilson, E. (2013). Open energy information. Available: https://openei.org/datasets/dataset/commercial-and-residential-hourly-load-profiles-for-all-tmy3-locations-in-the-united-states.

Xu, Y. (2020). A review of cyber security risks of power systems: From static to dynamic false data attacks. *Prot. Control Mod. Power Syst.* 5 (1), 8–19. doi:10.1186/s41601-020-00164-w

Zhang, Y., Wang, A., and Zhang, H. (2021). Overview of smart grid development in China. *Power Syst. Prot. Control* 49 (5), 180–187. doi:10.19783/j.cnki.pspc.200573