IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 20, 2021, accepted April 24, 2021, date of publication April 30, 2021, date of current version May 17, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3076746

A Data-Driven Approach for Blockchain-Based Smart Grid System

ZENG ZENG[1], MEIYA DONG[2,3], WEIWEI MIAO[1], MINGMING ZHANG[1], AND HAO TANG[1]
[1]Information and Telecommunication Branch, State Grid Jiangsu Electric Power Company, Nanjing 210024, China
[2]Tsinghua Wuxi Research Institute of Applied Technologies, Wuxi 214072, China
[3]College of Information and Computer, Taiyuan University of Technology, Taiyuan 030024, China

Corresponding author: Weiwei Miao (weiwei_sgcc@163.com)

ABSTRACT The smart grid is emerging as a future paradigm for power networks. While it has many successful applications, peer-to-peer trading in the local energy market (LEM) is still challenging due to the lack of security and trading mechanisms. In this paper, we design a data-driven, secure, and smart solution DS^2 to address this problem. We first propose a five-layer design of LEM based on blockchain. We then model peer-to-peer trading in LEM as a cost minimization problem and derive an efficient *online* solution leveraging matrix factorization and integer linear programming. DS^2 is implemented and evaluated on a private Ethereum blockchain. We show that DS^2 achieves a mean absolute percentage error (MAPE) of 12.8% compared with the *offline optimal* method through extensive simulations on the real-world dataset.

INDEX TERMS Smart grid, LEM, P2P trading, blockchain.

I. INTRODUCTION

With the prosperity of IoT and distributed renewable energy technologies, the smart grid has achieved rapid development in recent years, enabling various applications such as smart metering and peer-to-peer energy trading [1]–[5]. Unlike the traditional electricity grid where customers can only purchase electricity from utility companies, the smart grid allows households to generate, store renewable energy and trade their excess energy with neighbors in a local energy market (LEM). The widespread use of the smart grid can not only improve the utilization efficiency of energy but also help to mitigate global warming by reducing the emission of greenhouse gases [3], [6], [7].

Despite its advantages, the design and implementation of the smart grid system raise new challenges. On the one hand, a smart grid system usually consists of numerous IoT devices such as wireless sensors and smart meters, making it vulnerable to security attacks. For example, smart meter data can be compromised on the fly by malicious users to under-report energy usage, leading to revenue loss for utility companies [8]. On the other hand, since most participants in the LEM are ordinary users with no expert knowledge, peer-to-peer transactions should be executed automatically without user intervention.

The associate editor coordinating the review of this manuscript and approving it for publication was Shih-Wei Lin.

In this paper, we investigate the design and implementation of the smart grid system in LEM. We propose a data-driven, secure, and smart infrastructure DS^2 based on blockchain technology. The main features of DS^2 are:

- **Secure.** DS^2 is built upon a private Ethereum blockchain. Each block in the chain is associated with a unique hash-based fingerprint and the previous block's hash. As a result, to alter a specific block's content, the attacker has to recompute the hash of every subsequent block and compromise 51% of peer nodes to alter stored records, which is computationally infeasible [9].
- **Smart.** DS^2 utilizes the most significant feature of Blockchain 2.0, *i.e.*, smart contract [10] to accomplish automatic trading. Once pre-defined rules in smart contracts are triggered, trading actions will be directly performed without intervention.
- **Data-driven.** Most existing approaches [11], [12] rely on fixed or known power demand, ignoring the fact that households' power consumption varies with both time and space. Different from these studies, DS^2 proposes to capture the spatio-temporal correlations of power demand and leverage the results to facilitate trading.

Specifically, in this study, we first present the layered design of the Blockchain-based Local Energy Market (BLEM). The peer-to-peer (P2P) trading in BLEM is modeled as an online optimization problem. We then propose to adopt matrix factorization techniques to represent the correlation of power demand data accurately. We construct a bipartite graph based on prediction results and transform the original BLEM problem to an integer linear program (ILP). By solving the ILP problem, we can obtain the optimal trading scheme. Smart contracts on the private blockchain are then triggered to execute transactions based on the optimal scheme.

Our main contributions are summarized as follows:

- We propose a five-layer design of BLEM and implement it on a private Ethereum blockchain.
- We model the P2P trading problem in BLEM as an online optimization problem and design efficient algorithms based on matrix factorization and integer linear programming.
- We validate DS^2 using real-world data from 370 users. And the results show that DS^2 achieves a MAPE of 12.8% compared with the offline optimal method.

The rest of this paper is organized as follows. Section II introduces related works. The detailed design of DS^2 is presented in Section III. Section IV presents experiment settings and evaluation results. Finally, we conclude this paper in Section V.

II. RELATED WORK

In recent years, a lot of efforts have been made to improve the security and scalability of the smart grid [13]–[17]. According to the implementation methodology, existing studies can be classified into the following categories:

A. OPTIMIZATION-BASED APPROACHES

The authors in [4] design a linear programming-based optimization model to evaluate end-user benefits of electricity storage in the local energy market. Two distinct market designs are then proposed with the combination of P2P trading and battery storage. The analysis shows that the combined features produce savings of up to 31% for the end-users. [5] presents a mixed-integer linear programming model for rooftop solar photovoltaic (PV) distributed generation with battery storage. It investigates the economic benefits of renewable source participation in P2P energy trading. Through a simulation of 500 households, the results show that savings up to 28% can be achieved by households. The authors in [18] focus on multi-class energy management in which energy is treated as a heterogeneous product based on attributes of its source. The proposed P2P trading platform then coordinates energy trading between prosumers with heterogeneous (i.e., beyond purely financial) preferences. The objective is to minimize costs associated with losses and battery depreciation using distributed convex optimization [19]. Further in [20], a two-stage aggregated battery control scheme is proposed for P2P energy sharing in community Microgrids. The first stage runs constrained non-linear programming to minimize the total energy cost of the community. The second stage conducts a rule-based control to adjust the control set-points according to the real-time measurements of the net load.

B. GAME THEORY-BASED APPROACHES

The authors in [21] propose a game-theoretic model for P2P energy trading. It assumes that buyers can adjust the energy consumption behavior based on the price and quantity of electricity. It then models the seller selection competition among buyers as a non-cooperative game and price competition among sellers as an evolutionary game. Two iterative algorithms are designed to solve the games, respectively. Morstyn *et al.* [22] study a scalable market design for P2P energy trading using bilateral contract networks. They present the utility-maximizing preferences for each type of participant. A scalable process for the agents to select utility-maximizing contract bundles is also designed. [23] investigates credit rating management for energy trading. It models the problem as a multi-leader and multi-follower game and proves the existence of the equilibrium strategy. It also designs a best-response algorithm to make participants of the market achieve the equilibrium iteratively.

C. BLOCKCHAIN-BASED APPROACHES

In [24], Luo *et al.* study a distributed electricity trading system for electricity prosumers in active distribution networks. The system consists of two layers. In the upper layer, the prosumers are modeled as a multi-agent system, and a set of electricity trading negotiation protocols are designed based on a multi-agent coalition. In the lower layer, a blockchain-based electricity transaction settlement system is proposed to enable trustworthy and secure electricity trading in the upper layer. Kang *et al.* [25] focus on localized P2P electricity trading among Plug-in Hybrid Electric Vehicles (PHEVs). A trading system based on a consortium blockchain named PETCON is designed to optimize electricity pricing as well as the amount of traded electricity. [26] also exploits consortium blockchain technology to design a unified P2P trading framework. In order to reduce transaction confirmation delays, it proposes a credit-based payment scheme that supports fast and frequent energy trading. In addition, it also presents an optimal pricing strategy using the Stackelberg game to maximize the utility of credit banks. [27] tackles the problem of secure transaction in decentralized energy trading. The proposed scheme *PriWatt* is not assumed to rely on any trusted third party. It provides transaction security and identity security based on cryptographic techniques, enabling agents to anonymously negotiate energy prices and trade energy ownership using distributed smart contracts. Further blockchain technology applications can be found in [28]–[31].

Either optimization-based or game theory-based approaches can solve the security issues in LEM. Meanwhile, unlike existing blockchain-based studies, we propose a data-driven approach to address the problem.

III. SOLUTION

A. BLOCKCHAIN PRELIMINARIES

As illustrated in Figure 1, blockchain is a distributed database that maintains an ordered list of records linked together through chains of blocks, each storing a group of transactions.

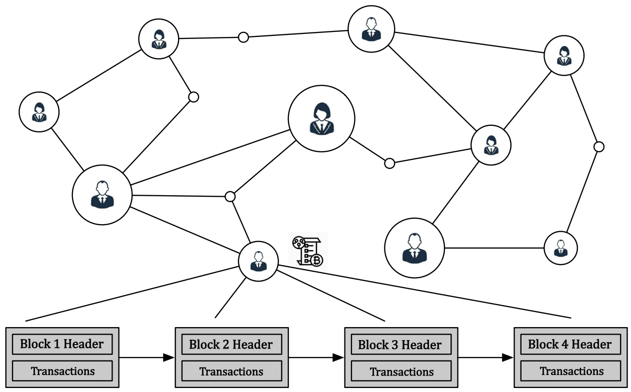


FIGURE 1. The organization of a blockchain.

Transactions are organized in a particular data structure called Merkle Tree [32], whose leaf node is labeled with the hash of a transaction while a non-leaf node is labeled with the hash of the labels of its child nodes. The root hash is further stored in a block's header, where a hash address of the previous block is also computed and stored. Since a slight change in data will lead to a drastic change of a hash fingerprint, attackers will have to alter every block that comes after the one they compromised, which is computationally challenging due to the infeasible-to-invert characteristic of cryptographic hash algorithms.

Each participant of a blockchain network stores a local copy of the whole blockchain (usually called ledger) and communicates with each other through public key infrastructure (PKI). Transaction broadcasted to the network will be stored into blockchain only when consensus over the validity is reached among all peer nodes. As a result, even if attackers could manage to tamper with a single ledger, they will have to take control of more than 50% of peer nodes to affect the whole blockchain. A smart grid structured in this way could gain self-organized resistance from malicious actions.

A blockchain like Ethereum also enables the use of smart contract, which serves as an agreement signed by trading parties in the form of executable programs. When a specific condition according to the contract is met, a smart contract will be triggered and executed automatically, accomplish the pre-defined agreement. For example, a contract linked with the smart meter could alert the customer when the balance falls below a certain threshold and automatically reactivate the power supply when a customer in debt pays the bill.

B. SYSTEM ARCHITECTURE

The proposed system architecture is illustrated in Figure 2. It is composed of five layers from bottom to top:

- **Power grid layer** is the existing power grid infrastructure. This layer is in charge of electric power transmission and distribution, physically connecting all the participants together.
- **Data layer** is responsible for the data storage and chain management as described above. It includes modules like hashing function, Merkle Tree, chain structure, *etc.*.

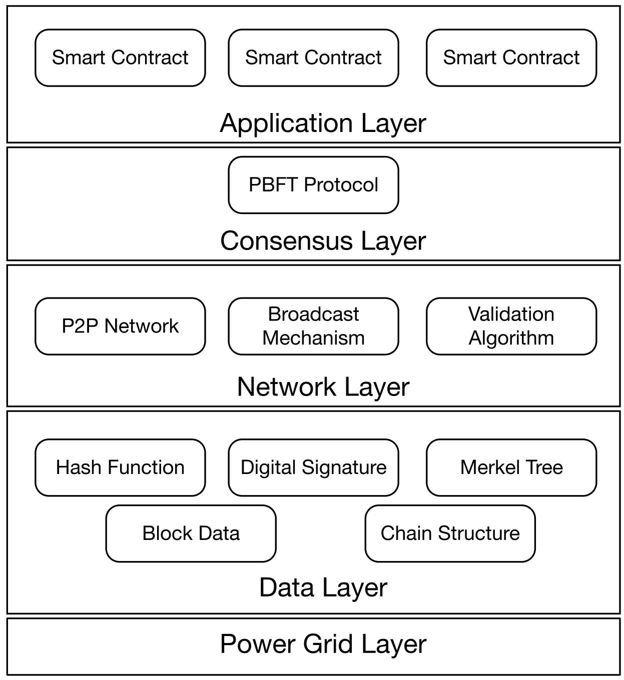


FIGURE 2. The layered design of BLEM.

- **Network layer** inter-connects nodes using peer-to-peer communication protocols. Each node in the network represents a participant in a smart grid system, such as a consumer, a retailer, or a supplier. A node is able to advertise itself to peer nodes and accept connections from others. Thus, node discovery and data transfer are achieved in this layer.
- **Consensus layer** is responsible for the management of block orders and validation of newly generated blocks. Instead of the widely used Proof-of-Work (PoW) protocol, we adopt the Practical Byzantine Fault Tolerance Algorithm (PBFT) [33] as the default consensus scheme. The difference is that PoW requires every node to solve a mathematical puzzle (usually called *mining*) to compete for the right to add a new block, which is computationally intensive, while PBFT is based on the voting mechanism and is able to tolerate no more than 33% node failures. Since participants in our system are not guaranteed to have powerful computing resources, PBFT is more suitable than PoW.
- **Application layer** is where smart contracts are deployed. Typically, a smart contract consists of a piece of program code, a storage file, and an account balance. The program code defines a set of rules, which cannot be changed after deployment. The storage file is used to store necessary data in the smart contract. It is persistent such that single node failure would not affect the smart contract. We set an account balance in order to restrain the over-consumption of computing and storage resources. As long as a smart contract is deployed, it is executed by all consensus nodes.

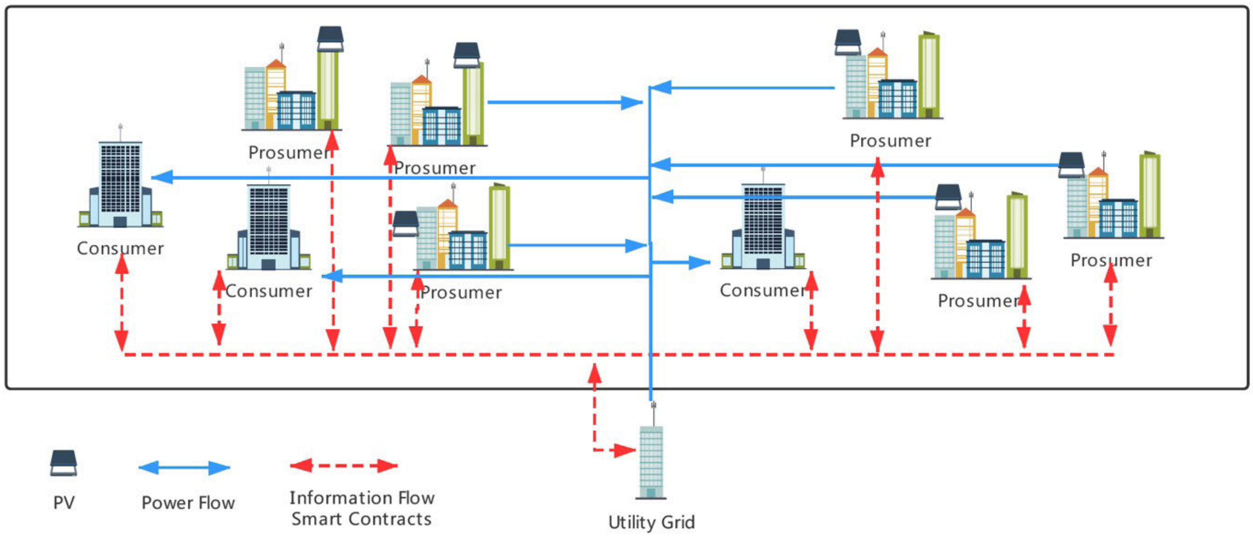


FIGURE 3. **The working scheme of BLEM.**

C. PROBLEM FORMULATION

As illustrated in Figure 3, we model the P2P trading problem in Blockchain-based Local Energy Market as follows. Consider a community of households $\mathcal{H}$. All households are connected through the local electricity distribution network. If $h \in \mathcal{H}$ is a prosumer, the household is equipped with solar photovoltaic panels, and it can sell its surplus solar energy to the neighbors. If h is a consumer, it does not have energy generation technologies and has to buy electricity from either the grid or prosumer households. For each household h_i, the renewable energy production at time slot t is denoted as g_i^t, and the electricity demand at time slot t is $d_{i,t}$. Suppose $s_{i,j}^t$ is the amount of electricity sold from household h_i to h_j with price p_i^t at time slot t. The objective of BLEM is to minimize the cost of electricity transactions.

$$\min \sum_{h_i \in \mathcal{H}} \sum_{h_j \in \mathcal{H}, i \neq j} s_{i,j}^t * p_i^t \tag{1}$$

The cost minimization is subject to the following constraints. First, the total amount of electricity sold by household h_i should never exceed its supply at time slot t:

$$\sum_{h_j \in \mathcal{H}} s_{i,j}^t \leq g_i^t. \tag{2}$$

Then, the net amount of electricity purchased by household h_i should satisfy its demand:

$$\sum_{h_k \in \mathcal{H}, k \neq i} s_{k,i}^t * \psi_{k,i} \geq d_{i,t} + \sum_{h_j \in \mathcal{H}, i \neq j} s_{i,j}^t, \tag{3}$$

where $\psi_{i,j}$ is the distribution network loss from household h_i to h_j, which is proportional to the distance between h_i and h_j. The left term of equation (3), $\sum_{h_k \in \mathcal{H}, k \neq i} s_{k,i}^t * \psi_{k,i}$ is the amount of electricity purchased by h_i, and $\sum_{h_j \in \mathcal{H}, i \neq j} s_{i,j}^t$ is the amount of electricity sold by h_i.

If h_j cannot purchase enough electricity from peer households, it has to buy electricity directly from the grid. In particular, the grid can also be treated as a special participant in the BLEM with unlimited supply and no demand at every time slot, *i.e.*, $g_i^t = +\infty$ and $d_{i,t} = 0$.

D. DEMAND PREDICTION

The existing studies on power demand prediction are mainly based on linear models such as auto-regressive integrated moving average (ARIMA) [34]–[36]. In recent years, methods based on deep learning have also been proposed to predict power consumption [37]–[40]. These approaches, however, are mainly applied to small or synthetic datasets. In this paper, we address the power consumption prediction problem in a BLEM, which involves hundreds or thousands of households. Such a task is more difficult due to the following challenges.

- Data incompleteness. Smart meter readings are often gathered using low power wide area network (LPWAN) technologies such as NB-IoT or Lora [41]. Data packets might be lost during wireless transmission.
- Computational complexity. A typical BLEM may contain hundreds of smart meters. Predicting such a large number of time series with complex models is computationally impractical.
- Spatio-temporal correlations. Power consumption from various houses is usually correlated with each other. Thus, it is unwise to predict power consumption independently.

To address these challenges, we propose to adopt the Temporal Regularized Matrix Factorization (TRMF) model [42] for city-wide power consumption prediction.

Problem 1: 1 Assume the power consumption data gathered from smart meters is denoted by a matrix $D \in \mathcal{R}^{N \times T}$. $D = [d_1; d_2; \cdots; d_N]$ is a multi-dimensional time series.

TABLE 1. The definition of symbols.

System Definition	
$\mathcal{H}$	the set of households in the BLEM
h_i	the household i, $h_i \in \mathcal{H}$
T	time slots
g_i^t	the renewable energy production of household h_i at time slot t
$d_{i,t}$	the electricity demand of household h_i at time slot t
$s_{i,j}^t$	the amount of electricity sold from household h_i to h_j
p_i^t	the unit price of electricity sold by household h_i
$\psi_{i,j}$	the distribution network loss from household h_i to h_j
Demand Prediction	
D	the electricity demand of all households over time $D \in \mathcal{R}^{N\times T}$
d_i	the power consumption data of household h_i, $d_i \in \mathcal{R}^{1\times T}$
F	the matrix of latent spatial embeddings, $F \in \mathcal{R}^{K\times N}$
f_i	the latent embedding of d_i, $f_i \in \mathcal{R}^{K\times 1}$
X	the matrix of latent temporal embeddings, $X \in \mathcal{R}^{K\times T}$
x_t	the latent temporal embedding at time t, $x_t \in \mathcal{R}^{K\times 1}$
$\mathcal{L}$	the set of lag indices
b	the order of TRMF model
W	the set of coefficient matrices, $W = \{W_j \| W_j \in \mathcal{R}^{K\times K}, 1 \leq j \leq b\}$
Ω	the set of observed entries in matrix D
$R_f(F)$	the regularizer for F
$R_w(W)$	the regularizer for W
$R_x(X)$	the temporaldependencies among time series data
Bipartite Graph Trading	
G	the bipartite graph
V_P	the vertex set of prosumers
V_C	the vertex set of consumers
E	the edge set
pg_i^t	the electricity supply of prosumer vertex v_i
pd_j^t	the electricity demand of consumer vertex v_j
$c_{i,j}$	the unit cost of v_j purchasing electricity from v_i
$dist_{i,j}$	the euclidean distance between v_i and v_j

Each row

$$d_i = (d_{i,1}, d_{i,2}, \cdots, d_{i,T}), \quad 1 \leq i \leq N \tag{4}$$

represents the power consumption data collected from the i-th consumer, where the j-th entry d_{ij} is the power consumption of consumer i at time j. Given a partially observed matrix D at time t, the goal is to predict the power consumption at time $t+1$, i.e., $[d_{1,t+1}, d_{2,t+1}, \cdots, d_{N,t+1}]$.

The matrix D can be factorized with two matrices following TRMF:

$$D \approx F^{\mathsf{T}} X, \tag{5}$$

where $F \in \mathcal{R}^{K\times N}$ and $X \in \mathcal{R}^{K\times T}$. The matrix $F = [f_1 f_2 \cdots f_N]$ consists of N column vectors $\{f_i | f_i \in \mathcal{R}^{K\times 1}, 1 \leq i \leq N\}$, where f_i is a latent embedding for d_i. Similarly, the matrix $X = [x_1\ x_2\ \cdots\ x_T]$ is composed of T column vectors $\{x_t | x_t \in \mathcal{R}^{K\times 1}, 1 \leq i \leq T\}$, where x_t is the latent temporal embedding at time t. Thus, the power consumption of consumer i at time t can be represented by the product of latent embeddings

$$d_{i,t} = f_i^{\mathsf{T}} x_t. \tag{6}$$

We then assume that x_t is a linear combination of previous states with random noise. The temporal dependencies among $\{x_t\}$ can be expressed explicitly:

$$x_t = \sum_{j=1}^{b} W_j x_{t-l_j} + \epsilon_t, \tag{7}$$

where b is the order of this model, $\mathcal{L} = \{l_1, l_2, \ldots, l_j, \ldots, l_b\}$ is the set containing the lag indices, each W_j in $W = \{W_j | W_j \in \mathcal{R}^{K\times K}, 1 \leq j \leq b\}$ is the coefficient matrix, and ϵ_t is a Gaussian noise vector with zero mean.

Thus, the objective is to solve the following problem:

$$\min_{F,X,W} \sum_{(i,t)\in\Omega} (d_{it} - f_i^{\mathsf{T}} x_t)^2 + \lambda_f R_f(F) + \lambda_w R_w(W) + \lambda_x R_x(X), \tag{8}$$

where Ω is the set of observed entries in matrix D, $R_f(F)$ and $R_w(W)$ are regularizers for F and W to avoid overfitting. In specific, $R_x(X)$ are introduced to model the temporal dependencies among time series:

$$R_x(X) = \frac{1}{2} \sum_{t=l_{max}}^{T} \left\| x_t - \sum_{j=1}^{b} W_j x_{t-l_j} \right\|^2 + \frac{\eta}{2} \sum_t \|x_t\|^2, \tag{9}$$

where $l_{max} = max(\mathcal{L}) + 1$. We solve the problem by alternating minimization. The parameters are initialized randomly, and then iteratively updated as follows:

$$F = \operatorname*{argmin}_F \sum_{(i,t)\in\Omega} (d_{it} - f_i^{\mathsf{T}} x_t)^2 \tag{10}$$

$$X = \operatorname*{argmin}_X \sum_{(i,t)\in\Omega} (d_{it} - f_i^{\mathsf{T}} x_t)^2 + \lambda_x R_x(X) \tag{11}$$

$$W = \operatorname*{argmin}_W \lambda_x R_x(X) + \lambda_w R_w(W) \tag{12}$$

The detailed algorithm for power usage prediction is illustrated in Algorithm 1. First, we train the prediction model by alternatively minimizing F, X, and W. Then, the missing entries are imputed by the product of learned F and X. We further infer the latent temporal embedding x_{T+1} based on temporal dependencies described in equation (7) and thus derive the power usage data at time $T+1$.

E. TRADING ON BIPARTITE GRAPHS

After precise predicting household demands, we then construct a bipartite graph $G^t = (V_P, V_C, E)$ as follows. A household h_i is defined as a *prosumer* if its power generation is larger than its own demand at time slot t. We create a node $v_i \in V_P$, and its electricity supply is defined as:

$$pg_i^t = g_i^t - d_i^t. \tag{13}$$

Oppositely, a household h_j is defined as a *consumer* if its power generation is larger than its own demand. We create a node $v_j \in V_C$, and its electricity demand is defined as:

$$pd_j^t = d_j^t - g_j^t. \tag{14}$$

Algorithm 1 Power Usage Prediction Algorithm

Input: $D \in \mathcal{R}^{N \times T}$, max_iter
Output: $pred = [d_{1,T+1}, d_{2,T+1}, \cdots, d_{N,T+1}]$

```
// train model
initialize W, F, X
for iter ← 1 to max_iter do
    update F by solving (10)
    update X by solving (11) using GRALS [43]
    update W by solving (12) using Cholesky
    factorization [44]
end for

// impute missing values
for i ← 1 to N do
    for t ← 1 to T do
        if D_{i,t} == nan then
            D_{i,t} = f_i^T x_t
        end if
    end for
end for

// predict future power usage
x_{T+1} = Σ_{j=1}^{b} W_j x_{T+1−l_j}
pred = F · x_{T+1}
return pred
```

We assume that households with equal electricity generation and demand do not participate in the P2P transactions at the current time slot. Since the distribution network loss $\psi_{i,j}$ is proportional to the distance between h_i and h_j, we define the unit cost of h_j purchasing electricity from h_i as

$$c_{i,j} = p_i * \psi_{i,j} = \alpha p_i * dist_{i,j}, \tag{15}$$

where $dist_{i,j}$ is the euclidean distance between h_i and h_j. The optimization problem then can be defined as follows:

$$\begin{aligned} \text{minimize} \quad & \sum_{i\in[1,m]} \sum_{j\in[1,n]} s^t_{i,j} * c_{i,j} \\ \text{subject to} \quad & \sum_{j\in[1,n]} s^t_{i,j} \le pg^t_i, \quad \forall i \in [1, m] \\ & \sum_{i\in[1,m]} s^t_{i,j} \ge pd^t_j, \quad \forall j \in [1, n] \\ & s^t_{i,j} \in \mathbb{N}, \quad \forall i \in [1, m], \ \forall j \in [1, n] \end{aligned} \tag{16}$$

where $m = |V_p|$ and $n = |V_c|$.

The optimization problem (16) is an integer linear program (ILP). In order to solve it efficiently, we first transform it into a linear program (17) by relaxing the integrality constraints of $s^t_{i,j}$. And then we prove that these two problems are equivalent, *i.e.*, the optimal solution of (17) is also the optimal

```solidity
pragma solidity ^0.4.17;

contract PowerPromotion {

    function update(uint p, uint q) public {
        require(owner == msg.sender);

        power = q;
        u_price = p;

        ExchangeOracle oracle = ExchangeOracle(exchange);
        oracle.update(u_price, power);
    }

    function sell(uint up) payable public {
        require(msg.value < power * u_price);
        require(up == u_price);

        owner.transfer(msg.value);

        emit PowerDelivered(msg.sender, msg.value / u_price);

        power = power - msg.value / u_price;
        if (power == 0) {
            emit PowerOut(owner);
        }

        ExchangeOracle oracle = ExchangeOracle(exchange);
        return oracle.update(u_price, power);
    }
}
```

FIGURE 4. Example code snippet of a prosumer's smart contract.

solution of (16).

$$\begin{aligned} \text{minimize} \quad & \sum_{i\in[1,m]} \sum_{j\in[1,n]} s^t_{i,j} * c_{i,j} \\ \text{subject to} \quad & \sum_{j\in[1,n]} s^t_{i,j} \le pg^t_i, \quad \forall i \in [1, m] \\ & -\sum_{i\in[1,m]} s^t_{i,j} \le -pd^t_j, \quad \forall j \in [1, n] \\ & s^t_{i,j} \ge 0, \quad \forall i \in [1, m], \ \forall j \in [1, n] \end{aligned} \tag{17}$$

Lemma 1: The coefficient matrix in (17) is totally unimodular.

Proof: By definition, a matrix is *totally unimodular* if the determinant of each square sub-matrix is 0, or ±1. The linear constraints of (17) can be written as:

$$As \le b, \tag{18}$$

where $s = [s^t_{1,1} s^t_{1,2} \ldots s^t_{1,n} \ldots s^t_{m,1} s^t_{m,2} \ldots s^t_{m,n}]^\intercal$ is the column vector of unknowns, A is the coefficient matrix and b is the column vector of constant terms. Without loss of generality, suppose $m = 3$ and $n = 2$, and the matrix A is of the following form:

$$A = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ -1 & 0 & -1 & 0 & -1 & 0 \\ 0 & -1 & 0 & -1 & 0 & -1 \end{bmatrix}$$

We can observe that matrix A has the following properties for any m and n:

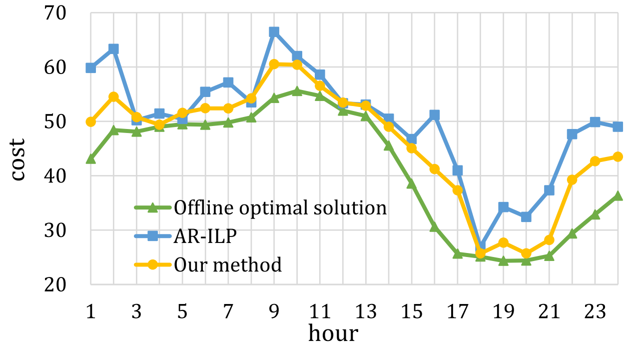


FIGURE 5. Comparison of system performance.

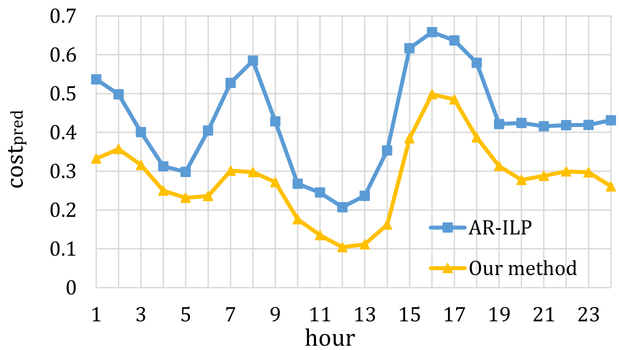


FIGURE 6. Comparison of $cost_{pred}$.

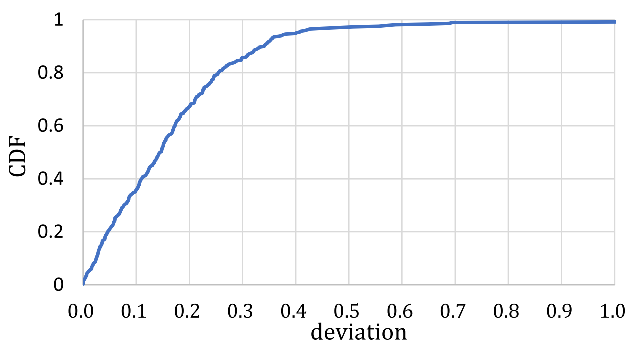


FIGURE 7. The CDF of $|\hat{d_{i,t}} - d_{i,t}|/|d_{i,t}|$.

- Each entry of A is either 0 or ± 1.
- Every column of A has at most two non-zero entries.
- The row indices $[1, m]$ can be partitioned into two disjoint sets M_1 and M_2, such that for each column j, $\sum_{i \in M_1} A_{i,j} = \sum_{i \in M_2} A_{i,j}$.

According to [46], if a matrix satisfies these properties, it is *totally unimodular*. Therefore, the coefficient matrix in (17) is *totally unimodular*. □

Theorem 1 [47]: If the coefficient matrix of a linear program satisfies the totally unimodular property, the polytope of feasible solutions of the linear program is integral, i.e., all the vertices of the feasible set are integer.

Theorem 2: The optimal solution of problem (17) is also the optimal solution of (16).

Proof: Suppose the optimal solutions of problem (16) and (17) are denoted as OPT_{ILP} and OPT_{LP} respectively. On the one hand, since problem (17) is a relaxation of problem (16), we have $OPT_{LP} <= OPT_{ILP}$. On the other hand, according to Theorem 1, the elements of vector s^* corresponding to OPT_{LP} are all integers. Thus, s^* is also a solution of problem (16), and $OPT_{ILP} <= OPT_{LP}$. As a result, we can get $OPT_{ILP} = OPT_{LP}$. □

Algorithm 2 Trading on Bipartite Graphs

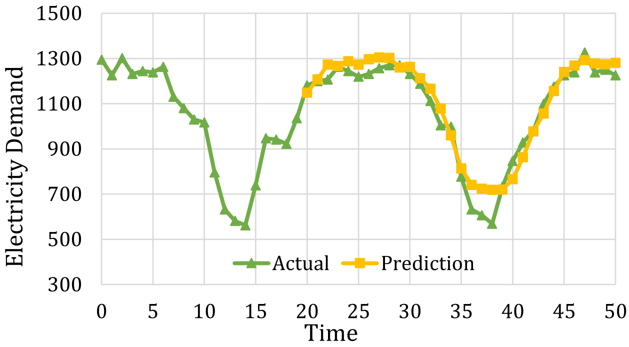
for $t \leftarrow 1, T$ **do**
  // bipartite graph construction
  $V_P = \emptyset, V_C = \emptyset, E = \emptyset$
  for household $h_i \in \mathcal{H}$ **do**
    if $g_i^t > d_i^t$ **then**
      create node v_i
      $V_P = V_P \cup \{v_i\}$
      $pg_i^t = g_i^t - d_i^t$
    else if $g_i^t < d_i^t$ **then**
      create node v_i
      $V_C = V_C \cup \{v_i\}$
      $pd_i^t = d_i^t - g_i^t$
    end if
  end for
  for $v_i \in V_P$ **do**
    for $v_j \in V_C$ **do**
      create edge $e = (v_i, v_j)$
      $E = E \cup \{e\}$
      the unit cost of sending electricity through e is $c_{i,j} = \alpha p_i * dist_{i,j}$
    end for
  end for
  $G = (V_P, V_C, E)$

  // trading
  solving problem (17) with Hungarian method [45]
  the optimal trading scheme at time slot t is $s^* = \operatorname{argmin}_{\{s_{i,j}^t\}} \sum_{i \in [1,m]} \sum_{j \in [1,n]} s_{i,j}^t * c_{i,j}$
end for

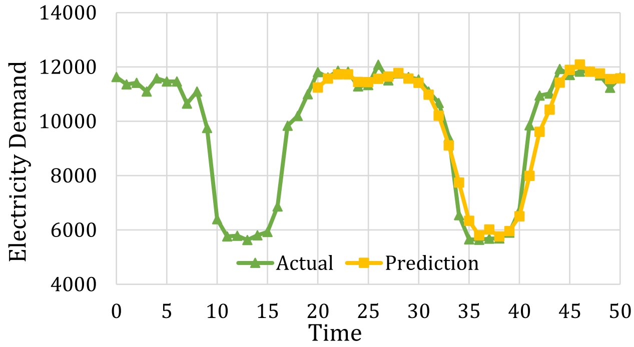
Based on previous analysis, at each time slot t, we first construct a bipartite graph based on households' electricity demand and generation. And then, we solve the optimization problem (17). $s^* = \operatorname{argmin}_{\{s_{i,j}^t\}} \sum_{i \in [1,m]} \sum_{j \in [1,n]} s_{i,j}^t * c_{i,j}$ is the trading scheme that minimizes trading cost. The detailed algorithm is illustrated in Algorithm 2.

F. IMPLEMENTATION

We implement DS^2 on a private Ethereum blockchain. Each household in the BLEM is associated with a unique address as its digital account. The address of the digital account is further attached with a smart contract which is responsible for depositing and withdrawing money. Transactions are accomplished via smart contracts deployed by both prosumers and customers. At the end of each time slot, a prosumer's smart



(a) Household 200



(b) Household 240

FIGURE 8. Electricity demand prediction by DS^2. We randomly select 2 households from the dataset.

contract updates its power supply, and a consumer's smart contract places orders for the next time slot. Smart contracts are implemented with Solidity program [48] running in Ethereum Virtual Machine. We utilize the JavaScript API provided by Ethereum to handle the interactions between smart contracts. In order to accelerate transactions, we use PBFT (Practical Byzantine Fault Tolerance) algorithm [33] for consensus, which is more time-efficient than PoW. We also assume that there are no transaction costs and compensation for miners. Figure 4 presents a code snippet of a prosumer's smart contract.

IV. EVALUATION

In this section, we evaluate the performance of DS^2 and compare it with both the offline optimal method and a baseline algorithm AR-ILP.

A. METHODOLOGY

1) DATASET

The data used in our experiment is extracted from UCI Electricity Load Diagrams 2011-2014 Dataset [49]. It contains electricity consumption data of 370 clients across four years. The data of each client is recorded for each 15-minute time slot. We aggregate the raw dataset and transform it into an hourly dataset.

2) EVALUATION METRIC

The total cost of BLEM consists of two parts. The first part is the sum of transaction costs among all prosumers and consumers as defined in equation (16). The second part is the prediction cost, which is the deviation of demand prediction. We take it into account because an inaccurate prediction results in either electricity waste or additional purchase.

$$cost = \alpha \cdot cost_{tran} + \beta \cdot cost_{pred},$$

where

$$cost_{tran} = \sum_{i\in[1,m]} \sum_{j\in[1,n]} s^t_{i,j} * c_{i,j},$$

$$cost_{pred} = \frac{\sum_i |\hat{d_{i,t}} - d_{i,t}|}{\sum_i |d_{i,t}|}.$$

3) EXPERIMENT SETUP

The smart contracts are implemented using Solidity, which is an object-oriented and high-level programming language targeting the Ethereum Virtual Machine (EVM). For the convenience of development, we use *solcjs* as the compiler for Solidity. The simulations are performed on a server running Ubuntu 16.10 with an i7 CPU and 64 GB RAM.

4) BASELINE ALGORITHMS

The ground-truth is obtained by an offline optimal method which solves problem (16) with known electricity demand data. It is worth noticing that we cannot get the offline optimal results in practice since we do not have future data. We also implement a baseline algorithm AR-ILP. The main idea of AR-ILP is first to use an autoregressive model to predict hourly electricity demand and then optimize the integer linear programming problem (16).

B. EXPERIMENT RESULTS

1) SYSTEM PERFORMANCE

For comparison, we present the system performance of DS^2, AR-ILP as well as the groundtruth in Figure 5. By running the three algorithms for ten times independently, we obtain that compared with the ground truth, the mean absolute percentage errors (MAPE) of DS^2 and AR-ILP are 12.8% and 24.9% respectively, where

$$MAPE(DS^2) = \frac{1}{r}\sum_{i=1}^{r} \frac{|cost(DS^2) - cost(groundtruth)|}{cost(groundtruth)}.$$

2) ANALYSIS OF $cost_{pred}$

Figure 6 shows the comparison results of $cost_{pred}$. The electricity demand in both algorithms is predicted using 30 days of historical data. We can find the DS^2 achieves less prediction cost than AR-ILP. The average $cost_{pred}$ values are 0.28 and 0.43 respectively.

3) ANALYSIS OF PREDICTION ACCURACY

We first compute the cumulative distribution function of prediction deviation $|\hat{d_{i,t}} - d_{i,t}|/|d_{i,t}|$ of all households. As shown in Figure 7, the median value of prediction deviation is 0.14, and the 90th percentile is 0.34. Then Figure 8 depicts the demand prediction results of DS^2 by randomly selecting two households from the dataset.

V. CONCLUSION AND FUTURE WORK

This paper proposes DS^2, a data-driven, secure, and smart power grid system in BLEM. By utilizing blockchain structure, DS^2 enables both secure transaction data storage and smart execution of transactions. We prototype DS^2 on the Ethereum blockchain. To demonstrate its effectiveness, we solve the local energy market (LEM) problem by reducing it into an optimization problem and implement the protocols using smart contracts. The evaluation shows that DS^2 achieves a MAPE of 12.8% compared with the offline optimal method.

Currently, we mainly focus on the P2P trading problem in this paper. Besides the trading mechanism, there are many other issues to be considered in the LEM. For example, given a LEM where there is no sufficient power supply, it is not clear how prosumers can price their generated electricity and trade with consumers to guarantee fairness. We will leave these problems for future study.

REFERENCES

[1] H. Farhangi, "The path of the smart grid," *IEEE Power Energy Mag.*, vol. 8, no. 1, pp. 18–28, Jan./Feb. 2009.

[2] V. C. Gungor, D. Sahin, T. Kocak, S. Ergut, C. Buccella, C. Cecati, and G. P. Hancke, "Smart grid technologies: Communication technologies and standards," *IEEE Trans. Ind. Informat.*, vol. 7, no. 4, pp. 529–539, Nov. 2011.

[3] X. Fang, S. Misra, G. Xue, and D. Yang, "Smart grid—The new and improved power grid: A survey," *IEEE Commun. Surveys Tuts.*, vol. 14, no. 4, pp. 944–980, Qua. 2012.

[4] A. Lüth, J. M. Zepter, P. Crespo del Granado, and R. Egging, "Local electricity market designs for peer-to-peer trading: The role of battery flexibility," *Appl. Energy*, vol. 229, pp. 1233–1243, Nov. 2018.

[5] S. Nguyen, W. Peng, P. Sokolowski, D. Alahakoon, and X. Yu, "Optimizing rooftop photovoltaic distributed generation with battery storage for peer-to-peer energy trading," *Appl. Energy*, vol. 228, pp. 2567–2580, Oct. 2018.

[6] C. W. Gellings, *The Smart Grid: Enabling Energy Efficiency Demand Response*. Boca Raton, FL, USA: CRC Press, 2020.

[7] X. Zhang, W. Pei, W. Deng, Y. Du, Z. Qi, and Z. Dong, "Emerging smart grid technology for mitigating global warming," *Int. J. Energy Res.*, vol. 39, no. 13, pp. 1742–1756, Oct. 2015.

[8] *Smart Meters Can be Hacked to Cut Power Bills*. Accessed: Jan. 10, 2021. [Online]. Available: https://www.bbc.com/news/technology-29643276

[9] M. B. Mollah, J. Zhao, D. Niyato, K.-Y. Lam, X. Zhang, A. M. Y. M. Ghias, L. H. Koh, and L. Yang, "Blockchain for future smart grid: A comprehensive survey," *IEEE Internet Things J.*, vol. 8, no. 1, pp. 18–43, Jan. 2021.

[10] N. Szabo, "Formalizing and securing relationships on public networks," *1st Monday*, vol. 2, no. 9, Sep. 1997. [Online]. Available: https://journals.uic.edu/ojs/index.php/fm/article/view/548, doi: 10.5210/fm.v2i9.548.

[11] Z. Liu, D. Wang, J. Wang, X. Wang, and H. Li, "A blockchain-enabled secure power trading mechanism for smart grid employing wireless networks," *IEEE Access*, vol. 8, pp. 177745–177756, 2020.

[12] S. Park, J. Lee, S. Bae, G. Hwang, and J. K. Choi, "Contribution-based energy-trading mechanism in microgrids for future smart grid: A game theoretic approach," *IEEE Trans. Ind. Electron.*, vol. 63, no. 7, pp. 4255–4265, Jul. 2016.

[13] W. Tushar, T. K. Saha, C. Yuen, D. Smith, and H. V. Poor, "Peer-to-peer trading in electricity networks: An overview," *IEEE Trans. Smart Grid*, vol. 11, no. 4, pp. 3185–3200, Jul. 2020.

[14] J. Guerrero, A. C. Chapman, and G. Verbic, "Decentralized P2P energy trading under network constraints in a low-voltage network," *IEEE Trans. Smart Grid*, vol. 10, no. 5, pp. 5163–5173, Sep. 2019.

[15] N. U. Ibne Hossain, M. Nagahi, R. Jaradat, C. Shah, R. Buchanan, and M. Hamilton, "Modeling and assessing cyber resilience of smart grid using Bayesian network-based approach: A system of systems problem," *J. Comput. Design Eng.*, vol. 7, no. 3, pp. 352–366, Jun. 2020.

[16] N. Mhaisen, N. Fetais, and A. Massoud, "Secure smart contract-enabled control of battery energy storage systems against cyber-attacks," *Alexandria Eng. J.*, vol. 58, no. 4, pp. 1291–1300, Dec. 2019.

[17] N. U. I. Hossain, R. Jaradat, S. Hosseini, M. Marufuzzaman, and R. K. Buchanan, "A framework for modeling and assessing system resilience using a Bayesian network: A case study of an interdependent electrical infrastructure system," *Int. J. Crit. Infrastruct. Protection*, vol. 25, pp. 62–83, Jun. 2019.

[18] T. Morstyn and M. D. McCulloch, "Multiclass energy management for peer-to-peer energy trading driven by prosumer preferences," *IEEE Trans. Power Syst.*, vol. 34, no. 5, pp. 4005–4014, Sep. 2019.

[19] S. Boyd, N. Parikh, and E. Chu, *Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers*. New York, NY, USA: Now, 2011.

[20] C. Long, J. Wu, Y. Zhou, and N. Jenkins, "Peer-to-peer energy sharing through a two-stage aggregated battery control in a community microgrid," *Appl. Energy*, vol. 226, pp. 261–276, Sep. 2018.

[21] A. Paudel, K. Chaudhari, C. Long, and H. B. Gooi, "Peer-to-peer energy trading in a prosumer-based community microgrid: A game-theoretic model," *IEEE Trans. Ind. Electron.*, vol. 66, no. 8, pp. 6087–6097, Aug. 2019.

[22] T. Morstyn, A. Teytelboym, and M. D. Mcculloch, "Bilateral contract networks for peer-to-peer energy trading," *IEEE Trans. Smart Grid*, vol. 10, no. 2, pp. 2026–2035, Mar. 2019.

[23] X. Zhang, S. Zhu, J. He, B. Yang, and X. Guan, "Credit rating based real-time energy trading in microgrids," *Appl. Energy*, vol. 236, pp. 985–996, Feb. 2019.

[24] F. Luo, Z. Y. Dong, G. Liang, J. Murata, and Z. Xu, "A distributed electricity trading system in active distribution networks based on multi-agent coalition and blockchain," *IEEE Trans. Power Syst.*, vol. 34, no. 5, pp. 4097–4108, Sep. 2019.

[25] J. Kang, R. Yu, X. Huang, S. Maharjan, Y. Zhang, and E. Hossain, "Enabling localized peer-to-peer electricity trading among plug-in hybrid electric vehicles using consortium blockchains," *IEEE Trans. Ind. Informat.*, vol. 13, no. 6, pp. 3154–3164, Dec. 2017.

[26] Z. Li, J. Kang, R. Yu, D. Ye, Q. Deng, and Y. Zhang, "Consortium blockchain for secure energy trading in industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 14, no. 8, pp. 3690–3700, Aug. 2018.

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

[28] M. T. Devine and P. Cuffe, "Blockchain electricity trading under demurrage," *IEEE Trans. Smart Grid*, vol. 10, no. 2, pp. 2323–2325, Mar. 2019.

[29] S. Noor, W. Yang, M. Guo, K. H. van Dam, and X. Wang, "Energy demand side management within micro-grid networks enhanced by blockchain," *Appl. Energy*, vol. 228, pp. 1385–1398, Oct. 2018.

[30] A. Pieroni, N. Scarpato, L. Di Nunzio, F. Fallucchi, and M. Raso, "Smarter city: Smart energy grid based on blockchain technology," *Int. J. Adv. Sci. Eng. Inf. Technol*, vol. 8, no. 1, pp. 298–306, 2018.

[31] J. J. Sikorski, J. Haughton, and M. Kraft, "Blockchain technology in the chemical industry: Machine-to-machine electricity market," *Appl. Energy*, vol. 195, pp. 234–246, Jun. 2017.

[32] M. Szydlo, "Merkle tree traversal in log space and time," in *Proc. Int. Conf. Theory Appl. Cryptograph. Techn.* Berlin, Germany: Springer, 2004, pp. 541–554.

[33] M. Castro and B. Liskov, "Practical Byzantine fault tolerance," in *Proc. OSDI*, vol. 99, 1999, pp. 173–186.

[34] P. Chujai, N. Kerdprasop, and K. Kerdprasop, "Time series analysis of household electric consumption with ARIMA and ARMA models," in *Proc. Int. MultiConference Eng. Comput. Scientists*, vol. 1, 2013, pp. 295–300.

[35] C. Deb, F. Zhang, J. Yang, S. E. Lee, and K. W. Shah, "A review on time series forecasting techniques for building energy consumption," *Renew. Sustain. Energy Rev.*, vol. 74, pp. 902–924, Jul. 2017.

[36] S. Barak and S. S. Sadegh, "Forecasting energy consumption using ensemble ARIMA–ANFIS hybrid algorithm," *Int. J. Elect. Power Energy Syst.*, vol. 82, pp. 92–104, Nov. 2016.
[37] S. Ai, A. Chakravorty, and C. Rong, "Household power demand prediction using evolutionary ensemble neural network pool with multiple network structures," *Sensors*, vol. 19, no. 3, p. 721, Feb. 2019.
[38] A. Ullah, K. Haydarov, I. Ul Haq, K. Muhammad, S. Rho, M. Lee, and S. W. Baik, "Deep learning assisted buildings energy consumption profiling using smart meter data," *Sensors*, vol. 20, no. 3, p. 873, Feb. 2020.
[39] M. M. Rana, R. Bo, and B. J. Choi, "Kalman filter based electricity market states forecasting: A state-space framework," in *Proc. IEEE 9th Annu. Int. Conf. CYBER Technol. Autom., Control, Intell. Syst. (CYBER)*, Jul. 2019, pp. 1432–1435.
[40] T.-Y. Kim and S.-B. Cho, "Predicting residential energy consumption using CNN-LSTM neural networks," *Energy*, vol. 182, pp. 72–81, Sep. 2019.
[41] R. S. Sinha, Y. Wei, and S.-H. Hwang, "A survey on LPWA technology: LoRa and NB-IoT," *ICT Exp.*, vol. 3, no. 1, pp. 14–21, Mar. 2017.
[42] H.-F. Yu, N. Rao, and I. S. Dhillon, "Temporal regularized matrix factorization for high-dimensional time series prediction," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 847–855.
[43] N. Rao, H.-F. Yu, P. K. Ravikumar, and I. S. Dhillon, "Collaborative filtering with graph information: Consistency and scalable methods," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, 2015, pp. 2107–2115.
[44] Y. Chen, T. A. Davis, W. W. Hager, and S. Rajamanickam, "Algorithm 887: CHOLMOD, supernodal sparse Cholesky factorization and update/downdate," *ACM Trans. Math. Softw.*, vol. 35, no. 3, pp. 1–14, 2008.
[45] H. W. Kuhn, "The hungarian method for the assignment problem," *Nav. Res. Logistics Quart.*, vol. 2, nos. 1–2, pp. 83–97, Mar. 1955.
[46] I. Heller and C. Tompkins, "An extension of a theorem of Dantzig's," *Linear Inequalities Rel. Syst.*, vol. 38, pp. 247–254, Apr. 1956.
[47] C. H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*. Chelmsford, MA, USA: Courier Corporation, 1998.
[48] *Solidity Programming Language*. Accessed: Jan. 10, 2021. [Online]. Available: https://soliditylang.org/
[49] *UCI Electricity Load Diagrams 2011-2014 Dataset*. Accessed: Jan. 10, 2021. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/ElectricityLoadDiagrams20112014

ZENG ZENG received the M.S. degree in computer science and technology from Nanjing University, Nanjing, China, in 2016. He is currently a Senior Researcher with the Information and Telecommunication Branch, State Grid Jiangsu Electric Power Company. His research interests include the Internet of Things and smart grid.

MEIYA DONG is currently pursuing the Ph.D. degree in electronic science and technology with the Taiyuan University of Technology, Shanxi, China. His current research interests include wireless sensor networks, the Internet of Things, and data processing.

WEIWEI MIAO received the M.S. degree in automation theory and application from Southeast University, Nanjing, China, in 1992. He is currently a Principal Researcher with the Information and Telecommunication Branch, State Grid Jiangsu Electric Power Company. His research interests include power communication systems, wireless access networks, and communication network management.

MINGMING ZHANG is currently a Principal Researcher with the Information and Telecommunication Branch, State Grid Jiangsu Electric Power Company. His research interests include wireless communication and wireless power networks.

HAO TANG received the M.S. degree in computer science and technology from Nanjing University, Nanjing, China, in 2015. He is currently an Assistant Researcher with the Information and Telecommunication Branch, State Grid Jiangsu Electric Power Company. His research interests include software process improvement and practical software engineering.