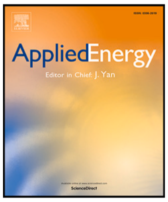
Contents lists available at ScienceDirect

Applied Energy

journal homepage: www.elsevier.com/locate/apenergy



Optimizing bidding strategy in electricity market based on graph convolutional neural network and deep reinforcement learning

Haoen Weng, Yongli Hu [*], Min Liang, Jiayang Xi, Baocai Yin

Beijing Key Laboratory of Multimedia and Intelligent Software Technology, Beijing Institute of Artificial Intelligence, Faculty of Information Technology, Beijing University of Technology, Pingleyuan 100, Chaoyang District, Beijing, 100124, Beijing, China

ARTICLE INFO

Keywords:
Electricity market
Optimal bidding strategy
Multi-agent systems (MAS)
Graph convolutional neural network (GCN)
Deep reinforcement learning (DRL)

ABSTRACT

Formulating optimal bidding strategies is pivotal for market participants to enhance electricity market profits. The main challenge for finding optimal bidding strategies is how to deal with system uncertainty, which stems from the inherent unpredictability and fluctuation within the electricity market. In the previous works, deep reinforcement learning (DRL) is proved a promising approach in multi-agent system with uncertainty. But few works model the relevance between agents for processing system uncertainty, especially the dynamic correlation in the operation of market. For this purpose, this paper proposes to model the correlation between agents to cope with the system uncertainty in a representative centralized double-sided auction market by combining graph convolutional neural network (GCN) with deep deterministic policy gradient (DDPG) algorithm, which is not only able to deal with the system uncertainty by aggregating correlative information of neighboring agents, but also helps obtain superior bidding strategies for the market participants. The proposed algorithm is evaluated on 4-bus, 30-bus and 57-bus congested network, where both supply side and demand side with elastic demand are modeled as RL agents. The results demonstrate that the proposed algorithm achieves higher system profits than the DRL based algorithms without GCN.

1. Introduction

Finding the optimal bidding strategy for market participants has become a research focus since the relaxation of regulations in the electricity market. With the increasing penetration of renewable energy, which introduces greater system uncertainty, developing bidding strategies for traditional market participants has become more challenging [1]. In the deregulated electricity markets with auction mechanism, the centralized double-sided auction is a widely adopted market scheme, such as most of the electricity market in China. In such market, independent system operator (ISO) matches sellers' supply offers and buyers' demand bids, and calculates market clearing results, including scheduled energy and market clearing price. The bidding strategies employed by Generation Companies (GenCos) and consumers exert a direct impact on determining the market clearing price. As a result, both GenCos and consumers are motivated to devise strategic bidding methods that allow them to influence market clearing outcomes and enhance their revenue [2].

To find optimal bidding strategy in the electricity market, three prevalent classes of approaches have been explored: heuristic methods, game-theoretic and mathematical approaches, and reinforcement learning (RL) algorithms, as listed in Table 1. For example, heuristic methods such as particle swarm optimization (PSO) [3] and genetic algorithms (GAs) [5], as well as co-evolutionary algorithms (CE) [4], have been adopted to simulate the market and assist market participants in bidding strategically. However, these approaches have limitations in simulating the bidding behaviors and improving bidding strategies in large complex power markets including consumers with elastic demand [13]. Additionally, the game-theoretic methods based on the main concept of Nash equilibrium have been employed to solve bi-level optimization problem to discover optimal bidding strategies [6–10]. Mathematical Programs with Equilibrium Constraints (MPEC) and Equilibrium Problem with Equilibrium Constraints (EPEC) are the most widely used game-theoretic methods in these studies. Nevertheless, these methods mostly rely on explicit mathematical formulations for bidding behaviors and ideal assumptions about both the market clearing environment and rivals' strategies, which are unrealistic under most electricity market rules [16,22]. This limited observability of information also results in much system uncertainty for market participants when developing their strategies [23].

System uncertainties, such as demand variations, renewable feed-in, equipment failures, and the complicated bidding behaviors of competitors, pose great challenges for market participants in devising effective

[*] Corresponding author.
E-mail address: huyongli@bjut.edu.cn (Y. Hu).

https://doi.org/10.1016/j.apenergy.2024.124978
Received 22 June 2024; Received in revised form 24 October 2024; Accepted 20 November 2024
Available online 6 December 2024
0306-2619/© 2024 Elsevier Ltd. All rights are reserved, including those for text and data mining, AI training, and similar technologies.

Table 1
Literature classification. ✓: Yes; ✗: No.

References	Main algorithms	Explicit uncertainty characterization	Information required	
			Environment	Rivals' bids
Heuristic approaches				
[3]	PSO	✗	✓	✓
[4]	CE-GA	✗	✓	✓
[5]	GA	Stochastic programming	✓	✗
Game-theoretic and mathematical approaches				
[6]	MPEC	Stochastic programming	✓	✓
[7]	MPEC	Stochastic programming	✓	✗
[8,9]	EPEC	✗	✓	✓
[10]	EPEC	Stochastic programming	✓	✗
[11,12]	Exhaustive search	Competitors' behavior analysis	✓	✗
Reinforcement learning				
[13]	Q-learning	✗	✗	✗
[14–16]	DDPG	✗	✗	✗
[17]	Q-learning	MARL framework	✗	✗
[18]	DQN	MARL framework	✗	✗
[19]	DDPG	MARL framework	✗	✗
[20]	DDPG-PER	MARL framework	✗	✗
[21]	DDPG	GCN	✓	✗
This paper	DDPG	GCN	✗	✗

bidding strategies based on available data and information. To address this problem, some heuristic approaches and game-theoretic methods integrate stochastic programming (SP), mostly utilizing scenario-based approaches, to model uncertainties arising from load variations, wind generation or rivals' bids [5–7,10]. This also reduces the reliance on the assumption of perfect information. However, scenario-based approaches provide poor out-of-sample performance and fail to accurately model competitors' behavior [12,24]. Therefore, the direct application of scenario-based approaches remains challenged in effectively handling uncertainty in real-world situations characterized by incomplete information about the environment and rivals' strategies. Additionally, some recently developed methods focus on competitors' behavior analysis to characterize uncertainty with imperfect knowledge. A Bayesian inference approach is proposed in [25] to estimate the net aggregate supply curve based on practically available data. Authors in [11,12] further increase the estimation accuracy by introducing the concept of the equivalent rival or decentralized equivalent rival (DER), and then use the estimated competitors' behaviors to find bidding strategy for market participants. However, DER-based approach requires exact physical parameters and structure about the original market clearing model, which is rarely accessible to market participants for constructing the equivalent model.

To address the aforementioned challenges, recently, agent-based simulation and multi-agent systems (MAS) have been proposed to investigate the participant behavior in electricity auction markets. They demonstrate high efficiency, flexibility, and robust generalization capabilities due to more realistic modeling of agent's behavior, such as RL algorithm in MAS [26–28]. RL emerges as a promising solution for problems characterized by uncertainty and incomplete information [29]. RL typically assumes limited information about environment and rivals. Guided by the reward signal, the RL agents can comprehend system uncertainty implicitly and learn their strategies through interactions with the environment. In this context, RL excels in analyzing the dynamic nature of market interactions and optimizing strategies for agents in systems with uncertainty.

RL is a machine learning approach, in which agents learn their strategies through continuous interaction with the environment with limited knowledge about their competitors' information. RL algorithms, such as Roth–Erev learning [30], Q-Learning [31] algorithms and their variants are commonly employed in the bidding problems of electricity market. These algorithms utilize a table to store estimated values for all actions or state–action pairs and update the table by interacting with the environment. Recently, Q-learning was used to address double-sided bidding problems in day ahead electricity markets [13]. However, these tabular based methods become impractical when dealing with large or continuous state spaces due to dimensionality explosion and discretization [32]. For this purpose, the Deep RL (DRL) method has been proposed by combining deep neural network (DNN) with RL. For example, Deep Q-Network (DQN) combines the principles of Q-learning with DNN, which can deal with high-dimensional and continuous state space [33]. Despite this, DQN's action space remains discrete. The Deep Deterministic Policy Gradient (DDPG) algorithm [34] is a policy-based RL algorithm, a deep version of the Deterministic Policy Gradient (DPG) method [35], which can deal with both continuous state and action space. It has been applied to solve the bidding problem of a load serving entity or gas-fired unit [14,15]. However, the system uncertainty in the multi-agent environment has not been well explored in all the researches mentioned above.

Many existing RL algorithms falls in the framework of independent learning, i.e., directly applying single-agent RL (SARL) to multi-agent environment, in which the agents formulate their strategies without explicitly considering the dynamics and non-stationarity within the multi-agent environment [36]. As a result, these methods face the challenge of system uncertainty. To alleviate this problem, some multi-agent RL (MARL) methods have been introduced to model more complex system. A popular framework of MARL is the centralized training with decentralized execution (CTDE), allowing agents to receive the observations and actions from other agents during training [37]. This method trains a critic network with global information to guide the training of actor network. During the testing phase, only local observations are required for execution, making it well-suited for modeling electricity market participants. Recent studies have applied MARL methods to address specific challenges in electricity markets. For instance, a study about double-sided auction market with renewable energy is simulated based on Multi-Agent Q-Learning (MAQL) [17]. A strategy optimization of the double-sided auction market strategy for microgrid with distributed generation is studied [18] based on multi-agent deep Q-network (MADQN). The Multi-Agent Deep Deterministic Policy Gradient (MADDPG) is adopted to address the day-ahead single-sided electricity market bidding problem of GenCo bidders [19]. MADDPG combined with Prioritized Experience Replay (PER), namely MADDPG-PER, is developed to find bidding strategies for both supply side and demand side in double-sided market [20].

Despite much efforts to address system uncertainty by adopting MARL framework, dealing with large-scale power systems with numerous agents remains a challenge. When the number of agents becomes large, the input dimension of the centralized critic network increases sharply, which not only brings the difficulty of value estimation of critic network, but also significantly increases the training time. Additionally, the effectiveness of CTDE relies on the presence of a central learner or controller capable of observing and aggregating information from all agents to manage and coordinate the training process. This constraint limits the wide applications of CTDE in real-world, as such central controller may be unavailable and distributed training is often more applicable in the competitive bidding game [38]. In this context, modeling the system correlation would be another promising way to deal with system uncertainty, as the market clearing results are the product of the interactions between all participants' quotations. For example, Graph convolutional neural networks (GCN), a deep learning method designed for handling graph-structured data by capturing the relationships between nodes, has garnered high attention and been applied to various applications in power systems, such as price or load forecasting by utilizing spatial–temporal neighbor information [39,40], and topology-aware power flow calculation [41].

Although modeling correlation by GCN is a potential and powerful technique to address system uncertainty, few previous works have adopted GCN in simulating bidding behaviors and finding optimal bidding strategy for market participants. The available related work in [21] provided a solution for modeling the spatial correlation of the system by using GCN to embed the physical topology information into the network of actor–critic algorithm for generation units in the single-sided uncongested system. The result demonstrates robust ability in increasing the system profits. This is a successful attempt to address system uncertainty from another perspective of considering system correlation. However, the effectiveness of the method in [21] also relies on prior knowledge of the physical connections of the power system, which may not be accessible to market participants. Additionally, for the issue of optimal bidding strategy, the spatial correlation between nodes in the network is not only derived from the physical system topology, but also from the relative bidding relationships of market participants. The former is a fixed prior relationship, while the latter reflects the correlation of complicated bidding behaviors. For the centralized double-sided bidding market including both supply-side and demand-side, these two types of information determine the final market clearing result, such as the nodal price. Therefore, simply embedding the physical network topology is not sufficient to model the spatial correlation between agents, especially in real-world double-sided auction markets with transmission constraints and demand-side bidding behavior, both of which have been ignored in [21]. Moreover, most existing works based on RL do not consider enough historical information and its temporal correlation for better bidding strategies. Note that in partially observable MAS, uncertainty arises due to limited access to global information, such as rivals' actions, which also influence the market clearing price. One way to mitigate this uncertainty in a partially observable environment is to leverage historical data [36,42]. By incorporating past information, agents can enhance their understanding of system dynamics, leading to more informed decision-making and improved adaptability.

In this work, aiming to solve the problem of optimizing bidding strategy in centralized double-sided auction market with both GenCos and consumers, we introduce GCN into the DDPG algorithm to cope with system uncertainty by modeling the correlation between market participants, forming a novel method for optimal bidding strategy in double-sided auction market, namely DDPG-GCN, in which the graph adjacency matrix can be learned in a self-adaptive manner. The main contributions of the paper are summarized as follows:

(1) A novel multi-agent system based on DDPG algorithm is proposed to solve the bi-level optimization problem of double-sided auction market and develop optimal bidding strategy for market participants, in which GCN is employed to model the correlation between agents.
(2) A self-adaptive adjacency matrix is learned. It can be used independently in situations where the prior knowledge of system topology is unavailable, or integrated with the system topology in a GCN to further reveal and model the complex spatial correlations between agents.
(3) Historical information is incorporated and its temporal correlation is efficiently captured by a designed t-Linear module to enhance the agents' comprehension of system dynamics and mitigate system uncertainties.

The rest of the paper is organized as follows. Section 2 introduces the preliminaries of RL and GCN. Section 3 formulates the bidding model of market participants and the ISO market clearing model. Section 4 introduces the proposed DDPG-GCN method in detail. Section 5 presents case study and results. Section 6 provides the conclusion and discusses the future work.

2. Preliminaries

The proposed method are built on RL and GCN, so that we introduce them as preliminaries.

2.1. Reinforcement learning

The RL method aims to solve the Markov Decision Process (MDP) and maximize the total discounted reward. An action value function $Q_\pi(s_t, a_t)$ is usually defined in RL as an estimation of the total discounted reward as follows,

$$Q_\pi(s_t, a_t) = E_\pi\left(\sum_{k=0}^{N_T} \gamma^k r_{t+k+1} | s_t, a_t\right) \tag{1}$$

where E_π is the expectation, which starts from state s_t, takes action a_t, and thereafter follows policy π for a horizon with length N_T. γ and r represent the discount factor and reward, respectively. The goal of RL is to find the optimal policy π^* that maximizes the action-value function:

$$Q^*(s_t, a_t) = \max_\pi Q_\pi(s_t, a_t) \tag{2}$$

To solve the above problem, Q-learning is firstly introduced, which stores the action value function of all possible states and actions using a lookup table [31]. Due to the dimensionality explosion of Q-learning when dealing with continuous state or action, a set of deep RL methods have been developed to overcome the drawbacks of the tabular-based RL method. For example, the DQN method [33] combined the principles of Q-learning and DNN, which can deal with continuous state space. The DDPG method [34]proposed an actor–critic algorithm which can work with continuous action space and state space.

2.2. Graph convolutional neural networks

The GCN network was first proposed in [43] and has been developed into a number of variants in recent years [44]. Generally, GCN represents the data in the form of graph and employs the essential operation of graph convolution to extract nodes features according graph structure. A graph $G = (V, E)$ consists of a set of nodes V and a set of edges E, in which $v_i \in V$ denotes a node and $e_{ij} \in E$ denotes an edge pointing from v_i to v_j. An adjacency matrix A is used to describe the connection between nodes. A is usually a $N \times N$ matrix with $A_{ij} = 1$ if $e_{ij} \in E$ and $A_{ij} = 0$ if $e_{ij} \notin E$, where N is the total number of nodes. Besides, a graph can have node features $\mathbf{X} \in \mathbb{R}^{N \times D}$, where the ith row of the matrix $\mathbf{X}$ represents the feature vector of the node v_i. Graph convolution is the main operator of GCN to extract information of graph data owe to its efficiency and convenience for compositing with other network structures [44]. A number of models have been proposed to implement graph convolution [45]. For example, from the perspective of information diffusion, [46] formulates the graph convolution as a diffusion process with K steps as follows,

$$\mathbf{Z} = \sum_{k=0}^{K} \mathbf{P}^k \mathbf{X} \mathbf{W_k} \tag{3}$$

where $\mathbf{P}^k$ represents the power series of the transition matrix with $\mathbf{P} = A/rowsum(A)$. $\mathbf{W} \in \mathbb{R}^{D \times M}$ is the parameter matrix. $\mathbf{Z} \in \mathbb{R}^{N \times M}$ is the output signal.

3. Problem formulation

In this section, we first build the auction market environment: day-ahead centralized double-sided market with bidding submission from both supply side and demand side with elastic demand, which is commonly formulated as a bi-level optimization problem. The market participants primarily consist of two categories: the supply side involving generation companies and the demand side including electricity retailers or large consumers.

3.1. Bidding model of supply side

The objective of each generation company is to maximize its profits. The energy production costs of each GenCo is generally modeled as a quadratic function as follows,

$$C_g(p_{gt}) = \frac{1}{2}k_g p_{gt}^2 + h_g p_{gt} + f_g,\ g \in \mathcal{G} \tag{4}$$

where p_{gt} is the power generation at time interval t, $\mathcal{G}$ is the set of GenCos. k_g, h_g and f_g are the constant coefficients of the quadratic cost model. The marginal cost of GenCo g is therefore a linear function of the output power by calculating the derivative of $C_g(p_{gt})$:

$$\rho_g(p_{gt}) = k_g p_{gt} + h_g \tag{5}$$

The supply bidding function of GenCos can be formulated as a linear function:

$$\lambda_{gt}^{bid} = \alpha_{gt}\left(k_g p_{gt} + h_g\right) \tag{6}$$

where λ_{gt}^{bid} is the bid price ($/MWh) submitted to ISO by GenCo g at time interval t. α_{gt} is the strategic bidding variable of GenCo g at time interval t (the range is set as 0.9 to 2.0). The profits of GenCo can be formulated as Eq. (7)

$$r_{gt} = \lambda_{it} p_{gt} - C_g(p_{gt}),\ g \in \mathcal{G}_i \tag{7}$$

where λ_{it} is the nodal price at bus i and $g \in \mathcal{G}_i$ indicates that GenCo g is located at bus i.

3.2. Bidding model of demand side

The retailers are electricity selling companies that have elastic demand, which act as load or demand side in the system and purchase electricity and further sell it to end users or industrial customers. The demand curve of retailer is assumed to be linear function with negative slope. Thus, the benefit function of the retailers can be modeled as follows,

$$B_d(p_{dt}) = -\frac{1}{2}k_d p_{dt}^2 + h_d p_{dt} + f_d,\ d \in \mathcal{D} \tag{8}$$

where p_{dt} is the electricity quantity at time interval t, $\mathcal{D}$ is the set of retailers. The marginal benefit of retailer d is the derivative of $B_d(p_{dt})$:

$$\rho_d(p_{dt}) = -k_d p_{dt} + h_d \tag{9}$$

The bidding function of retailers can be formulated as:

$$\lambda_{dt}^{bid} = \alpha_{dt}\left(-k_d p_{dt} + h_d\right) \tag{10}$$

where λ_{dt}^{bid} is the bid price ($/MWh) submitted to ISO by retailer d at time interval t. α_{dt} is the strategic bidding variable of retailer d at time interval t (the range is set as 0.85 to 1.0). The profits of retailer can be formulated as follows,

$$r_{dt} = B_d(p_{dt}) - \lambda_{jt} p_{dt},\ d \in \mathcal{D}_j \tag{11}$$

where λ_{jt} is the nodal price at bus j, and $d \in \mathcal{D}_j$ indicates that retailer d is located at bus j.

3.3. Market clearing model of double-sided auction

Based on the bid submitted by GenCos and retailers, ISO derives the market clearing result by using the following market clearing model:

$$\max_{p_{gt},p_{dt}} \sum_{d\in D} \lambda_{dt}^{bid} p_{dt} - \sum_{g\in \mathcal{G}} \lambda_{gt}^{bid} p_{gt} \tag{12}$$

s.t.

$$\sum_{g\in \mathcal{G}} p_{gt} = \sum_{d\in D} p_{dt} \tag{13}$$

$$-\mathrm{LF}_{l,max} \le \sum_{i=1}^{n} \mathrm{PSF}_{l-i}\left(\sum_{g\in i} p_{gt} - \sum_{d\in i} p_{dt}\right) \le \mathrm{LF}_{l,max} \tag{14}$$

$$0 \le p_{gt} \le p_{gt}^{bid},\ \forall g \in \mathcal{G} \tag{15}$$

$$0 \le p_{dt} \le p_{dt}^{bid},\ \forall d \in \mathcal{D} \tag{16}$$

where Eq. (12) maximizes social welfare by considering both supply side and demand side. Eq. (13) is the power balance constraint. $\mathrm{LF}_{l,max}$ is the maximum capacity of transmission line l; n is the set of buses; PSF_{l-i} is power shift factor, which represents the power flow change on the transmission line l when one unit power injects at bus i. The transmission congestion can be managed by the locational marginal price of agent i at time interval t (denoted as $\lambda_{it}^{cleared}$) derived by the dual variables of Eq. (13) and (14). p_{gt}^{bid} and p_{dt}^{bid} are the bid quantities of GenCo g and retailer d at time step t. It is assumed that the bid quantities of participants are their maximum capacities p_g^{max} or p_d^{max}, and only the bid price can be changed [19–21,47,48].

Based on Eq. (4)–(16), a bi-level optimization problem is formed. For the lower level, the ISO tries to maximize the social welfare with the decision variables of p_{dt} and p_{gt}. For the upper level, the market participants try to maximize their own profit against the rivals by searching the space of λ_{gt}^{bid} or λ_{dt}^{bid}. The upper-level and lower-level problems are solved alternately. We set each market participant as a RL agent. Each RL agent solves its own upper-level optimization problem by DRL algorithm, and the lower-level ISO market clearing problem is solved after collecting all the bids from the market participants. For each RL agent i, we define:

(1) State s_{it}: considering that the bidding strategy and operational parameters (e.g., maximum capacity) are private information to other participants, we assume that only the historical nodal prices are available as the state variable for each agent ($T_h \ge 1$ is the historical order to consider more temporal information for decision-making)

$$s_{it} = \{\lambda_1, \lambda_2, \dots, \lambda_I\}_{t-T_h}^{t-1} \tag{17}$$

(2) Action a_{it}: the strategic bidding variable α_{gt} and α_{dt} defined in Eq. (6) and (10), respectively.

(3) Reward r_{it}: the profit r_{gt} and r_{dt} defined in Eq. (7) and (11), respectively.

4. Methodology

To solve the bi-level optimization problem in the double-sided auction electricity market, we propose the DDPG-GCN method, which combines the DDPG algorithm with GCN and constructs a self-adaptive adjacency matrix to capture potential correlation relation between market participants. The schematic of the proposed DDPG-GCN method is shown in Fig. 1. Overall, for the actor or critic network of an agent, the spatial information is modeled by the GCN module, and the temporal information is captured by a linear layer along the time dimension (t-Linear). Then, a feature fusion layer (FFL) is used to fuse all the spatial–temporal features. In this way, the spatial–temporal correlation between agents is modeled for reducing system uncertainty.

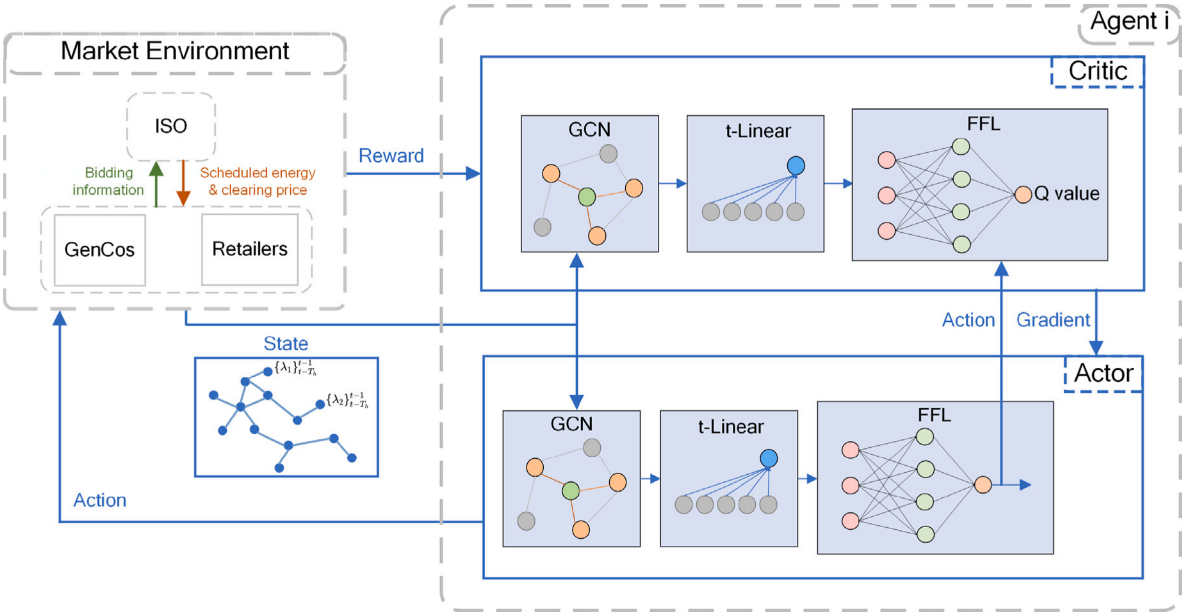


Fig. 1. Schematic of the proposed framework. Each market agent i comprises an actor and a critic network. The GCN module is incorporated into both the actor and critic networks to capture spatial correlations, forming the DDPG-GCN. Additionally, a t-Linear layer is also added to model temporal dependencies. The spatial–temporal features is then merged by the FFL module. Each agent trains its DDPG-GCN algorithm by interacting with the market environment continuously. The environment is updated based on the collective actions of all agents.

4.1. Graph convolution layer

The determination for the adjacency matrix is crucial for the performance of a GCN model. Most GCN networks require a predefined adjacency matrix based on prior knowledge. For instance, the physical and connectivity information of the road network structure is used to build the adjacency matrix for traffic flow forecasting [49]. Similarly, we consider using the power system topology to construct the graph of the GCN. That is, an undirected graph is constructed with $A_{ij} = 1$ if there is power branch connecting node i to node j, otherwise $A_{ij} = 0$. Meanwhile, we take the nodal price as the feature of a node to construct a nodal price graph, as the reward of an agent mainly depends on its nodal price. However, the adjacency matrix only based on the system topology is not enough to reveal the complicated correlation between the electricity price of nodes, because the market clearing nodal price is not only determined by the physical topology parameters but also the bid of all the market agents of supply side and demand side. Besides, the power system topology is not available in some scenarios due to privacy issues or market rule design. To tackle the above problems, inspired from the work in [50], we adopts the following self-adaptive adjacency matrix in our GCN network:

$$\tilde{A}_{adp} = \mathrm{Q}_2\left(\mathrm{Q}_1\left(E_1 E_2^T\right)\right) \tag{18}$$

where $E_1, E_2 \in \mathbb{R}^{N\times c}$ are the source and target node embeddings. Typically, $c < N$ to map the nodes into low-dimensional space [51]. The nodes embeddings are learnable and initialized randomly. Q_1 is the ReLu activation function, i.e.,

$$A_{adp} = \mathrm{Relu}\left(E_1 E_2^T\right) \tag{19}$$

Q_2 is a normalization function implemented by the following steps:

$$\begin{aligned} D_{adp_{ii}} &= \sum_j A_{adp_{ij}} \\ D_{adp1} &= \mathrm{diag}\left(1/\left(D_{adp_{ii}}\right)\right) \\ \tilde{A}_{adp} &= D_{adp1} A_{adp} \end{aligned} \tag{20}$$

we use Q_2 instead of softmax function to avoid full connection of all nodes and keep the sparsity of $\tilde{A}_{adp}$ [52].

After constructing the graph, we build a new diffusion graph convolution layer by integrating the graph of the physical system topology and the adaptive correlation graph of the nodes as follows,

$$\mathbf{Z} = \sum_{k=0}^{K} \left(\mathbf{P}^k \mathbf{X} \mathbf{W}_{1k} + \tilde{\mathbf{A}}_{adp}^k \mathbf{X} \mathbf{W}_{2k}\right) \tag{21}$$

where $\mathbf{W}_{1k}$ and $\mathbf{W}_{2k}$ are the parameters of GCN. When the system topology is not available, the above diffusion graph convolution layer will degrade only with the adaptive correlation graph as follows,

$$\mathbf{Z} = \sum_{k=0}^{K} \tilde{\mathbf{A}}_{adp}^k \mathbf{X} \mathbf{W}_{2k} \tag{22}$$

4.2. DDPG-GCN algorithm

DDPG is an actor–critic, model-free algorithm based on the deterministic policy gradient that can operate in continuous state and action space [34]. It combines the advantages of policy-based and value-based method. DDPG consists of two neural networks: the actor network and the critic network, which are used to approximate the policy function and action-value function, respectively. The actor makes actions and interacts with the environment, while the critic evaluates the performance of the actor and thus guides the learning process of actor. The actor network, denoted as $\mu(s|\theta^\mu)$, represents a policy function parameterized by θ^μ (referred to as network μ). The critic network, represented as $Q(s,a|\theta^Q)$, is an action-value function with parameters θ^Q (referred to as network Q). To facilitate the training process, each of these networks has a corresponding target network: the actor target network μ' with parameters $\theta^{\mu'}$, and the critic target network Q' with parameters $\theta^{Q'}$. For agent i, the loss of the network Q is determined as follows,

$$L_i\left(\theta^Q\right) = \frac{1}{m}\sum_j \left(Q_t^{\text{target }(j)} - Q(s_{it}^{(j)}, a_{it}^{(j)}|\theta^Q)\right)^2 \tag{23}$$

where j is the index of state–action pair samples. a_{it} denotes the action taken by agent i at time step t. m denotes the batch size. $Q_t^{\text{target }(j)}$ is defined as:

$$Q_t^{\text{target }(j)} = r_{it}^{(j)} + \gamma Q(s_{i,t+1}^{(j)}, a_{i,t+1}^{(j)}|\theta^{Q'}) \tag{24}$$

Table 2
System profile used in this paper. The format of parameters in the list is [GenCos; retailers]; A market participant is set as strategic RL agent if agent setting is 1.

Parameters	4-bus (4 GenCos and 4 retailers)	30-bus (6 GenCos and 3 retailers)	57-bus (7 GenCos and 3 retailers)
Bus	[1,2,3,4; 1,2,3,4]	[1,2,22,27,23,13; 7,15,30]	[1,2,3,6,8,9,12; 8,9,12]
p_i^{max} (MWh)	[200,250,300,300; 100,200,120,320]	[100,80,80,120,50,50; 80,120,100]	[576,100,140,100,550,100,300; 150,121,377]
k_i (\$/(MWh)^2)	[0.05,0.05,0.07,0.05; 0.10,0.10,0.10,0.10]	[0.25,0.20,0.20,0.25,0.20,0.20; 0.30,0.20,0.26]	[0.06,0.02,0.20,0.02,0.04,0.02,0.07; 0.2,0.2,0.1]
h_i (\$/MWh)	[18,20,21,17; 60,70,80,100]	[18,20,22,16,22,25; 70,80,75]	[20,40,20,40,20,40,20; 85,75,90]
Agent Setting	[1,1,1,1; 1,1,1,1]	[1,1,1,1,0,0; 0,1,1]	[1,1,1,1,1,1,1; 1,1,1]

Algorithm 1: DDPG-GCN for optimal bidding strategy

```
Initialize critic network $Q(s,a|\theta^Q)$ and actor network $\mu(o|\theta^\mu)$
  for each agent bidder
Initialize target network $Q'$ and $\mu'$ with $\theta^Q$ and $\theta^\mu$
Initialize minibatch $m$, replay buffer $\mathcal{R}$ and get the initial state
for $t$=1 to $T$ do
    Get noise $n_t$ from (29)
    For each agent $i$, select the action:
      $a_{it} = \min(\max(\mu_i(s_{it}|\theta^\mu) + n_t, -1), 1)$
    Get the bid variable $\alpha_{it}$ by (30)
    Market clearing using (12)-(16), obtain payoff and next
      observation for each agent
    for agent $i$=1 to $N$ do
        Store the transition $(s_{it}, a_{it}, r_{it}, s_{i,t+1})$
        Sample a random minibatch of m transitions
          $(s_{ij}, a_{ij}, r_{ij}, s_{i,j+1})$ from $\mathcal{R}$
        Update GCN critic network by (26)
        Update GCN actor network by (28)
        Update target network by: $\theta_i^{Q'} \leftarrow (1-\tau)\theta_i^Q + \tau\theta_i^{Q'}$
                                  $\theta_i^{\mu'} \leftarrow (1-\tau)\theta_i^\mu + \tau\theta_i^{\mu'}$
    end
end
```

where

$$a_{i,t+1}^{(j)} = \mu(s_{i,t+1}^{(j)}|\theta^{\mu'}) \tag{25}$$

Then the network Q is updated by gradient descent:

$$\theta_i^Q \leftarrow \theta_i^Q - \eta_Q \nabla L_i\left(\theta_i^Q\right) \tag{26}$$

where η_Q is the learning rate of network Q. The policy network μ is updated by maximizing the expected Q value:

$$\nabla J\left(\theta^\mu\right) \approx \frac{1}{m}\sum_j \nabla_a Q(s_{it}^{(j)}, a_{it}^{(j)}|\theta^Q)\nabla_{\theta^\mu}\mu(s_t^{(j)}|\theta^\mu) \tag{27}$$

$$\theta_i^\mu \leftarrow \theta_i^\mu + \eta_\mu \nabla J\left(\theta_i^\mu\right) \tag{28}$$

where the gradient is approximated using samples. To balance exploration and exploitation, a Gaussian noise $n_t \sim \mathcal{N}\left(0,\sigma_t^2\right)$ is added to the action value a_t with σ_t as follows,

$$\sigma_t = \max\left(0.5 \times 0.9995^t, 0.03\right) \tag{29}$$

where the minimum value of σ_t is set to 0.03 to preserve a certain level of exploration. When the action is taken by the actor network, a linear map is applied to transform the action value a_t to strategic bidding variable α_t:

$$\alpha_t = \alpha_{\min} + \frac{(a_t - a_{\min})(\alpha_{\max} - \alpha_{\min})}{a_{\max} - a_{\min}} \tag{30}$$

where [$\alpha_{\min}$, $\alpha_{\max}$] is the range of strategic variable, which is [0.9, 2] for GenCos or [0.85, 1] for loads. [$a_{\min}$, $a_{\max}$] is set to [−1, 1] because tanh function is applied as the activation function of the output layer of the actor network.

Commonly, the actor and critic networks of the DDPG discussed above are both composed of MLP. To propose DDPG-GCN, a graph convolution layer (Eq. (21) or (22)) is added. As illustrated in Fig. 1, the input spatial–temporal state graph is firstly forwarded to the GCN to model the spatial correlation between agents and nodes. Additionally, to extract the temporal dependencies, the output of the GCN is subsequently passed to the t-Linear layer. It is noteworthy that we utilize a simple yet effective linear layer along the time dimension to extract temporal features rather than Recurrent Neural Networks (RNNs) like Gate Recurrent Unit (GRU) or Temporal Convolutional Networks (TCN), as we found increased computational burden but little improvement compared to t-Linear. A potential reason for this could be that the long-term historical correlation of nodal prices is weak, or that its long-term correlation has no great impact on formulating bidding strategies. Finally, a FFL module is applied to fuse the spatial–temporal features to output the final action for actor network or action value for critic network. This enables agents to effectively adapt to the dynamic and uncertain market transaction process. The detailed implementation of the proposed algorithm is provided in Algorithm 1.

5. Experiments

In this section, we evaluate the proposed method on different network topologies and analyze the experimental results.

5.1. Experiment setting

Two types of popular RL algorithm, DDPG (actor–critic) and DQN (value-based), are used to simulate the multi-agent system and validate the proposed algorithm. Considering that power systems of different scales may influence the performance of GCN, three networks with different topologies, i.e., 4-bus system [13], 30-bus system [53] and IEEE 57-bus systems are used to evaluate the proposed method. The profile and setting of the three systems are listed in Table 2.

The output feature dimension of GCN network is 4, 1, 4 for 4-bus, 30-bus and 57-bus system, respectively. The diffusion step K is set to 1. The node embedding dimension c for 4-bus, 30-bus and 57-bus system is 4, 10, 20, respectively. The output dimension of the t-Linear is set to 1. The FFL in the actor network is a single output linear layer; The FFL in the critic is a single hidden layer fully connected neural network with hidden dimension set as 32, 64 and 128 for 4-bus, 30-bus and 57-bus system, respectively. The settings of other hyper-parameters are provided in Table A.8 in Appendix. Similarly, for DQN, we add a GCN module in the value network of DQN. The network of DQN is 1-layer GCN followed by a single hidden layer fully connected neural network. The action resolution, learning rate, memory capacity of DQN is set to 0.005, 0.001, 1000. The historical order T_h is set to 3. Each algorithm is repeated for 6 experiments with different random seeds. An episode includes 24 steps, each step denotes one hour.

The neural network is built by the deep learning platform of PyTorch, and the market clearing model is built based on the toolbox MATPOWER [54]. The experiment is conducted in a hardware environment of AMD Ryzen 5800H 3.2 GHz CPU and 16 GB RAM. The code for this study is available at https://github.com/18heweng666/DDPG-GCN-for-bidding-strategy.git.

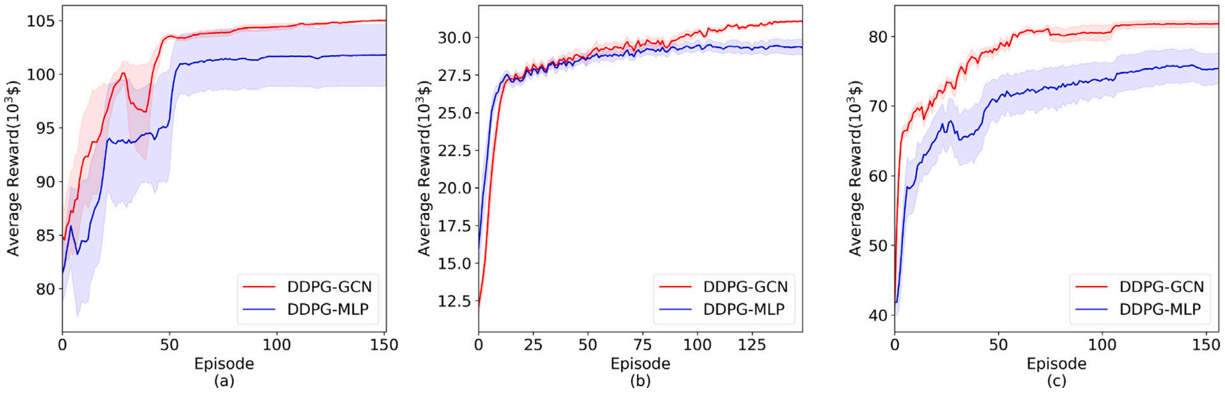


Fig. 2. Comparison of the average profits of agents based on DDPG. (a) 4 bus. (b) 30 bus. (c) 57 bus.

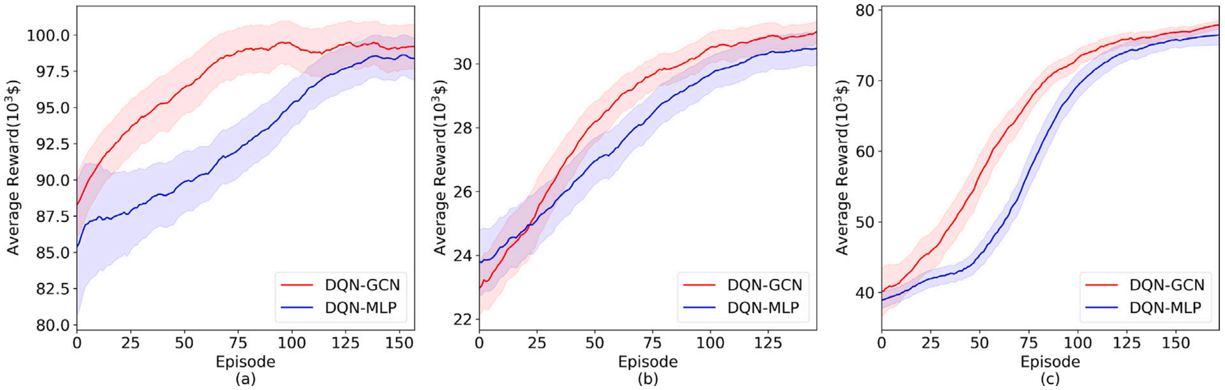


Fig. 3. Comparison of the average profits of agents based on DQN. (a) 4 bus. (b) 30 bus. (c) 57 bus.

Table 3
Average test episode profit and computational cost of different algorithms across various systems. "*" denotes the method from [21] extended in our market framework.

Model	Profits (10^3\$)/training time (s)/inference time per ep. (s)		
	4-bus	30-bus	57-bus
DDPG-MLP	15859/188/0.06	5277/201/0.05	15187/261/0.07
DDPG-GCN-ada	16171/373/0.13	5431/359/0.14	15891/737/0.19
DDPG-GCN-pri*	16409/291/0.09	5431/288/0.09	16053/631/0.13
DDPG-GCN-ada+pri	**16603**/371/0.13	**5592**/376/0.16	**16215**/777/0.19
DQN-MLP	15620/169/0.02	5426/177/0.02	15543/254/0.03
DQN-GCN-ada	15522/182/0.06	5490/206/0.05	15607/351/0.10
DQN-GCN-pri*	**15749**/174/0.04	5510/199/0.03	15685/328/0.06
DQN-GCN-ada+pri	15724/197/0.07	**5603**/215/0.06	**15770**/383/0.11

5.2. Experimental results

The experimental results are shown in Fig. 2 for DDPG and Fig. 3 for DQN. To display the figures clearly, appropriate smoothing of the results is applied. The shaded area in the figures is the half of the standard deviation over all experiments.

For both DDPG and DQN, it demonstrates that GCN performs better than traditional MLP-based DRL in all the three systems with different scales, even for the small graph of 4-bus system. For DDPG, GCN gains not only the final average profits, but also the stability and convergence of training with low fluctuation of the curve. It is considered resulting from the advantage of GCN that aggregates the information of neighbor nodes and reduces the system uncertainty. Similar results can be observed for DQN, which means that GCN can be integrated with different types of DRL algorithms. Using the trained models, we test for 20 episodes to get the average episode profit, as presented in Table 3. "ada" and "pri" denote using only the adaptive adjacency matrix or the prior system topology matrix, respectively. The results are analyzed in detail from the following perspectives.

(1) **DDPG vs. DQN:** When GCN combined with DDPG and DQN, i.e., DDPG-GCN and DQN-GCN, there are both improvements. It is shown that the overall performance of DDPG-GCN is superior to that of DQN-GCN. This superiority can be attributed to the fact that GCN can be embedded into two sub-networks of DDPG for enhancing the algorithm's spatial information modeling capabilities, unlike the single value network of DQN.
(2) **GCN vs. MLP:** GCN performs better than MLP in different situations of adjacency matrices selection for both two algorithms. For example, after using GCN (GCN-ada+pri), the test average profits of DDPG increase 6.8%, 6.0% and 4.7% for 57-bus, 30-bus and 4-bus system, respectively. As system networks become larger and more complex, the profits of participants will be influenced by other nodes on a larger scale, thereby facing greater uncertainty and posing challenge for MLP-based RL. In this context, GCN can model the information correlation of participants and nodes in systems of different scales by aggregating neighbor information, especially with more outstanding performance compared to MLP as the scale increases.

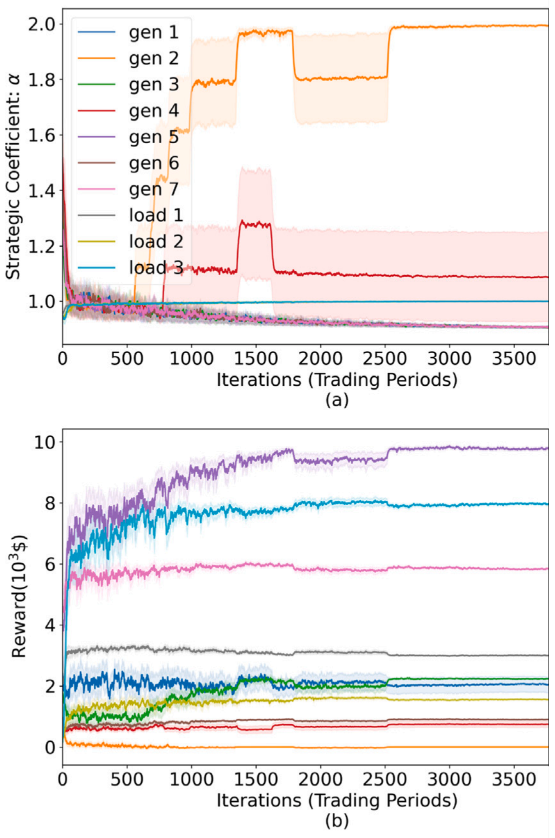


Fig. 4. Training process of the agents based on DDPG-GCN: 57-bus. (a) strategic variable. (b) reward.

(3) **Prior knowledge vs. Self-adaption:** It is shown that the test average profits of DDPG based on self-adaptive matrix (GCN-ada) increase by 4.6%, 2.9% and 2.0% for the 57-bus, 30-bus and 4-bus system, respectively. This demonstrates that in the case where the prior adjacency matrix is not available, using the adaptively learned adjacency matrix can still improve system revenue. This improvement becomes more pronounced when the system scale is large with sufficient structural information. However, the prior knowledge adjacency matrix constructed using the system topology outperforms the adaptively trained matrix, which highlights the importance of system topology information for the optimal bidding strategy. Additionally, the combination of the adaptive adjacency matrix and the prior one can bring further performance improvement to the system, which means that relying solely on the topology information of the system cannot fully model the correlation between nodal prices, while combining adaptive adjacency matrix can supplement the bidding information among market participants, effectively alleviating this problem.

The training process of RL agents' bidding strategic variables and the agents' rewards for the 57-bus system based on DDPG-GCN are visualized in Fig. 4. The figure shows that the rewards for most of the agents converge successfully. Additionally, it can be observed that multiple different bids of an agent may converge to the same reward, as seen with GenCo 4.

The adaptive adjacency matrix of 57-bus system learned by GCN is visualized in Fig. 5. It is shown that the critic network remains more sparsity than the actor network and it focuses mainly on some nodes, such as node index 0–2, 18–20. This indicates that a smaller amount of neighbor's information is sufficient to evaluate the value of actions, while information from a wider range of sources is needed to make an bidding action. This also helps explain why DQN-GCN, which only embeds GCN into the value network for estimating the action value, is not as effective as DDPG-GCN.

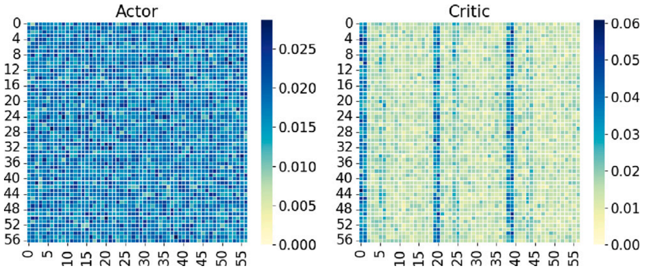


Fig. 5. Average adaptive adjacency matrix of all agents of 6 different experiments: 57-bus. (a) actor network. (b) critic network.

Table 4
Average episode profit for each participant using different algorithms to reveal rivals' behaviors (4-bus)

Participant	Profits		
	Method [12]	Proposed	MPEC
GenCo 1	544	1214	1282
GenCo 2	0	1376	1376
GenCo 3	135	94	157
GenCo 4	1815	1848	2114
Retailer 1	860	860	860
Retailer 2	2098	2098	2098
Retailer 3	2059	2059	2059
Retailer 4	7054	7054	7054

5.3. Evaluation of modeling capability for bidding behavior

To evaluate the proposed method's capability in modeling competitors' bidding behaviors, we compare it with two other behavioral modeling algorithms: one that estimates competitors' behaviors using the concept of DER [12], and the traditional MPEC based on full knowledge of rivals' bids. Here, we first train the system with all participants employing the proposed DDPG-GCN strategy. Then, for each participant, its strategy is replaced by other two algorithms and its profit under 20 episodes in 4-bus system is recorded. The results are shown in Table 4.

Unsurprisingly, the results show that the MPEC based on the exact knowledge of the other participants' bids can achieve the highest profits for all participants. For all the retailers in this case, both DER-based method and DDPG-GCN can achieve the highest profit. For GenCos, the proposed DDPG-GCN shows better performance than DER-based method with higher profits for GenCo 1, GenCo 2 and GenCo 4, especially for GenCo 2. Although the DER-based method is effective in most scenarios, we found that in certain scenarios, the estimated electricity prices and cleared quantities using the DER-based method may deviate greatly from the true values. For instance, in 'gap areas' where sudden fluctuations in cleared quantities occur due to changes in bid prices. The results demonstrate that the proposed method can effectively model competitors' bidding behaviors by capturing the spatial–temporal correlation between participants and system nodes.

5.4. Evaluation on real world data

To verify the generalization performance of the proposed algorithm, the above fixed p_d^{max} is replaced by time-varying demand to give retailers more flexibility. We select the hourly load data obtained from PJM [55], ranging form June 1st, 2021, to January 31st, 2022, lasting

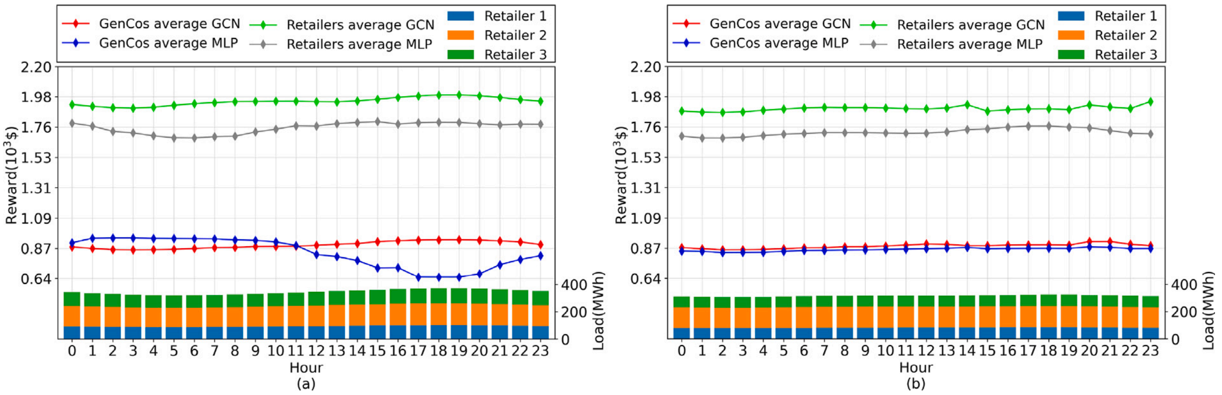


Fig. 6. Average profits evaluated on 30-bus system with real world data. (a) summer scenario. (b) winter scenario.

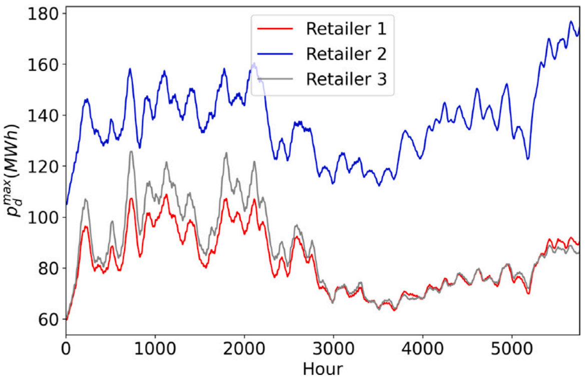


Fig. 7. Training load curve for 30-bus system (smoothed).

Table 5
The average total profit of DDPG for an episode in winter and summer scenario for 30-bus system.

Scenario	GenCos			Retailers		
	MLP	GCN	Increase (%)	MLP	GCN	Increase(%)
Summer	0.83	0.87	4.8	1.75	1.97	12.6
Winter	0.85	0.88	3.5	1.71	1.89	11.0

245 days. The data in January and July, 2023 is used as test data for winter and summer scenarios, respectively. The load curves are scaled appropriately and shown in Fig. 7. Here we use DDPG and 30-bus system for evaluation. The hourly test average profits is shown in Fig. 6, and the details is listed in Table 5. It can be seen that supply side and demand side gain more profits for 7.1%, 11.1% in summer (9.8% in total), respectively, and 2.9%, 10.1% in winter (7.7% in total), respectively, which indicates that under dynamic load conditions, GCN has better generalization ability than MLP. This phenomenon is particularly evident in summer, where load demand has greater volatility and uncertainty compared to winter. In this case, the proposed DDPG-GCN can deal with system uncertainty and bring more improvements to system profits.

5.5. The impact of historical order

To show the impact of different orders of historical information on system profits, DDPG-GCN and DDPG-MLP are tested on 57-bus system with different orders. The results are presented in Table 6. It is observed that incorporating more historical information into the model leads to 2.1% and 1.5% increase in system profits for DDPG-GCN and DDPG-MLP when the order is 3 and 2, respectively. It demonstrates that the historical information can alleviate the adverse effects of partial observation and is helpful for improving system profits. Furthermore, it is reasonable to expect that as the degree of uncertainty or local observability increases in the market, integrating historical information into the model may enhance the stability of the system.

Table 6
The impact of historical order (57-bus)

Method	Order				
	1	2	3	4	5
DDPG-MLP	15 048	15 273	15 187	15 170	14 327
DDPG-GCN	15 888	15 662	16 215	15 989	15 888

Table 7
Average episode profits of the DDPG-GCN and DDPG-MLP for 57-bus system with disconnected branches.

Disconnected lines number	DDPG-MLP	DDPG-GCN	Increase(%)
0	15 187	16 215	6.77
1	14 444	15 141	4.83
3	14 482	15 722	8.56
5	14 706	15 478	5.25
8	14 986	16 212	8.18
10	15 011	16 212	8.00

5.6. Robustness assessment with line outage

To further evaluate the robustness and generalization ability of GCN, we consider the grid topology adaptivity of the proposed algorithm by simulating system contingency. Concretely, we test DDPG with randomly disconnecting some power branches in 57-bus system. Here, different number of branches are disconnected, and three experiments are conducted using different random seeds to get the average test episode profits. Note that we use the trained model for testing without re-training; therefore, this will not result in a notable change in computation time. The result of average test episode profits of agents is listed in Table 7. It is shown that performs better than MLP with different number of disconnected branches, which confirms that GCN improves the generalization ability of algorithms with better adaptiveness to new topology compared to MLP-based DRL.

6. Conclusion

In this paper, a novel DDPG algorithm combined with GCN is proposed to improve the profit of agents by modeling the correlation between nodes in centralized day-ahead double-sided auction market, in which both supply side and demand side are modeled as RL agents. The proposed algorithm is evaluated using three different systems—4-bus, 30-bus, and 57-bus, yielding profit improvements of 4.7%, 6.0%, and 6.8%, respectively. Additionally, incorporating temporal information also enhances the agents' understanding of the environment, resulting in a profit increase of 1.5% to 2.1%. The results show that the proposed algorithm can enhance the strategic bidding ability of agents in the complicated double-sided auction market and serve as an effective tool to deal with system uncertainty and increase market profits. Meanwhile, DQN has also been tested, confirming that GCN can be combined with other types of DRL algorithms.

Our analysis focused on a specific, yet representative auction market, highlighting the significant potential in jointly applying methods from GCN and RL-based multi-agent systems across various transaction mechanisms and market frameworks, such as multi-energy and multi-entity markets. We hope that the application demonstrated in this work underscores the importance of integrating these two technologies in future research. Additionally, it would be interesting to explore more suitable methods for constructing the graph and to apply more advanced GCN models in these contexts.

CRediT authorship contribution statement

Haoen Weng: Writing – original draft, Methodology, Conceptualization. **Yongli Hu:** Writing – review & editing, Validation. **Min Liang:** Visualization. **Jiayang Xi:** Data curation. **Baocai Yin:** Validation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research is supported by the National Natural Science Foundation of China under Grant No. U21B2038 and U19B2039, National Key R and D Program of China (No. 2021ZD0111902), and R&D Program of Beijing Municipal Education Commission, China (KZ202210005008).

Appendix. An illustrative example of DDPG-GCN

Here is an illustrative example of 57-bus system using the proposed DDPG-GCN approach.

- Step 1: Initialize the GCN critic and GCN actor networks with their target networks for each RL agent. The detailed settings of the hyper-parameters are summarized in Table A.8. Get the random initial state s: [20.8, 58.8, 53.3, 28.5, 27.3, 27.3, 32.2, …] (which contains the historical electricity prices of all nodes).
- Step 2: Use the actor network to select an action: −0.01, add random Gaussian noise by Eq. (29) and clip the action to [−1, 1]: −0.19. Map the clipped action to the decision range to obtain agent's strategic bidding variable: 1.35.
- Step 3: Repeat Step 2 until all RL agents have completed. Then collect all the strategic bidding variables of RL agents: [1.35, 1.72, 1.61, 1.62, 1.88, 2, 1.35; 0.93, 0.87, 0.97]. Calculate the actual bid prices of all agents by Eqs. (6) and (10): [74.4, 72.2, 77.4, 68.1, 83.5, 84, 55.6; 51.4, 44.4, 50.7].
- Step 4: Run ISO market clearing model using MATPOWER to obtain nodal price and calculate profits for each RL agents. Nodal prices: [74.4, 74.4, 74.4, …] (57 nodes in total); Agent quantities: [102, 100, 0, 100, 0, 0, 300; 0, 0, 0]; Agent profits: [5271, 3342, 0, 3342, 0, 0, 13152; 0, 0, 0]. Then use profit as the reward for agents and use the nodal price to update the next state.
- Step 5: Store the transition, and randomly sample a batch of transitions to update the GCN actor and GCN critic networks for each RL agent
- Step 6: Repeat Step 2 to 5 until the training of an experiment is complete.

Table A.8
Hyper-parameter settings.

Parameter	Value
Batch size	128
Replay buffer size	1000
Learning rate (actor)	0.0003
Learning rate (critic)	0.0003
Discount fator	0.9
Soft update rate	0.01
Number of episodes	200
Episode length	24
Historical order of state	3

Data availability

Data will be made available on request.

References

[1] Shen J-j, Cheng C-t, Jia Z-b, Zhang Y, Lv Q, Cai H-x, et al. Impacts, challenges and suggestions of the electricity market for hydro-dominated power systems in China. Renew Energy 2022;187:743–59.

[2] Li J, Li Z. Multi-market bidding strategy considering probabilistic real time ancillary service deployment. In: 2016 IEEE electrical power and energy conference. 2016, p. 1–8. http://dx.doi.org/10.1109/EPEC.2016.7771714.

[3] Zhang G, Zhang G, Gao Y, Lu J. Competitive strategic bidding optimization in electricity markets using bilevel programming and swarm technique. IEEE Trans Ind Electron 2010;58(6):2138–46.

[4] Zaman F, Elsayed SM, Ray T, Sarker RA. Co-evolutionary approach for strategic bidding in competitive electricity markets. Appl Soft Comput 2017;51:1–22.

[5] Chen X, Xie J. Optimal bidding strategies for load server entities in double-sided auction electricity markets with risk management. In: 2006 international conference on probabilistic methods applied to power systems. IEEE; 2006, p. 1–6.

[6] Sharifi R, Anvari-Moghaddam A, Fathi SH, Vahidinasab V. A bi-level model for strategic bidding of a price-maker retailer with flexible demands in day-ahead electricity market. Int J Electr Power Energy Syst 2020;121:106065.

[7] Baringo L, Conejo AJ. Offering strategy of wind-power producer: A multi-stage risk-constrained approach. IEEE Trans Power Syst 2015;31(2):1420–9.

[8] Hong Q, Meng F, Liu J, Bo R. A bilevel game-theoretic decision-making framework for strategic retailers in both local and wholesale electricity markets. Appl Energy 2023;330:120311.

[9] Chen S, Conejo AJ, Sioshansi R, Wei Z. Equilibria in electricity and natural gas markets with strategic offers and bids. IEEE Trans Power Syst 2019;35(3):1956–66.

[10] Moiseeva E, Hesamzadeh MR. Bayesian and robust nash equilibria in hydrodominated systems under uncertainty. IEEE Trans Sustain Energy 2017;9(2):818–30.

[11] Kiannejad M, Salehizadeh MR, Oloomi-Buygi M, Shafie-khah M. Artificial neural network approach for revealing market competitors' behaviour. IET Gener Transm Distrib 2020;14(7):1292–7.

[12] Kiannejad M, Salehizadeh MR, Oloomi-Buygi M. Two-stage ANN-based bidding strategy for a load aggregator using decentralized equivalent rival concept. IET Gener Transm Distrib 2021;15(1):56–70.

[13] Namalomba E, Feihu H, Shi H. Agent based simulation of centralized electricity transaction market using bi-level and Q-learning algorithm approach. Int J Electr Power Energy Syst 2022;134:107415.

[14] Xu H, Sun H, Nikovski D, Kitamura S, Mori K, Hashimoto H. Deep reinforcement learning for joint bidding and pricing of load serving entity. IEEE Trans Smart Grid 2019;10(6):6366–75.

[15] Ren K, Liu J, Liu X, Nie Y. Reinforcement learning-based bi-level strategic bidding model of gas-fired unit in integrated electricity and natural gas markets preventing market manipulation. Appl Energy 2023;336:120813.

[16] Liang Y, Guo C, Ding Z, Hua H. Agent-based modeling in electricity market using deep deterministic policy gradient algorithm. IEEE Trans Power Syst 2020;35(6):4180–92.

[17] Chiu W-Y, Hu C-W, Chiu K-Y. Renewable energy bidding strategies using multiagent q-learning in double-sided auctions. IEEE Syst J 2021;16(1):985–96.

[18] Fang X, Zhao Q, Wang J, Han Y, Li Y. Multi-agent deep reinforcement learning for distributed energy management and strategy optimization of microgrid market. Sustain Cities Soc 2021;74:103163.

[19] Du Y, Li F, Zandi H, Xue Y. Approximating nash equilibrium in day-ahead electricity market bidding with multi-agent deep reinforcement learning. J Mod Power Syst Clean Energy 2021;9(3):534–44.

[20] Yin B, Weng H, Hu Y, Xi J, Ding P, Liu J. Multi-agent deep reinforcement learning for simulating centralized double-sided auction electricity market. IEEE Trans Power Syst 2024.

[21] Rokhforoz P, Montazeri M, Fink O. Multi-agent reinforcement learning with graph convolutional neural networks for optimal bidding strategies of generation units in electricity markets. Expert Syst Appl 2023;225:120010.

[22] Yu L, Wang P, Chen Z, Li D, Li N, Cherkaoui R. Finding Nash equilibrium based on reinforcement learning for bidding strategy and distributed algorithm for ISO in imperfect electricity market. Appl Energy 2023;350:121704.

[23] Li T, Shahidehpour M. Strategic bidding of transmission-constrained GENCOs with incomplete information. IEEE Trans Power Syst 2005;20(1):437–47.

[24] Chen B, Che Y, Zheng Z, Zhao S. Multi-objective robust optimal bidding strategy for a data center operator based on bi-level optimization. Energy 2023;269:126761.

[25] Mitridati L, Pinson P. A Bayesian inference approach to unveil supply curves in electricity markets. IEEE Trans Power Syst 2017;33(3):2610–20.

[26] Sensfuß F, Ragwitz M, Genoese M, Möst D. Agent-based simulation of electricity markets. A literature review. 2007.

[27] McArthur SD, Davidson EM, Catterson VM, Dimeas AL, Hatziargyriou ND, Ponci F, et al. Multi-agent systems for power engineering applications—Part I: Concepts, approaches, and technical challenges. IEEE Trans Power Syst 2007;22(4):1743–52.

[28] Vale Z, Pinto T, Praca I, Morais H. MASCEM: electricity markets simulation with strategic agents. IEEE Intell Syst 2011;26(2):9–17.

[29] Thrun S, Littman ML. Reinforcement learning: an introduction. AI Mag 2000;21(1). 103–103.

[30] Roth AE, Erev I. Learning in extensive-form games: Experimental data and simple dynamic models in the intermediate term. Games Econ Behav 1995;8(1):164–212.

[31] Watkins CJ, Dayan P. Q-learning. Mach Learn 1992;8:279–92.

[32] Glavic M, Fonteneau R, Ernst D. Reinforcement learning for electric power system decision and control: Past considerations and perspectives. IFAC-PapersOnLine 2017;50(1):6918–27.

[33] Mnih V, Kavukcuoglu K, Silver D, Graves A, Antonoglou I, Wierstra D, et al. Playing atari with deep reinforcement learning. 2013, arXiv preprint arXiv:1312.5602.

[34] Lillicrap TP, Hunt JJ, Pritzel A, Heess N, Erez T, Tassa Y, et al. Continuous control with deep reinforcement learning. 2015, arXiv preprint arXiv:1509.02971.

[35] Silver D, Lever G, Heess N, Degris T, Wierstra D, Riedmiller M. Deterministic policy gradient algorithms. In: International conference on machine learning. Pmlr; 2014, p. 387–95.

[36] Zhang K, Yang Z, Başar T. Multi-agent reinforcement learning: A selective overview of theories and algorithms. In: Handbook of reinforcement learning and control. Springer; 2021, p. 321–84.

[37] Lowe R, Wu YI, Tamar A, Harb J, Pieter Abbeel O, Mordatch I. Multi-agent actor-critic for mixed cooperative-competitive environments. In: Advances in neural information processing systems, vol. 30, 2017.

[38] Zhu Z, Hu Z, Chan KW, Bu S, Zhou B, Xia S. Reinforcement learning in deregulated energy market: A comprehensive review. Appl Energy 2023;329:120212.

[39] Lin W, Wu D, Boulet B. Spatial-temporal residential short-term load forecasting via graph neural networks. IEEE Trans Smart Grid 2021;12(6):5373–84.

[40] Yang Y, Tan Z, Yang H, Ruan G, Zhong H, Liu F. Short-term electricity price forecasting based on graph convolution network and attention mechanism. IET Renew Power Gener 2022;16(12):2481–92.

[41] Liu S, Wu C, Zhu H. Topology-aware graph neural networks for learning feasible and adaptive AC-OPF solutions. IEEE Trans Power Syst 2022.

[42] Amato C, Chowdhary G, Geramifard A, Üre NK, Kochenderfer MJ. Decentralized control of partially observable Markov decision processes. In: 52nd IEEE conference on decision and control. IEEE; 2013, p. 2398–405.

[43] Scarselli F, Gori M, Tsoi AC, Hagenbuchner M, Monfardini G. The graph neural network model. IEEE Trans Neural Netw 2008;20(1):61–80.

[44] Wu Z, Pan S, Chen F, Long G, Zhang C, Philip SY. A comprehensive survey on graph neural networks. IEEE Trans Neural Netw Learn Syst 2020;32(1):4–24.

[45] Zhou J, Cui G, Hu S, Zhang Z, Yang C, Liu Z, et al. Graph neural networks: A review of methods and applications. AI Open 2020;1:57–81.

[46] Atwood J, Towsley D. Diffusion-convolutional neural networks. In: Advances in neural information processing systems, vol. 29, 2016.

[47] Pereira MV, Granville S, Fampa MH, Dix R, Barroso LA. Strategic bidding under uncertainty: a binary expansion approach. IEEE Trans Power Syst 2005;20(1):180–8.

[48] Niu H, Baldick R, Zhu G. Supply function equilibrium bidding strategies with fixed forward contracts. IEEE Trans Power Syst 2005;20(4):1859–67.

[49] Geng X, Li Y, Wang L, Zhang L, Yang Q, Ye J, et al. Spatiotemporal multi-graph convolution network for ride-hailing demand forecasting. In: Proceedings of the AAAI conference on artificial intelligence, vol. 33, (01):2019, p. 3656–63.

[50] Wu Z, Pan S, Long G, Jiang J, Zhang C. Graph WaveNet for deep spatial-temporal graph modeling. In: International joint conference on artificial intelligence.

[51] Goyal P, Ferrara E. Graph embedding techniques, applications, and performance: A survey. Knowl-Based Syst 2018;151:78–94.

[52] Guo K, Hu Y, Sun Y, Qian S, Gao J, Yin B. Hierarchical graph convolution network for traffic forecasting. In: Proceedings of the AAAI conference on artificial intelligence, vol. 35, (1):2021, p. 151–9.

[53] Alsac O, Stott B. Optimal load flow with steady-state security. IEEE Trans Power Appar Syst 1974;(3):745–51.

[54] Zimmerman RD, Murillo-Sánchez CE, Thomas RJ. MATPOWER: Steady-state operations, planning, and analysis tools for power systems research and education. IEEE Trans Power Syst 2011;26(1):12–9. http://dx.doi.org/10.1109/TPWRS.2010.2051168.

[55] PJM market data, [*Online*]. Available: https://www.pjm.com/.