frontiers | Frontiers in Energy Research

TYPE Original Research
PUBLISHED 08 January 2024
DOI 10.3389/fenrg.2023.1324232

OPEN ACCESS

EDITED BY
Hengrui Ma,
Qinghai University, China

REVIEWED BY
Yixin Liu,
Tianjin University, China
Jun Li,
Nanjing Institute of Technology (NJIT), China
Chenhao Sun,
Changsha University of Science and Technology, China

*CORRESPONDENCE
Zhenbing Zhao,
zhaozhenbing@ncepu.edu.cn

RECEIVED 19 October 2023
ACCEPTED 19 December 2023
PUBLISHED 08 January 2024

CITATION
Sun S, Guo W, Wang Q, Tao P, Li G and Zhao Z (2024), Optimal scheduling of microgrids considering real power losses of grid-connected microgrid systems. *Front. Energy Res.* 11:1324232. doi: 10.3389/fenrg.2023.1324232

Optimal scheduling of microgrids considering real power losses of grid-connected microgrid systems

Shengbo Sun[1], Wei Guo[1], Qiwei Wang[2], Peng Tao[2], Gang Li[2] and Zhenbing Zhao[2]*

[1]State Grid Hebei Marketing Service Center, Shijiazhuang, Hebei, China, [2]North China Electric Power University, Baoding, Hebei, China

Energy conservation, emission reduction and vigorous development of new energy are inevitable trends in the development of the power industry, but factors such as energy storage loss, solar energy loss and line loss in real power situations have led the problem to a complex direction. To address these intricacies, we use a more precise modeling approach of power loss and propose a collaborative optimization method integrating the Deep-Q-Network (DQN) algorithm with the multi-head attention mechanism. This algorithm calculates weighted features of the system's states to compute the Q-values and priorities for determining the next operational directives of the energy system. Through extensive simulations that replicate real world microgrid (MG) scenarios, our investigation substantiates that the optimization methodology presented here effectively governs the distribution of energy resources. It accomplishes this while accommodating uncertainty-induced losses, ultimately achieving the economic optimization of MG. This research provides a new approach to deal with problems such as energy loss, which is expected to improve economic efficiency and sustainability in areas such as microgrids.

KEYWORDS

microgrid, energy management, deep reinforcement learning (deep RL), real power loss, attention mechanism (AM)

1 Introduction

1.1 Background and related works

With the exacerbating energy crisis and environmental pollution, solar and wind energy have played an increasingly vital role as distributed energy resources due to their abundant and pollution-free nature. However, solar and wind energy are random and intermittent, posing difficulties for grid integration and dispatch. Microgrids have emerged as an effective solution to facilitate the comprehensive utilization of renewable energy (Zhang and Kang, 2022). Microgrids show enormous potential in resolving renewable energy integration thanks to their flexible operation and ease of control. Their efficient and cost-effective operation is a prerequisite for sustainable development. Nevertheless, the multi-source characteristic of renewable energy sources introduces complexity to the control problem in microgrid systems. Based on recent surveys, it has been observed that as much as 13% of the total generated power is dissipated as losses at the distribution level (Wu et al., 2010; Patel

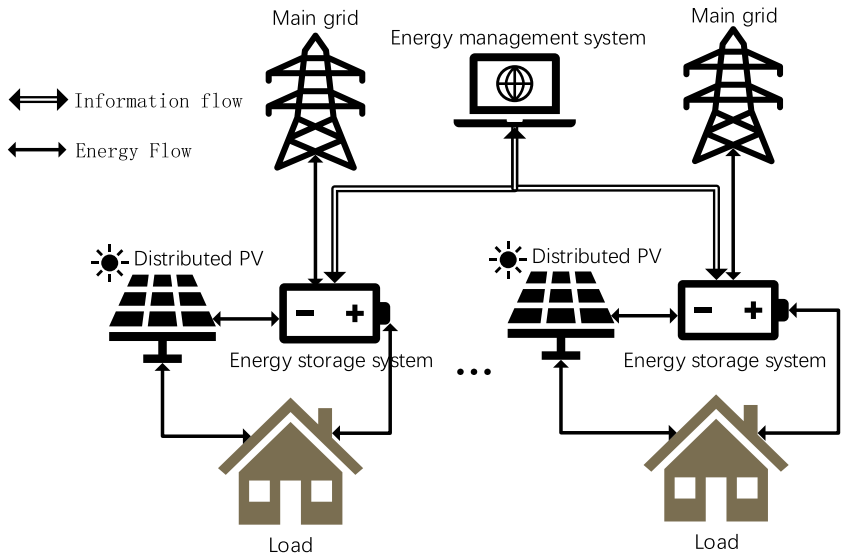

FIGURE 1
MG system structure diagram.

and Patel, 2016) applied ant colony optimization (ACO) to the reconfiguration of microgrids with distributed generation (DG) in order to minimize power losses (Kumari et al., 2017). introduced a particle swarm optimization (PSO) approach aimed at reducing DG costs and enhancing the voltage profile while addressing power loss concerns. Both of the aforementioned methods ascertain the optimal placement of DG using optimization algorithms. However, they do not account for the distinction between linear and nonlinear loads in their calculations. On the basis of this problem, this paper proposes a more accurate model of the actual line loss.

Energy system scheduling for microgrids has been investigated in a number of previous studies. Numerous studies utilize model-based control paradigms, including model predictive control (MPC) (Gan et al., 2020), mixed-integer linear programming (MILP) (Paterakis et al., 2015), dynamic and stochastic programming (Farzaneh et al., 2019), and alternating direction method of multiplier (ADMM) (Ma et al., 2018). However, once a large number of DERs connected to the MG in a disorderly way, the operation of the power grid will be largely influenced by its randomness and uncertainty. This makes it difficult to obtain the accurate system model. To solve these challenges, a model-free technique using reinforcement learning (RL) has been proven beneficial for energy system scheduling since the model of the environment is not necessary in this method. It is now emerging as the pre-eminent tool for unknown environmental decision-making issues. The authors of (Kim et al., 2016) present an RL algorithm that enables service providers and customers to acquire pricing and energy consumption strategies without any prior knowledge, thus reducing system costs (Fang et al., 2020). explored a dynamic RL-based pricing scheme to attain optimal prices when dealing with fast-charging electric vehicles connected to the grid. To reduce the electricity bills of residential consumers, a model for load scheduling using RL was developed in the literature (Lee and Choi, 2022), where the residential load includes dispatches-available load, non-dispatches-available load, and local PV generation. In recent research findings, to address the dynamically changing operational conditions of appliances, a federated DQN approach has been proposed for managing energy in multiple homes (Remani et al., 2019). This research showcased exceptional performance of the DQN method in addressing continuous state space energy management challenges. Nevertheless, in MG scenarios, the performance of the DQN model in energy scheduling is significantly compromised by the inherent uncertainty of renewable energy sources. Furthermore, there is currently no well-defined strategy in place to address the complex issue of multivariate losses.

1.2 Contributions

To overcome the aforementioned challenges, this paper proposes an optimization method for grid-connected MG energy storage scheduling based on the DQN cooperative algorithm, aiming at minimizing the cost of electricity expenses, which is named AP DQN. Specifically, the proposed algorithm combines the multi-headed attention mechanism with the PER mechanism in DQN to improve its performance. In this configuration, the DQN interacts with the environment to obtain Q values and form rewards, and uses prioritized experience replay to stabilize learning. In addition, the algorithm computes the weighted features of the state using the multi-headed attention mechanism, and uses the weighted features to compute the Q-value and priority, which can make the state-action pair information of the terminal closer to the merit-seeking target, thus improving the overall convergence speed of the DQN. The case study verifies the effectiveness of the proposed algorithm for grid-connected MG energy storage scheduling with real-world data. The MPCLP algorithm is subsequently benchmarked against the optimal global solution.

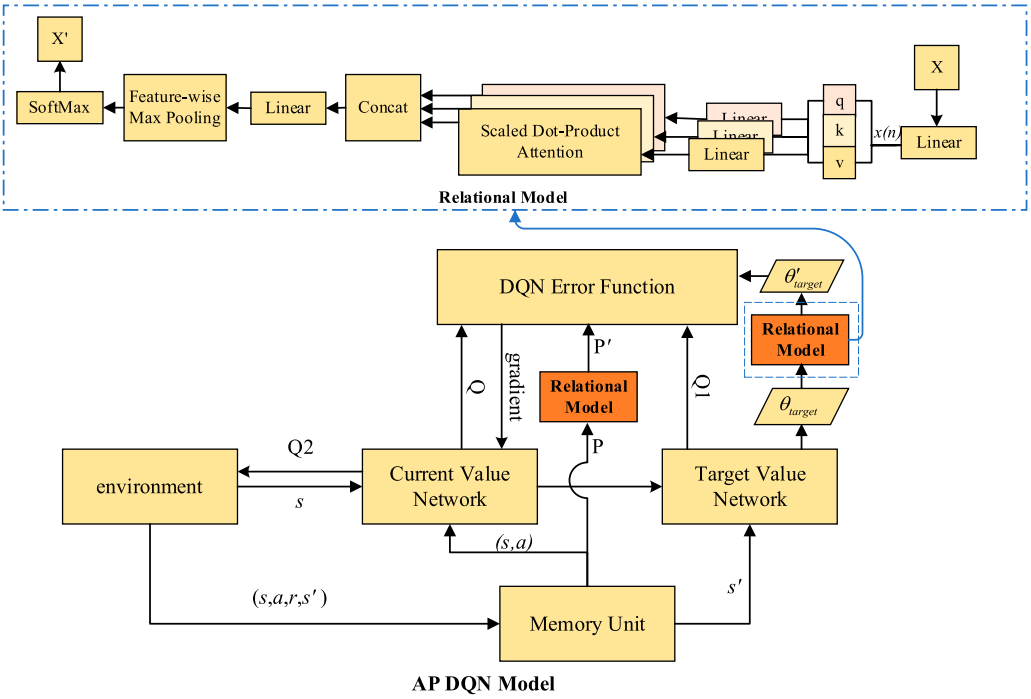

FIGURE 2
Algorithm operation process diagram.

TABLE 1 Hyper-parameters.

ε_{str}	ε_{stp}	d	err	α	β
1.1	0.01	0.0001	0.01	0.8	0.6

The primary contributions of this paper can be summarized as follows:

1. A precise mathematical model encompassing both linear and nonlinear power losses is developed to address the issue of multivariate loss factors in MGs.
2. A game combination optimization scheme based on deep reinforcement learning algorithm DQN is constructed based on the problem of difficult to handle multivariate uncertainties in MGs.
3. The AP DQN algorithm incorporating the multi-head attention mechanism is proposed for the problem of lossy features. Experimental results show that the method greatly improves the exploration efficiency. From the perspective of cost objective, our model outperforms the standard DQN by 33.5% and outperforms the MPCLP-based mechanism by up to 17.74%.

2 Microgrid's DRL model

2.1 Environment model

The environment model serves as the MG system environment that interacts with the agent. In this project, we considered a MG with internal user loads, a photovoltaic field and an energy storage system (ESS), which is connected to the main grid through only one distribution line. Figure 1 illustrates the conceptual MG model that is envisioned in this study. The MG is managed by an energy management system (EMS), which fully controls all operations of the MG, including the processes of charging and discharging the ESS, as well as the power trading activities between the MG and the main grid. To enhance the stability and ensure the uninterrupted operation of mission-critical activities, it is necessary to monitor the state of the microgrid's emergency load reserve during main grid outages, called the state of charge (SOC) in the following article. We divide the MG system into 24 time slots and each time slot is denoted as t. To enable analytical calculations, the microgrid's power is assumed to be balanced, and a quasistatic time-varying energy model is employed.

Reinforcement learning can be characterized as a Markov Decision Process (MDP) comprising a state space S, an action space A, a utility or payoff function r (utility and payoff functions are used in the report), a state transfer probability matrix P and a discount factor γ (Moradi et al., 2018). The learning process is the process of making action decisions after obtaining the next state and reward return through the interaction between the agent and the environment, and then continuously optimizing. The discount factor γ modulates the agent's consideration of the long-term consequences of their decisions on future states: (1) small values of γ force agents to focus more on the immediate payoffs of the next few steps and significantly reduce the payoffs of future steps; (2) large values of γ force actors to think more strongly about future payoffs and thus become more farsighted.

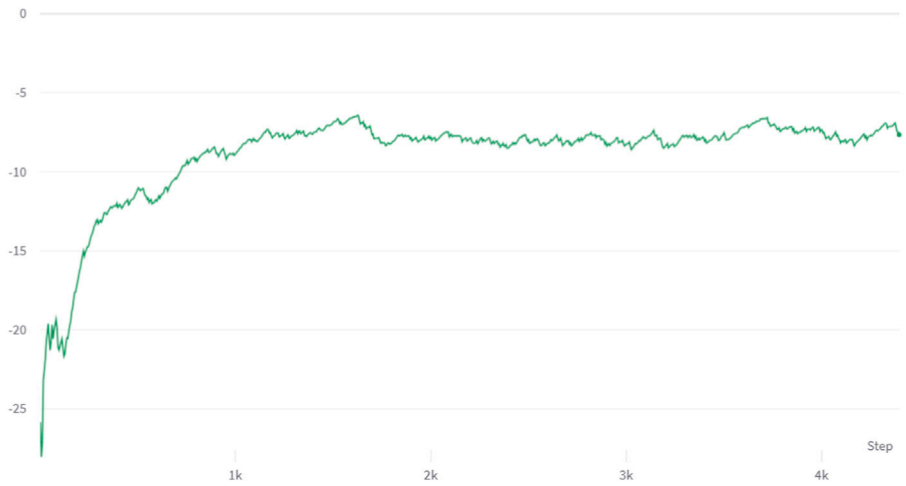

FIGURE 3
Mean episode reward with AP DQN (smoothing 0.8).

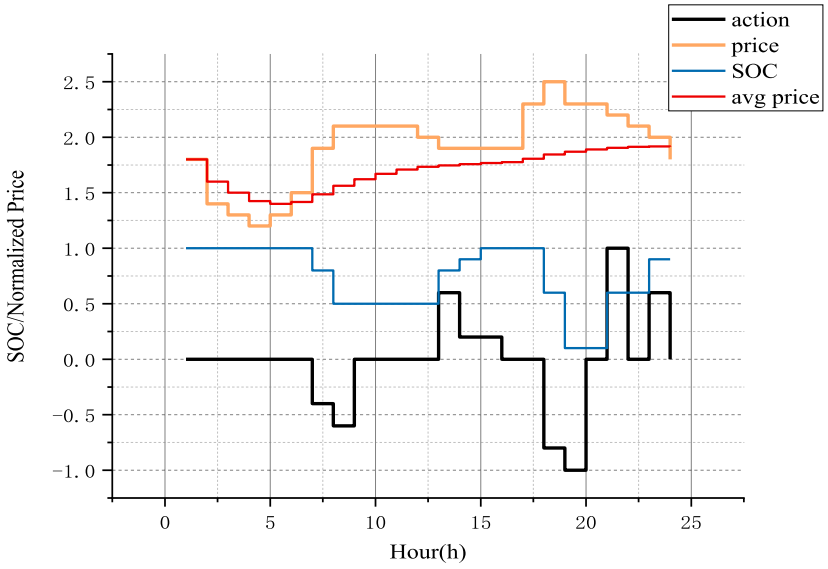

FIGURE 4
24-h average MG scheduling partial result with AP DQN.

2.1.1 ESS model

In this system, ESS mainly performs charging and discharging operations with an action space range of −1 to 1. A positive value represents charging, while a negative value indicates discharging. We define $A_t \in \{-1, -0.8, ..., 0.8, 1\}$ as the discrete action set. In each time slot t, the ESS is limited to performing either a charging action or a discharging action, but not both simultaneously. The state of the SOC is updated as follows (Chen and Su, 2018):

$$SOC_{t+1} = \begin{cases} SOC_t + \dfrac{A_t \times P_r \times \eta_c \times \Delta t}{E_r \times \eta_d}, A_t \geq 0 \\ SOC_t + \dfrac{A_t \times P_r \times \Delta t}{E_r \times \eta_d}, else \end{cases}$$

where parameters η_c, η_d, P_r, E_r represent the charging efficiency of the ESS, discharging efficiency of the ESS, rated power of the ESS, and energy storage capacity of the ESS, respectively. The energy trading mechanism incorporates the consideration of wear and tear costs. The ESS wear cost coefficient, denoted as k, is defined as follows:

$$k = \frac{C_i}{\eta_d \times E_r \times \delta \times N_c}$$

where parameters C_i, δ, N_c represent the initial investment cost of the ESS, the depth-of-discharge and the number of life cycles at a rated of the depth-of-discharge, respectively.

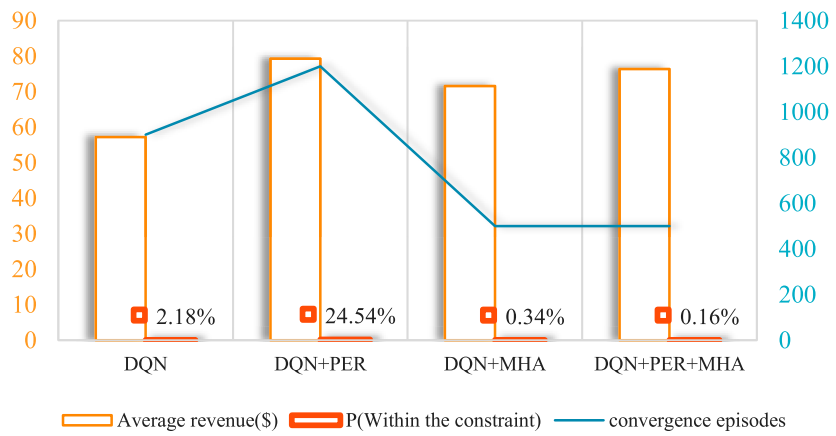

FIGURE 5
Results of ablation experiments.

2.1.2 PV model

The DC power generated by the PV module undergoes filtering in the DC circuit to eliminate current fluctuations and electromagnetic interference. It is then converted into AC power in the inverter circuit. The resulting AC power is rectified to obtain sinusoidal AC power. Subsequently, the output-side filter circuit is employed to mitigate high-frequency interference signals generated during the inverting process. This enables integration into the grid or direct supply to the load. The losses incurred during these transformations can be mathematically expressed as follows:

$$P_{loss}^{pv} = \left(P_{loss}^{DC} + P_{loss}^{AC}\right)/P_C$$

where parameters $P_{loss}^{pv}, P_{loss}^{DC}, P_{loss}^{AC}, P_C$ represent the photovoltaic inverter losses, the DC/AC loss and the installed capacity, respectively.

2.2 Real power loss of loads

Given the diverse characteristics of loads and their varying operational conditions, we adopt distinct methods for evaluating power losses. In the case of linear loads, we calculate losses by subtracting the output power from the input power to achieve greater accuracy. For nonlinear loads, we consider power factor adjustments to account for the influence of factors such as harmonics and phase differences. The expression for real power loss in the load is as follows:

$$P_L = \sum_{i=1}^{N}\left[\left(P_i^{lin} - P_i^{lout}\right) + \delta_p\left(\bar{P}_i^{nin} - \bar{P}_i^{nout}\right)\right]$$

where parameters $P_L, P_i^{lin}, P_i^{lout}$ represent the real power loss of loads, the linear loads power input, the linear loads power output. The parameters $\delta_p, \bar{P}_i^{nin}, \bar{P}_i^{nout}$ represent the power factor, the average nonlinear loads power input, the average nonlinear loads power output (Sima et al., 2023). The N act as the number of loads.

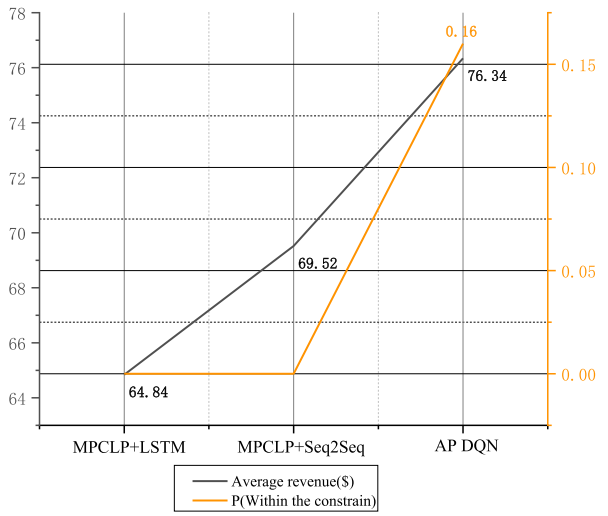

FIGURE 6
Results of comparison experiments.

2.3 Objective function and constrains designs

To keep the energy trading decisions of the MG within a reasonable range, we specify that the ESS must reserve enough energy for the critical tasks, named the target SOC, to minimize the MG operation cost under this constraint. The constraint functions are as follows:

$$A_t \times \frac{P_r}{E_r} \leq 1 - SOC_t, A_t \in (0, 1]$$

$$A_t \times \left(-\frac{P_r}{E_r}\right) \leq SOC_t - SOC_{target}, A_t \in [-1, 0]$$

With such a constraint, the system is able to reserve enough emergency power for the MG in the case of an accident scenario. In addition, the objective function of the optimization is described as follows:

TABLE 2 Comparison and ablation results of different model.

Technique	*p* (%)	Avg revenue ($)	Avg convergence episodes
MPCLP + LSTM	0.00	64.84	—
MPCLP + Seq2Seq	0.00	69.52	—
DQN	2.18	57.18	9000
DQN + PER	24.54	79.28	12000
DQN + MHA	0.34	71.58	4000
Our Model	**0.16**	**76.34**	**2000**

The bold values represents the method we proposed.

$$Obj_t = min\sum_{t=1}^{24}\left[(Pr_t + k) \times A_t \times P_r \times \frac{L_t}{L_t - P_L + PV_t^{'}}\right]$$
$$\text{in which } PV_t{}' = PV_t - P_{loss}^{pv}$$

where Pr_t denotes the electric price at time t, L_t denotes the consumer load power at time t, and $PV_t{}'$ represents the actual PV power in the MG.

3 Materials and methods

First, this paper designs a more accurate mathematical model of multivariate loss factors for microgrids with respect to loss uncertainty as well as ambiguity. Then, based on the problem of loss feature diversity, an optimization scheme of deep reinforcement learning algorithm DQN combined with multi-attention mechanism is proposed, which utilizes the principle of attention to process loss data of different sizes more efficiently, and ultimately derives the optimal scheduling actions of the energy management system for microgrids according to the objective of economic optimization.

3.1 Heading baseline-DQN

The issue examined in this paper pertains to a high-dimensional uncertainty problem that is not amenable to traditional algorithmic solutions. Reinforcement learning is a frontier area of machine learning and is a hot topic in the field of intelligent systems research. Reinforcement learning distinguishes itself from supervised learning in terms of the availability of training labels or targets. In supervised learning, the correct labels are provided to train the model. In contrast, reinforcement learning operates without explicit targets and adopts a trial-and-error approach. The model learns from its past mistakes to iteratively enhance its decision-making abilities for future actions (Mnih et al., 2013).

In the traditional approach to solving the reinforcement learning problem, a Q-table is constructed to store the Q-values, which represent the expected rewards of taking specific actions in particular states. The Q-table is updated utilizing an iterative updating rule that takes into account the recursive relationship between the Q-values. Nevertheless, when a continuous state space is encountered, it becomes impractical to create a state-action table to record every possible combination of states and actions. To overcome the limitation, a neural network known as the DQN is employed. The DQN takes the states as inputs and generates the Q-values for each possible action as outputs, which is trained through the trial-and-error process (Mnih et al., 2013). The Q-values are subsequently updated using the Bellman equation as follows:

$$Q(S_t, A_t) = r_t + \gamma \times max_{A_{t+1}}(Q(S_{t+1}, A_{t+1}))$$

where $Q(S_t, A_t)$ is the Q-value at time t, and $max_{A_{t+1}}(Q(S_{t+1}, A_{t+1}))$ denotes the maximum Q-value taking optimal action at the subsequent step. Under the policy, the value of taking action A_t at S_t must equivalent to the expected reward of transitioning to the next state S_{t+1} plus the discounted expected Q-value of taking the best decision A_{t+1} at S_{t+1} (Mnih et al., 2013). The interdependence among the Q-values at consecutive steps ensures that the iterative update rule enables the discovery of an optimal policy, leading to the convergence of Q-values towards their optimal values. This recursive relationship facilitates the convergence of the Q-value iteration process, allowing for the determination of an optimal policy.

3.2 AP DQN method

In this section, we design the AP DQN method. There are two main modules in this algorithm, one of them is a learning network model based on PER DQN, and the other is a relational network model that includes the multi-head attention mechanism. The multi-head attention mechanism in our work is applied to focus on relevant samples in the experience replay process as well as the Q-value handling process. The innovations of this algorithm are mainly represented in the following: the multi-head attention mechanism is adopted to enable the network to process the input sequences in parallel, and the model is able to realize the information fusion and sharing so as to enhance the learning ability; the network structure is improved comprehensively, and the addition of the relational model layer to weight the Q-value provides stronger adaptive learning flexibility for the network weights. The algorithm operation process diagram is shown in Figure 2. In this certain case, we put the mathematical models of ESS and PV and constraints of devices in the environment module.

Where Q indicates $Q(S_t, A_t)$, Q1 indicates $max_{A_{t+1}}(Q(S_{t+1}, A_{t+1}))$, and Q2 indicates $argmax_{A_{t+1}}(Q(S_{t+1}, A_{t+1}))$. p indicates the stored experience tuple (s, a, r, s').

We use a multi-headed attention mechanism in experience replay memories. This allows for the selection of experience replay samples by using the multi-headed attention mechanism to focus on all past samples in the memory pool and select those that are most important and relevant for the current learning. At the same time, the multi-headed attention mechanism is used in the calculation of Q values. When calculating $\boldsymbol{Q}(\boldsymbol{S_t}, \boldsymbol{A_t})$ values for each action $\boldsymbol{A_t}$ under the state $\boldsymbol{S_t}$, the different features of s can be weighted using multi-headed attention, so that the Q value calculation focuses more on those state features that are most important at the moment. This can produce more accurate Q-value estimates.

3.2.1 Algorithmic framework

In this section we design the algorithm framework, the operation process is as follows:

1. First, we initialize the playback memory unit, the priority weights array $\boldsymbol{P}$, and the Q network and target network parameters.
2. We capture the experience tuple $(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{s'})$ in the environment and store it in the memory unit.
3. For each experience tuple $(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{s'})$ stored, the attention-weighted feature $\boldsymbol{x'}$ of $\boldsymbol{s}$ is computed using the multi-headed attention mechanism:

(a). Calculate attention headers with number of K:

$$\boldsymbol{attn\_head_k} = \boldsymbol{Softmax}(\boldsymbol{W_k X_s} + \boldsymbol{b_k}), \boldsymbol{k} = \boldsymbol{1}, \boldsymbol{2}, ..., \boldsymbol{K}$$

where $\boldsymbol{W_k}$ indicates the attention parameter matrix, $\boldsymbol{X_s}$ is the matrix corresponding to the state, and $\boldsymbol{b_k}$ is decided by the attention value.

(b). Fuse the attention header to obtain the final attention value $\boldsymbol{attn}$.

(c). Calculate the weighted characteristics:

$$\boldsymbol{x'} = \sum_{\boldsymbol{i}=\boldsymbol{1}}^{\boldsymbol{N}} \boldsymbol{attn}[\boldsymbol{i}]^{*}\boldsymbol{X_s}[\boldsymbol{i}]$$

4. Calculate the priority $\boldsymbol{p}$ of each tuple, $\boldsymbol{Q_{target}}$ is calculated using the target network parameters, and $\boldsymbol{\theta_{target}}$ is the target network parameter:

$$\boldsymbol{p} = \left(|\boldsymbol{r}| + \boldsymbol{\gamma}^{*}\boldsymbol{max_{a'}}\left(\boldsymbol{Q_{target}}\left(\boldsymbol{x'}, \boldsymbol{a'}; \boldsymbol{\theta_{target}}\right)\right)\right)^{\boldsymbol{\alpha}}$$

5. Select the experience tuple with the number of batch size for learning by priority.

(a). Calculate the Q value for each experience using the Q network and the weighted state $\boldsymbol{x'}$.

(b). Calculate the loss of each experience using the Q target:

$$\boldsymbol{L} = \left(\boldsymbol{Q_{target}} - \boldsymbol{Q}(\boldsymbol{x'}, \boldsymbol{a'}; \boldsymbol{\theta})\right)^{2}$$

(c). Gradient descent updates the Q-network parameters $\boldsymbol{\theta}$.

6. Update the priority array $\boldsymbol{P}$ and the target network parameter $\boldsymbol{\theta_{target}}$
7. Repeat steps 2-6 for training.

3.2.2 Reward function design

A segmented reward function is designed to guide the trading strategy provided that all conditions are satisfied, where the reward value depends on:

1. The state difference from the target SOC.
2. The final cost obtained from the MG operation.

Below the target SOC, it is imperative to prioritize charging the ESS promptly, irrespective of the price. Similarly, the price must be high enough to discharge the ESS below the target SOC. Therefore, the charging and discharging criteria for the ESS differ depending on whether the SOC is below or above the target level. To optimize the utilization of the remaining storage capacity, the charging price for the ESS should decrease as the state of charge (SOC) approaches full SOC. This incentivizes efficient charging when there is ample capacity available. Conversely, the price for discharging the ESS should be higher when the SOC is closer to the target SOC. This approach encourages the effective utilization of the remaining available energy and ensures that the SOC is maintained at the desired level. In addition to this setting, two penalty factors are introduced to have further control of the ESS operational behavior. The first penalty term $\boldsymbol{Pnt_t^{ESS}}$ is applied when the action chosen by the agent violates a constraint within the system. The second penalty term $\boldsymbol{Pnt_t^{PV}}$ is assigned when the ESS with available energy capacity fails to store excess solar energy. The first penalty term is introduced to account for the constraint of the ESS, aiming to extend the operational lifetime of the unit, while the second penalty term serves the purpose of maximizing the storage of solar energy within capacity limit of the ESS. The reward function R is as follows:

$$\boldsymbol{R}\left(\boldsymbol{Pr_t}, \boldsymbol{Pr_t^{avg}}, \boldsymbol{SOC_t}\middle|\boldsymbol{A_t}\right) = \left(\boldsymbol{Pr_t^{avg}} - (\boldsymbol{Pr} + \boldsymbol{k})\right) \times (\boldsymbol{SOC_{t+1}} - \boldsymbol{SOC_t}) \times \boldsymbol{E_r} - \boldsymbol{Pnt_t^{ESS}} - \boldsymbol{Pnt_t^{PV}}$$

$$\boldsymbol{Pr_t^{avg}} = \frac{\sum_{\boldsymbol{t}=\boldsymbol{0}}^{\boldsymbol{24}} \boldsymbol{Pr_t}}{\boldsymbol{24}}$$

$$\boldsymbol{Pnt_t^{ESS}} = \begin{cases} \boldsymbol{0}, \text{else} \\ \boldsymbol{10}, \text{if } \boldsymbol{SOC_t} + \boldsymbol{A_t} > \boldsymbol{1} \text{ or } \boldsymbol{SOC_t} - \boldsymbol{A_t} < -\boldsymbol{1} \end{cases}$$

$$\boldsymbol{Pnt_t^{PV}} = \begin{cases} \boldsymbol{0}, \text{if } \boldsymbol{PV_t} \le (\boldsymbol{L_t} + \boldsymbol{A_t} \times \boldsymbol{P_r}) \\ \boldsymbol{exp}(2.5 \times (1 - \boldsymbol{SOC_{t+1}}))1 - \boldsymbol{SOC_{t+1}}))2.5 \times -1, \text{if } \boldsymbol{PV_t} > (\boldsymbol{L_t} + \boldsymbol{A_t} \times \boldsymbol{P_r}) \end{cases}$$

where $\boldsymbol{Pr_t^{avg}}$ represents the average price observed throughout the 24 time slots preceding time $\boldsymbol{t}$.

3.2.3 Relational model

The main idea of the relational model is the weighted encoding of states using a multi-headed attention mechanism. The attention mechanism can be understood as a process of addressing information, where the attention value is computed by calculating the attention distribution based on the key and associating it with the value. This computation is performed with respect to a task-specific query vector Q, allowing the attention mechanism to focus on relevant information and selectively combine it with the query. By dividing each query, key, and value into multiple branches, multiple different attention calculations are performed on Q, K, and V to obtain multiple different outputs, and then these different outputs are stitched together to obtain the final output. Indeed, this process represents the essence of attention, which helps mitigate the complexity of neural networks. Instead of feeding all N inputs into the network for computation, attention selectively chooses task-relevant information to be inputted. This approach is similar

to the concept of gating mechanisms in Recurrent Neural Networks (RNNs), where the network learns to focus on relevant information and effectively allocate computational resources (Azam and Younis, 2021).

Due to the priority sampling strategy, PER introduces a bias towards selecting higher priority samples during training (Schaul et al., 2015). This bias has the potential to lead to overfitting of the results obtained by the DQN algorithm. Therefore, to correct for bias, we introduce the relational model to adjust the sampling weights. The built-in attention mechanism allows direct monitoring of the training process by highlighting the areas that agents focus on when making decisions. It naturally incorporates the policy gradient algorithm in reinforcement learning, where each time-step attention mechanism samples from $\boldsymbol{L} = \boldsymbol{m}^{*}\boldsymbol{m}$ to a position requiring attention based on a random attention policy $\boldsymbol{\pi_g}$. This policy is represented using a neural network whose output is composed of the probabilities of location selection. Among them, the formula for calculating and updating the policy gradient algorithm is as follows:

$$
\begin{aligned}
\nabla \boldsymbol{J}(\boldsymbol{\theta}) &= \sum_{\boldsymbol{s}} \boldsymbol{\mu}_{\pi}(\boldsymbol{s}) \sum_{\boldsymbol{a}} \boldsymbol{q}_{\pi}(\boldsymbol{s}, \boldsymbol{a}) \nabla_{\theta} \boldsymbol{\pi}(\boldsymbol{a}|\boldsymbol{s}, \boldsymbol{\theta}) \\
&= \boldsymbol{E}_{\pi}\left[\boldsymbol{\gamma}^{t} \sum_{\boldsymbol{a}} \boldsymbol{q}_{\pi}(\boldsymbol{S}_{t}, \boldsymbol{a}) \nabla_{\theta} \boldsymbol{\pi}(\boldsymbol{a}|\boldsymbol{S}_{t}, \boldsymbol{\theta})\right] \\
&= \boldsymbol{E}_{\pi}\left[\boldsymbol{\gamma}^{t} \sum_{\boldsymbol{a}} \boldsymbol{q}_{\pi}(\boldsymbol{S}_{t}, \boldsymbol{a}) \boldsymbol{\pi}(\boldsymbol{a}|\boldsymbol{S}_{t}, \boldsymbol{\theta}) \frac{\nabla_{\theta} \boldsymbol{\pi}(\boldsymbol{a}|\boldsymbol{S}_{\theta}, \boldsymbol{\theta})}{\boldsymbol{\pi}(\boldsymbol{a}|\boldsymbol{S}_{\theta}, \boldsymbol{\theta})}\right] \\
&= \boldsymbol{E}_{\pi}\left[\boldsymbol{\gamma}^{t} \boldsymbol{q}_{\pi}(\boldsymbol{S}_{t}, \boldsymbol{A}_{t}) \frac{\nabla_{\theta} \boldsymbol{\pi}(\boldsymbol{A}_{t}|\boldsymbol{S}_{t}, \boldsymbol{\theta})}{\boldsymbol{\pi}(\boldsymbol{A}_{t}|\boldsymbol{S}_{t}, \boldsymbol{\theta})}\right] \\
&= \boldsymbol{E}_{\pi}\left[\boldsymbol{\gamma}^{t} \boldsymbol{G}_{t} \frac{\nabla_{\theta} \boldsymbol{\pi}(\boldsymbol{A}_{t}|\boldsymbol{S}_{t}, \boldsymbol{\theta})}{\boldsymbol{\pi}(\boldsymbol{A}_{t}|\boldsymbol{S}_{t}, \boldsymbol{\theta})}\right] \\
\boldsymbol{\theta}_{t+1} &= \boldsymbol{\theta}_{t} + \boldsymbol{\alpha} \boldsymbol{\gamma}^{t} \boldsymbol{G}_{t} \frac{\nabla_{\theta} \boldsymbol{\pi}(\boldsymbol{A}_{t}|\boldsymbol{S}_{t}, \boldsymbol{\theta})}{\boldsymbol{\pi}(\boldsymbol{A}_{t}|\boldsymbol{S}_{t}, \boldsymbol{\theta})}
\end{aligned}
$$

where $\nabla \boldsymbol{J}(\boldsymbol{\theta})$ indicates the strategy gradient and $\boldsymbol{G_t}$ indicates the cumulative rewards. $\boldsymbol{\alpha}$ indicates the step length and $\boldsymbol{\gamma}$ indicates the discount factor.

4 Experiments and results

In this section, we present simulation results to demonstrate the effectiveness of the proposed algorithm. These results serve as empirical evidence supporting the performance and efficacy of the algorithm. Specifically, the DQN architecture employed in this study consists of one input layer with four neurons, three fully connected hidden layers with 40 and 80 neurons, and one output layer with 14 neurons. This configuration allows for effective learning and decision-making within the energy management algorithm. $\boldsymbol{\varepsilon}$ greedy strategy and hyperparameters of PER are listed in Table 1. The mean episode reward with AP DQN is shown in Figure 3. The customer load, solar power and dynamic tariff are obtained from the self-built datasets. The $\boldsymbol{P_r}$ and $\boldsymbol{E_r}$ of the lithium-ion battery ESS used in the experiment are 1,000 kW and 5,000 kWh respectively.

Due to the large range of resultant data, we chose the average MG scheduling results over a time horizon of 24 h as a demonstration of the scheduling strategy, and the result with AP DQN is shown in Figure 4. Due to the large time horizon involved in the dataset, the obtained ESS scheduling strategy is measured in terms of the final economic cost and the percentage of the system working within the constraints. We used a model predictive control linear programming (MPCLP) based algorithm (Matute et al., 2018) for comparison and performed ablation experiments. MPCLP is a linear programming optimization method, which commonly employs an optimization software to work out the problem. It provides good optimization accuracy while satisfying the assumptions of a linear dynamic system. Among them, MPCLP uses two prediction models, LSTM and Seq2Seq, respectively. The results of the ablation experiments are shown in Figure 5. The results of the comparison experiments are shown in Figure 6.

As seen in Figure 4, Positive values of action in the figure indicate charging, negative values indicate discharging, and SOC ranges from 0 to 1. It can be concluded that the EMS will combine the state of the SOC at the moment with the floating tariff to give the best possible action within the constrains.

As seen in Figure 5, the base DQN has poor performance in the ablation experiment, but the average gain rises significantly with the addition of PER, however, this is a result of large-scale constraint violations. With the addition of the multi-head attention mechanism, the algorithm is able to obtain an average return close to the PER DQN while maintaining a certain range of constraints. After adding the multi-headed attention mechanism to DNQ together with PER, the result of maximizing the average gain and minimizing the probability of constraint violation can be obtained.

As seen in Figure 6, AP DQN has the highest average profit in the comparison experiment, but there is a default rate of 0.16%, although this is an acceptable range. The reason for this is that the traditional linear programming approach has a strict adherence to the constraints and therefore a p-value of 0. In contrast, the proposed AP DQN algorithm can violate the constraints driven by the reward values to a minor degree, thus achieving the goal of maximizing the average profit.

The results obtained from the comparative experiments and ablation studies using different models are summarized in Table 2. Comparison and Ablation Results of Different Model. As can be seen from the table that our model outperforms the standard DQN by 33.5%, the MPCLP based mechanism by 17.74% at most. Compared with PER DQN, our model is a better choice in terms of algorithmic efficiency and conditional constraints.

5 Conclusion

In this paper, we propose an AP DQN algorithm. The algorithm not only maximizes monetary benefits but also maintains the reliability of the MG at the same time, being able to maintain sufficient energy reserves for critical operations. The algorithm presented uses a multi-headed attention mechanism as well as a prioritized experience replay mechanism to use current information for optimizing energy trading decisions. The algorithm we propose is a model-free reinforcement learning method, which usually has strong generalization ability. This method learns a wide range of strategies from a large amount of empirical data so that it can make reasonable decisions in uncovered states and can adapt better to various situations and conditions. In

comparison with the MPCLP approach, it can be concluded that the reinforcement learning based approach has a higher average monetary benefit in the presence of higher system reliability. It is worth noting that the reward function in RL can be further adjusted and optimized to improve the overall results. Fine-tuning the reward function has a significant impact on the performance of the RL algorithm. Additionally, it is important to consider that value-based RL methods generate discrete trading decisions, whereas MPCLP decisions are continuous in nature. This distinction can affect the comparison of results obtained from the two approaches. In future work, policy-based reinforcement learning is an appropriate direction to be investigated to obtain continuous decisions.

Data availability statement

The data analyzed in this study is subject to the following licenses/restrictions: The dataset in this article is a power industry dataset that cannot be disclosed according to regulations. Requests to access these datasets should be directed to ZZ, zhaozhenbing@ncepu.edu.cn.

Author contributions

SS: Conceptualization, Formal Analysis, Investigation, Project administration, Supervision, Writing–original draft. WG: Data curation, Formal Analysis, Methodology, Writing–original draft. QW: Validation, Visualization, Writing–original draft. PT: Investigation, Validation, Writing–review and editing. GL: Formal Analysis, Investigation, Visualization, Writing–review and editing. ZZ: Conceptualization, Data curation, Methodology, Supervision, Writing–original draft.

Funding

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. This research was funded by the S&T Program of Hebei (22284504Z).

Acknowledgments

Heartfelt thanks to everyone who contributed to this paper.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

References

Azam, M. F., and Younis, M. S. (2021). Multi-horizon electricity load and price forecasting using an interpretable multi-head self-attention and EEMD-based framework. *IEEE Access* 9, 85918–85932. doi:10.1109/ACCESS.2021.3086039

Chen, T., and Su, W. (2018). Local energy trading behavior modeling with deep reinforcement learning. *IEEE Access* 6 (2), 62806–62814. doi:10.1109/ACCESS.2018.2876652

Fang, C., Lu, H., Hong, Y., Liu, S., and Chang, J. (2020). Dynamic pricing for electric vehicle extreme fast charging. *IEEE Trans. Intell. Transp. Syst.* 22 (1), 531–541. doi:10.1109/TITS.2020.2983385

Farzaneh, H., Shokri, M., Kebriaei, H., and Aminifar, F. (2019). Robust energy management ofresidential nanogrids via decentralized mean field control. *IEEE Trans. Sustain. Energy* 11 (3), 1995–2002. doi:10.1109/TSTE.2019.2949016

Gan, L. K., Zhang, P., Lee, J., Osborne, M. A., and Howey, D. A. (2020). Data-driven energy management system with Gaussian process forecasting and mpc for interconnected microgrids. *IEEE Trans. Sustain. Energy* 12 (1), 695–704. doi:10.1109/TSTE.2020.3017224

Kim, B. G., Zhang, Y., Schaar, M. V., and Lee, J. W. (2016). Dynamic pricing and energy consumption scheduling with reinforcement learning. *IEEE Trans. Smart Grid* 7 (5), 2187–2198. doi:10.1109/TSG.2015.2495145

Kumari, R. L., Kumar, G. N., Nagaraju, S. S., and Jain, M. B. (2017). "Optimal sizing of distributed generation using particle swarm optimization," in 2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies, Kerala, India, July, 2017. doi:10.1109/ICICICT1.2017.8342613

Lee, S., and Choi, D. H. (2022). Federated reinforcement learning for energy management of multiple smart homes with distributed energy resources. *IEEE Trans. Ind. Inf.* 18 (1), 488–497. doi:10.1109/TII.2020.3035451

Ma, W. J., Wang, J., Gupta, V., and Chen, C. (2018). Distributed energy management for networked microgrids using online admm with regret. *IEEE Trans. Smart Grid* 9 (2), 847–856. doi:10.1109/TSG.2016.2569604

Matute, J. A., Marcano, M., Zubizarreta, A., et al. (2018). "Longitudinal model predictive control with comfortable speed planner," in IEEE International Conference on Autonomous Robot Systems and Competitions, Torres Vedras, Portugal, April, 2018, 25–27. doi:10.1109/ICARSC.2018.8374161

Mnih, V., Kavukcuoglu, K., Silver, D., et al. (2013). Playing atari with deep reinforcement learning. *CoRR*. doi:10.48550/arXiv.1312.5602

Moradi, H., Esfahanian, M., Abtahi, A., and Zilouchian, A. (2018). Optimization and energy management of a standalone hybrid micro-grid in the presence of battery storage system. *Energy* 147, 226–238. doi:10.1016/j.energy.2018.01.016

Patel, A. G., and Patel, C. (2016). "Distribution network reconfiguration for loss reduction," in 2016 International Conference on Electrical, Electronics, and Optimization Techniques, Chennai, India, March, 2016, 3937–3941. doi:10.1109/ICEEOT.2016.7755453

Paterakis, N. G., Erdinc, O., Bakirtzis, A. G., and Catalao, J. P. S. (2015). Optimal household appliances scheduling under day-ahead pricing and load-shaping demand response strategies. *IEEE Trans. Ind. Inf.* 11 (6), 1509–1519. doi:10.1109/TII.2015.2438534

Remani, T., Jasmin, E. A., and Ahamed, T. P. I. (2019). Residential load scheduling with renewable generation in the smart grid: a reinforcement learning approach. *IEEE Syst. J.* 13 (3), 3283–3294. doi:10.1109/JSYST.2018.2855689

Schaul, T., Quan, J., Antonoglou, I., et al. (2015). Prioritized experience replay. *CoRR*. doi:10.48550/arXiv.1511.05952

Sima, L., Miteva, N., and Dagan, K. J. (2023). A novel approach to power loss calculation for power transformers supplying nonlinear loads. *Electr. Power Syst. Res.* 223, 109582. doi:10.1016/j.epsr.2023.109582

Wu, Y., Lee, C. Y., Liu, L. C., and Tsai, S. H. (2010). Study of reconfiguration for the distribution system with distributed generators. *IEEE Trans. Power Deliv.* 25 (3), 1678–1685. doi:10.1109/TPWRD.2010.2046339

Zhang, Z., and Kang, C. (2022). Challenges and prospects for constructing the new-type power system towards a carbon neutrality future. *Proc. CSEE* 42, 2806–2819. doi:10.13334/j.0258-8013.pcsee.220467