Energy 329 (2025) 136661

Contents lists available at ScienceDirect

Energy

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

# A double deep reinforcement learning-based adaptive framework for decision-optimal wind power interval prediction

Chenghan Li, Ye Guo, Yinliang Xu *

*Institute of Data and Information, Tsinghua Shenzhen International Graduate School , Tsinghua University, Shenzhen 518055, China*

ARTICLE INFO

*Keywords:*
Quantile prediction
Long short-term memory
Attention mechanism
Reinforcement learning
Double Deep Q-Network

ABSTRACT

Prediction intervals (PI) effectively quantify forecasting uncertainty and serve as inputs for subsequent decision-making tasks. While it is traditionally assumed that reducing prediction errors will correspondingly reduce decision errors, this assumption is not invariably valid. This paper introduces an adaptive decision-optimal framework for optimal interval forecasting in wind power, designed to alleviate the economic dispatch challenges posed by wind power uncertainty within power systems. Methodologically, this framework employs a close-loop method based on the Double Deep Q-Network algorithm, where the forecasting module leverages a pre-trained model with Bi-Directional Long Short-Term Attention to enhance extracting features of historical data and increase quantile forecast precision. Then, Double Deep Q-Network can select decision-optimal quantile proportions. The validity of the framework is demonstrated through experiments utilizing real-world wind power data from the Belgian Elia company, validated across IEEE 6-bus and 30-bus cases. The method decreased average operation cost and risk by 0.36%/4.38% in the IEEE 6-bus and 0.76%/6.29% in the IEEE 30-bus compared to benchmarks. This framework offers a robust solution for wind power probability forecasting and the decision of power systems, thereby enhancing the stability and economic efficiency of power system operations.

## 1. Introduction

With the development of new energy, the integration of a high proportion of new energy sources introduces significant uncertainty into the power system, which substantially increases the risk of an imbalance between power supply and demand. This not only leads to frequent rescheduling and the curtailment of wind and solar energy but also significantly raises the economic costs of power dispatch. In extreme cases, such imbalances could even cause widespread power outages, posing a severe threat to the safe and stable operation of the power grid. Traditional power dispatching is conducted on predictive models and sequential execution of decisions, which are contingent upon the informational foundation provided by forecasts of renewable energy generation [1]. As the integration of renewable energy sources increases, inaccuracies in forecasting can amplify the decision-making risks within power systems [2].

Forecasting problems manifest in various forms, with variables classified into univariate, multivariate [3], and covariate forecasting. Regarding the form of forecasting, they are categorized as point forecasting [4] and probabilistic forecasting [5]. Wind power probability forecasting can be seen as covariate probability forecasting and is characterized by a prediction interval that defines the forecast region. This interval is delineated by two boundaries and a nominal coverage probability (NCP), indicating the likelihood that the actual value will fall within the specified range. PIs are extensively applied in the electricity industry, for example, quantifying wind power uncertainty [6], dictating reserve capacity requirements, and informing wind power bids in the day-ahead market [7], with NCPs typically ranging from 90% to 95%. Moreover, the estimated PI also facilitates uncertainty budgeting in robust optimization for unit commitment [8] and microgrid [9] scheduling, thereby minimizing operational costs.

Meteorological conditions often influence wind power forecasting [10,11]. Previous studies have shown that numerical weather prediction can effectively improve forecasting accuracy. For example, Liu et al. [12] proposed a novel algorithm to enhance the representation ability of neural networks for NWP features. Meanwhile, Zhan et al. [13] proposed a network based on multiple sources of numerical weather prediction data and temporal attention mechanisms for wind power forecasting and analyzed the long-term temporal error characteristics of NWP data. Wind power often contains a large amount of noise due to the influence of the randomness of the atmospheric

* Corresponding author.
*E-mail address:* xu.yinliang@sz.tsinghua.edu.cn (Y. Xu).

https://doi.org/10.1016/j.energy.2025.136661
Received 13 January 2025; Received in revised form 14 April 2025; Accepted 17 May 2025
Available online 27 May 2025

**Nomenclature**

| | |
|---|---|
| ACD | Average Coverage Deviation |
| AIS | Average Interval Score |
| ANN | Artificial Neural Network |
| AW | Average Width |
| COML | Cost-Oriented Machine Learning |
| CRPS | Continuous Ranked Probability Score |
| DBN | Deep Belief Network |
| DDQN | Double Deep Q-Network |
| KKT | Karush–Kuhn–Tucker |
| LSTM | Long-Short-Term-Forecasting |
| MLP | Multi-Layer Perceptron |
| NCP | Nominal Coverage Probability |
| NWP | Numerical Weather Prediction |
| OPI | Optimal Prediction Intervals |
| PI | Prediction Interval |
| PTDF | Power Transfer Distribution Factor |
| QIR | Quantile Intersection Rate |
| QR | Quantile Network |
| QRNN | Quantile Regression Neural Network |
| SPO | Sequence-Prediction Optimization |

environment. Numerous studies have explored methods to reduce or eliminate noise from wind speed data using various decomposition techniques. These include Wavelet Transform [14], Empirical Model Decomposition [15] and Ensemble Empirical Mode Decomposition [16]. Mehdi Neshat et al. proposed an adaptive decomposition method to decrease noise with long short-term memory neural model to forecast wind speed accurately [17]. Yuzgec et al. [18] combined the Empirical Model Decomposition and Echo State Network for wind power prediction and achieved good results. However, considering the lack of interpretability and high computational complexity associated with signal decomposition techniques, some studies have used multi-objective optimization algorithms to improve prediction accuracy. For example, Lv et al. [19] proposed a method for multivariate wind speed prediction based on a multiobjective feature selection approach and a hybrid deep learning model. Meng et al. [20] used a multi-objective cross-optimization algorithm to adjust the parameter pairs of the extreme learning machine autoencoder to improve the prediction of wind power. However, the research questions mentioned above are focused only on wind power prediction and do not adequately consider the uncertainty of the prediction. Extensive studies have shown that quantifying the uncertainty of new energy is very important for the operation of the power system [21–23].

In recent years, the probabilistic forecasting of wind power has increasingly adopted non-parametric methods to minimize reliance on prior knowledge, statistical inference, or assumptions about error distributions. Common nonparametric probabilistic forecasting techniques include kernel density estimation [24] and quantile regression [25]. Quantile regression(QR) targets the prediction of specific quantiles, thereby characterizing the probability density distribution of future power outputs with a single parameter. Unlike kernel density estimation, which necessitates the estimation of the entire probability density function, quantile regression demands fewer computational resources and allows for the selection of relevant points based on specific requirements. This approach is particularly advantageous for assessing the uncertainty of wind power, particularly in scenarios with frequent extreme values. For instance, the literature [26] has integrated Artificial Neural Networks (ANNs) with quantile regression to develop Quantile Regression Neural Networks (QRNNs), which quantify

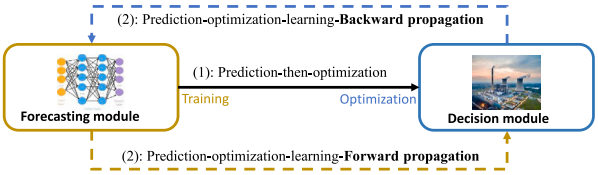



**Fig. 1.** The interaction of forecasting and decision-making .

the impact of uncertainty on wind power output probability forecasts. Building on quantile regression, Deep Confidence Networks [27] have been employed to bolster the probabilistic forecasting model's capacity to handle uncertainty, enabling the model to more effectively learn high-order nonlinearities and non-stationary characteristics within wind speed time series, thereby achieving enhanced performance.Meanwhile, reinforcement learning has also been applied to probabilistic prediction of wind power. Liu et al. [28] Proposed a probabilistic prediction method of reinforcement learning based on physical knowledge to forecast wind power under extreme weather. Zhao et al. [29] uses reinforcement learning to assign different model weights to accurately predict wind power dynamically.

While the aforementioned model is capable of predicting multiple quantiles simultaneously, it is susceptible to the quantile crossover issue. Quantile crossover occurs when the model's predictions for lower quantiles exceed those for higher quantiles, violating the fundamental monotonicity of the cumulative distribution function and resulting in implausible quantile estimates [30,31]. To address these challenges, Wang et al. [32] introduce a Deep Belief Network (DBN) model for wind power, which integrates extreme learning machines and quantile regression. This model transforms the intricate task of nonparametric probabilistic forecasting with artificial neural networks into a tractable linear programming problem, enabling precise approximation of a wide range of quantiles. Furthermore, the DBN model effectively mitigates the quantile crossover problem through the imposition of constraints. Additionally, Li et al. [33] present a parallel quantile regression model that incorporates artificial constraints to alleviate the quantile crossover issue further. Despite advances in wind power quantile forecasting, existing research, as mentioned above, does not ensure the simultaneous prediction of multiple quantiles without crossover. The introduction of artificial constraints, while addressing the crossover, may compromise the accuracy of the predictions [34].

Although prediction interval methods enhance predictive accuracy from a statistical standpoint, they often overlook the practical decision value of forecasts in the context of power system operations [35]. The concept of assessing forecast utility is rooted in the economic and operational benefits derived from their applications during decision-making processes. Consider the case of robust optimization problems, such as robust power scheduling, where probabilistic projections are integral to determining operational costs. It has been demonstrated that enhanced forecast accuracy does not invariably translate into increased operational decision-value [36–38]. For instance, a forecasting result with a bias in wind power availability may be more advantageous than a highly accurate point forecast with a minimal mean square error. This is because the cost of addressing an energy deficit by overestimating wind power supply in the day-ahead market is typically higher than the cost of managing an energy surplus through underestimation. Analogous findings are evident in the realm of unit commitment [39] (see Fig. 1).

In recent research, there also has been a growing advocacy for decision-oriented forecasting methodologies [40,41]. (see Fig. 1) The principal challenge lies in effectively connecting the forecasting process to decision-making processes. Efforts to address this issue have focused on the development of loss functions that account for decision-making perspectives. For instance, to bridge the gap between point forecasts

and decision-making, a loss function known as "Sequential Prediction Optimization" (SPO) [42] has been introduced. This approach uses historical data to approximate the objective function, but such approximations may introduce errors that can undermine the utility of forecasts. In reference, a decision cost-oriented machine learning (COML) framework is presented by Zhang et al. [43], which aims to conduct value-oriented probabilistic predictions. This is achieved by optimizing the probability distribution of quantile regression models to align with decision-making objectives. However, the COML framework simplifies the quantile regression model to a single-stage optimization problem by applying the Karush–Kuhn–Tucker (KKT) conditions, potentially limiting the model's complexity. The decision problem addressed in considers only a single decision variable, namely the quantity of available wind power, which is subject to a capacity constraint. This allows for a straightforward establishment of the link between optimal objectives and forecasts. In contrast, more intricate decision-making scenarios, such as those with multiple decision variables and constraints, lack a clear connection between predictions and optimal decision goals, complicating the implementation and validation of decision-oriented forecasts.

In summary, while advancements have been achieved in the prediction of wind power intervals, several limitations persist: (1). Current quantile prediction models are vulnerable to variations in the magnitude of losses across different target tasks. This sensitivity results in disparate convergence rates of the loss functions, potentially causing underfitting in some tasks and overfitting in others. Consequently, this leads to the quantile crossover issue, which undermines the reliability and precision of quantile predictions. (2). A discrepancy exists between the forecasted wind power generation and the actual subsequent decisions. This divergence can result in inconsistencies in prediction and dispatching, thereby compromising the effectiveness of energy management.

To address the aforementioned issues, this paper presents a decision-optimized, non-crossing joint probabilistic forecasting framework tailored for the day-head scheduling of wind turbines. The proposed method for optimal decision-making integrates a strategy learning module based on Double Deep Q-Network(DDQN) designed to select probability distributions that minimize decision-related costs. In this framework, the policy learning task is represented by an agent, while the forecasting component resides within the environment. The agent and environment are linked through a reward system, which is defined as the negative of the decision problem's objective value.In this paper, quantile selection refers to the process of using the DDQN algorithm to select the optimal quantile from multiple candidates to optimize decision-related costs and risks. Consequently, the agent is equipped to learn strategic quantile selections aligned with the decision problem's optimal goals. The key contributions of this paper are as follows:

- **Feature Extraction Enhancement:** Compared with previous studies where NWP data and historical data were combined into a multidimensional vector as input [10–13], which rely solely on sequential data processing and often fail to capture complex temporal and extra features dependencies, this paper proposes a novel forecasting module based on BiLSTM, integrated with cross-attention and self-attention mechanisms and historical data and NWP data are entered separately to extract features. This innovative approach significantly enhances the model's ability to extract both global and local features, thereby achieving a remarkable improvement in performance. Specifically, it results in a 46.5% improvement for $|E_{ACD}|$, a 28% improvement for $E_{AW}$, a 45% improvement for $E_{AIS}$, and a 14% improvement for $E_{MPICD}$ compared to the best baseline BiLSTM.
- **Non-Cross Joint Quantile Prediction:**Previous studies have primarily focused on single-quantile predictions and have not addressed the issue of quantile crossing in multiple-quantile predictions[25,26,33,34], which arises due to the lack of monotonicity constraints. This issue leads to unreliable forecasting results. This paper proposes a novel joint quantile loss function underpinned by multi-task learning principles.The loss function is capable of predicting a set of quantiles simultaneously and addresses the issue of quantile crossing by dynamically adjusting the weights of each loss function to balance the rate of loss reduction for different quantiles.This approach facilitates the concurrent prediction of multiple quantiles without crossing, as evidenced by the zero quantile intersection rate.
- **Decision-Optimized Forecasting:** In contrast to traditional dispatch methods [6,40,43], which are based on sequential prediction and optimization, this paper introduces a Decision-Optimal Prediction Interval Framework Based on DDQN. Specifically, this paper employs DDQN to adjust the forecasting outcomes, thereby reducing the decision-making risks and operational costs of the power system.The results show that our method can reduce average operating cost by 0.36% and 4.38% in the IEEE 6-bus system compared to benchmarks and 0.76% and 6.29% in the IEEE 30-bus system, thereby enhancing the stability and economic efficiency of power system operations.

The structure of this paper is as follows: Section 2 presents the mathematical modeling of the prediction interval and the objective function associated with the wind power decision problem. Section 3 provides a detailed description of the framework proposed in this study. Section 4 is dedicated to a discussion of the experimental methodology and the analysis of the results obtained. Finally, Section 5 summarizes the findings and presents the conclusions of the research.

## 2. Problem formulation

### 2.1. Economic dispatching function

The dispatch-decision optimization problem, considering the probabilistic prediction of wind power, can be formulated as a two-layer optimization problem for day-ahead dispatching. The model is extended to include multiple time steps as follows:

$$\begin{aligned}
&\sum_{t=1}^{T} \min_{\theta} R(p_{g,t}, p^{*}_{g,t}) \\
&\text{s.t.} \quad [p^{\hat{\alpha}}_{w,t}, p^{\hat{\alpha+1-\beta}}_{w,t}] \sim f(p_{\hat{w},t} \mid x_t; \theta) \quad \forall t \in \{1, \ldots, T\} \\
&\quad p_{\hat{w},t} \in [p^{\hat{\alpha}}_{w,t}, p^{\hat{\alpha+1-\beta}}_{w,t}] \quad \forall t \in \{1, \ldots, T\} \\
&\quad \{p_{g,t}\} \in \arg\min \sum_{t=1}^{T} C(p_{\hat{w},t}, p_{g,t})
\end{aligned} \tag{1}$$

In this formulation, $R$ represents the decision-risk measurement function, $p_{g,t}$ denotes the actual power dispatching scheme at time $t$, and $p^{*}_{g,t}$ refers to the optimal power dispatching scheme at time $t$. The dispatch cost function is denoted by $C$, while $x_t$ and $\theta$ represent the model input at time $t$ and the parameters of the wind power forecasting model, respectively. The symbol $p_{\hat{w},t}$ denotes the forecasted value of wind power at time $t$, and $[p^{\hat{\alpha}}_{w,t}, p^{\hat{\alpha+1-\beta}}_{w,t}]$ represents the prediction interval for wind power forecasting at time $t$, where $\alpha$ and $\beta$ are parameters defining quantile value and confidence level of the interval. The value $p_{w,t}$ is derived from the median of the forecasting interval at time $t$, and $T$ denotes the total number of time steps in day-head dispatching.

In addressing the economic dispatch problem of power at time $t$ with wind power probability forecasting, the objective is to optimize the cost of the power generation schedule by accurately predicting wind power,

thereby achieving a cost-effective power system operation.

$$\sum_{t=1}^{T}\sum_{i=1}^{G}\left(a_i \cdot p_i^{g\,2} + b_i \cdot p_i^{g}\right)$$

subject to:

$$\begin{aligned}
&\sum_{j\in L} d_j = \sum_{i\in G} p_i^g + \sum_{i\in W} p_i^w \\
&i \in N : p_i + \sum_{j\in \text{in}} p_{ij} - \sum_{k\in \text{out}} p_{ki} = d_i \\
&-F \le PTDF(p_i - d_j) \le F \\
&p_i^{\min} \le p_i^g \le p_i^{\max} \quad \forall i \in G
\end{aligned} \tag{2}$$

In the context of this optimization problem, $p_i^g$ denotes the power output of generator $i$ at a specific time $t$ and $p_i^w$ denotes the wind power of wind generator $i$. The variable $d_j$ represents the demand at node $j$ at time $t$. The cost associated with operating generator $i$ is a quadratic function of time $t$. Let $G$ represent the set of generators, $W$ the set of wind generators, and $L$ the set of loads. The power balance constraint for each node is given by $i \in N : p_i + \sum_{j\in \text{in}} p_{ij} - \sum_{k\in \text{out}} p_{ki} = d_i$. The constant $F$ signifies the maximum permissible deviation of the power transfer distribution factor (PTDF) [43], denoted by $PTDF$, which is used to assess the impact of power flow on the electrical grid. Finally, $p_i^{\min}$ and $p_i^{\max}$ define the minimum and maximum power output limits for generator $i$, ensuring that the generator operates within safe and predefined bounds.

### 2.2. Prediction interval

Let $\bar{\mathcal{X}}_{t-1}$ denote the historical features up to time $t$. Since wind power forecasting is a covariate-based forecast, it can be decomposed into historical data $\mathcal{X}_{t-1}$ and additional features $\mathcal{X}'_{t-1}$, such that $\bar{\mathcal{X}}_{t-1} = (\mathcal{X}_{t-1}, \mathcal{X}'_{t-1})$.and let $\{Y_{t+l}, \dots, Y_{t+l}\}$ represent the forecasting result set with a lead time $l$. Probabilistic wind forecasting with a lead time $l$ involves specifying the conditional density function $f_{t+l|t}(Y_{t+l}|\bar{\mathcal{X}}_{t-1})$ and the conditional cumulative distribution function $F_{t+l|t}(Y_{t+l}|\bar{\mathcal{X}}_{t-1})$. Specifically, the $\alpha_{t+l}$-quantile result $p_{t+l}^{\alpha_{t+l}}$ is given by $F^{-1}_{t+l|t}(\alpha_{t+l})$, where the quantile probability $\alpha_{t+l}$ ranges within $[0,1]$. Consequently, the Prediction Interval with a NCP of $(1-\beta)\times 100\%$ can be constructed as follows:

$$PI_{t+l} = \left[p_{t+l}^{\alpha_{t+l}}, p_{t+l}^{\alpha_{t+l}+1-\beta}\right]. \tag{3}$$

### 2.3. Decision-risk objective function

Specifically, during the formulation of power dispatch plans, the forecasting module's overestimation and underestimation of actual new energy measurements correspond to the risks of power surplus and power deficit, respectively. This risk, denoted as $R_{base}$, can be quantified from a posterior evaluation perspective as the decision error between the ideal dispatch (the dispatch decision assuming no prediction error, i.e., based on actual measurements) and the actual dispatch. Additionally, we consider the risk that the actual value falls outside the predicted interval. Therefore, a risk assessment index $R$ is proposed to quantify the risks of power generation surplus and deficit based on precise measurements of new energy generation power.

$$\begin{aligned}
R_{\text{base}} =& \gamma_s \left[\sum_{k=1}^{L} d_k - \sum_{i=1}^{G} p_i^g - \sum_{j=1}^{W} p_j^{\hat{w}}\right]_+ + \\
& \gamma_e \left[\sum_{i=1}^{G} p_i^g + \sum_{j=1}^{W} p_j^{\hat{w}} - \sum_{k=1}^{L} d_k\right]_+
\end{aligned} \tag{4}$$

$$R = \begin{cases} R_{base} & \text{if } \hat{p}^{\alpha^*} \le p_w \le \hat{p}^{\alpha^*+1-\beta} \\ R_{base} + \frac{2(\hat{p}^{\alpha^*} - p_w)}{\beta} & \text{if } p_w < \hat{p}^{\alpha^*} \\ R_{base} + \frac{2(p_w - \hat{p}_t^{\alpha^*+1-\beta})}{\beta} & \text{if } \hat{p}^{\alpha^*+1-\beta} < p_w \end{cases} \tag{5}$$

In the formula, $[v]_+$ is defined as $\max\{v, 0\}$. $p_w$ means the actual value for wind. The risk weighting factors for the surplus and deficit parts of the wind generation power are $\gamma_e$ and $\gamma_s$, respectively, with the stipulation that $\gamma_s \gg \gamma_e$. The objective of the end-to-end dispatch decision model is to identify the optimal wind forecasting value that minimizes the risk of power imbalance in the grid due to the uncertainty of new energy forecasts.

$$\min_{\hat{p_w}\sim[\hat{p^{\alpha_i}}, \hat{p^{\alpha_i+1-\beta}}]} [R(p_g, p_g^*)]. \tag{6}$$

## 3. Methodology

The closed-loop framework proposed is depicted in Fig. 2. Initially, we pre-train the prediction module $M$, which is subsequently integrated into the framework. As illustrated, an agent, represented by a neural network (NN), selects the optimal quantile proportion $\alpha^*$ for the lower bound quantile under the current state. In our framework, historical data feature $\mathcal{X}_{t-1}$ serves as the state $s_t$ input to the agent. The risk function $R$ for the constructed Prediction Intervals serves as a pivotal feedback reward, seamlessly integrating the two tasks. The key elements of this formulation are concisely summarized as follows:

*State:* At time $t$, the Reinforcement Learning agent utilizes the historical data input feature vector $\mathcal{X}_{t-1} = \{x_{t-l}, x_{t-(l-1)}, \dots, x_{t-1}\}$ as the state, where $l$ denotes the look-back length. $\mathcal{X}_{t-1}$ remains part of the input to the Forecasting module $M$.

*State Transition Function:* To generate quantile forecasts for wind at time $t$, the agent's state, denoted by $\mathcal{X}_{t-1}$, encompasses both the current and historical data input features of the prediction module. Similarly, for time $t+1$, the agent's state is $\mathcal{X}_t = \{x_{t-l+1}, x_{t-l}, \dots, x_t\}$. The transition from $\mathcal{X}_{t-1}$ to $\mathcal{X}_t$ represents a mapping of continuous variables and is independent of any action taken. Given the complexity of modeling state transitions in high-dimensional continuous spaces, addressing the formulated Deep Reinforcement Learning problem with a model-based Reinforcement Learning approach, which entails learning a dynamic model, is particularly challenging. Consequently, we choose to tackle the aforementioned RL problem using a model-free method instead of the traditional model-based RL approach.

*Action:* To determine the risk-optimal quantile proportion within the continuous interval $(0, \beta)$, discretization of the probability proportion is necessary. The discrete action space $\mathcal{A}$ consists of a set of quantile proportions, with its cardinality $|\mathcal{A}|$ representing the number of quantile proportions. An increase in the number of actions results in finer-grained modeling of the $(0, \beta)$ range for the lower bound quantile proportion. For a given NCP, the action space is defined as follows:

$$\mathcal{A} = \left\{\frac{i \cdot \beta}{|\mathcal{A}|}\right\}. \tag{7}$$

The cardinality of the action space, denoted as $|\mathcal{A}| = n$, indicates the number of available actions. At any given time $t$, the optimal action $\alpha^*$ is chosen from this set. Although the number of actions affects the agent's policy, its impact is relatively minor because the difference between adjacent quantile proportions decreases as the size of the action space grows.

*Reward:* The reward function is defined as the negative of the risk value(Eq. (5)), as given follow:

$$r_t = -R. \tag{8}$$

*Agent:* The agent addresses the problem of selecting the optimal probability proportion by incrementally learning a policy that determines the optimal lower bound quantile proportion for constructing Decision-optimal Prediction Intervals.

*Environment:* The pre-trained forecasting module $M$, integrated within the environment, enables the prediction of multiple quantiles.The forecasting module and the forecasting process will be detailed in the subsequent section.

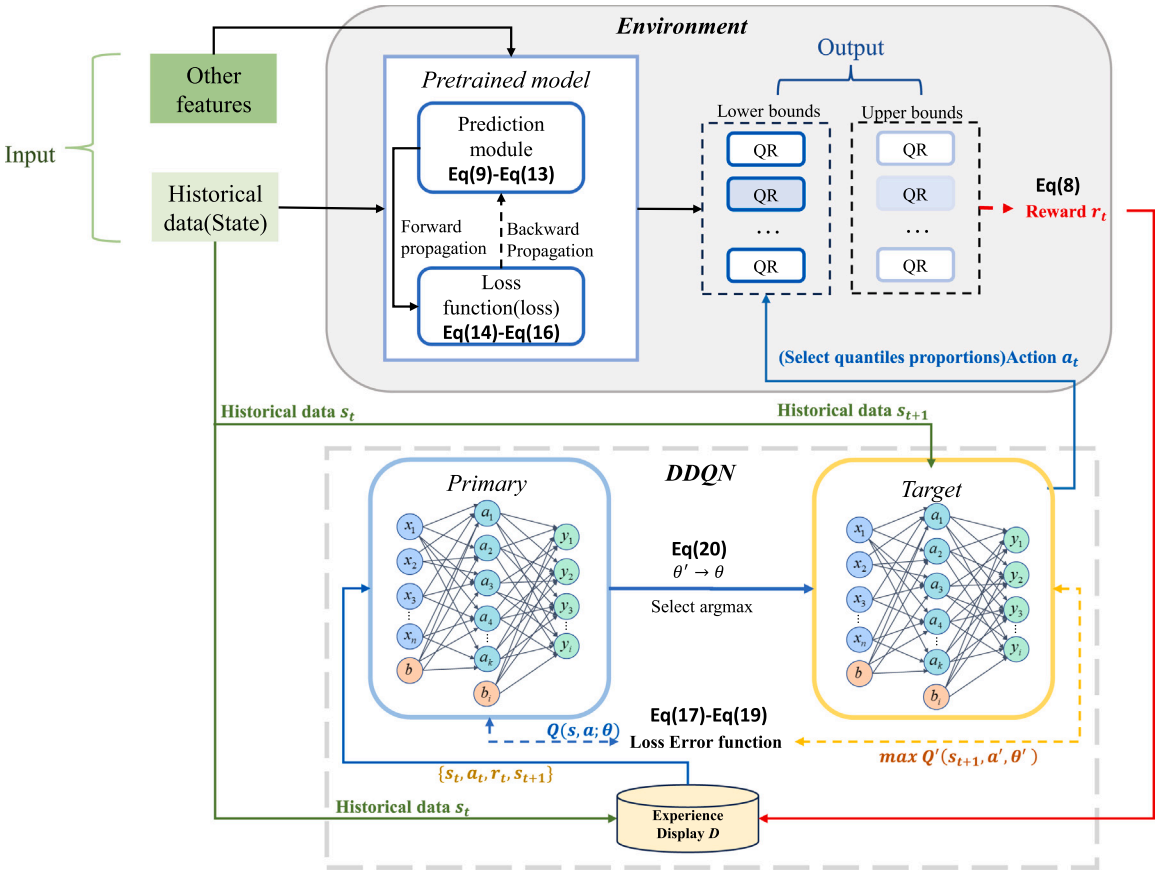



**Fig. 2.** The architecture of proposed method.

### 3.1. Forecasting module

In this study, we introduce a novel forecasting module(as shown in Fig. 3) that leverages joint quantile forecasting and multi-feature input to address the crossover issue in quantile forecasting.The model's effective avoidance of quantile crossing enhances the faithful of subsequent Decision Optimization Prediction Interval problems.This module integrates BiLSTM networks with attention mechanisms to process historical data $\mathcal{X}$ and other characteristics $\mathcal{X}'$ (for example, wind speed and temperature). The BiLSTM networks generate feature representations $h_t$ for historical data and $h'^i_t$ for other features, where $i$ indicates the $i$-th feature. These representations are input into self-attention and cross-attention mechanisms to capture temporal dependencies and cross-feature interactions, yielding attention outputs $Z_t^1$ and $Z_t^2$. The outputs are concatenated to form a comprehensive feature representation $Z_t$. Subsequently, a multi-layer perceptron (MLP) predicts the quantiles $\alpha_j$ based on $Z_t$, with $j$ indexing the different quantiles in the set $\{\alpha_1, \alpha_2, \ldots, \alpha_n\}$. The steps are as follow:

(1) Feature Input: This feature input step employs a sliding-window sampling method to extract input samples. To facilitate subsequent reinforcement learning decision modules and enhance prediction accuracy, the training set's multivariate features are categorized into historical data $\mathcal{X}_t = \{x_{t-l}, x_{t-l+1}, \ldots, x_{t-1}\} \in R^{L\times 1}$, where $l$ denotes the sequence length, and other features $\mathcal{X}'_{t-1} = \{x^i_{t-l}, x^i_{t-l+1}, \ldots, x^i_{t-1}\}_{i=1}^N \in R^{L\times N}$, where $N$ represents the number of additional features.

(2) BiLSTM Encoding Module: Owing to the superior feature extraction capabilities of the BiLSTM module, it is utilized as the Encoder for feature extraction. The formula is as follows:

$$F_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{9a}$$

$$I_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{9b}$$

$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \tag{9c}$$

$$O_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{9d}$$

$$C_t = F_t * C_{t-1} + I_t * \tilde{C}_t \tag{9e}$$

$$h_t = O_t * \tanh(C_t). \tag{9f}$$

In the equations, $W_f, W_i, W_c, W_o$ represent the weight matrices for the forget, input (partitioned into two parts), and output gates, respectively. $b_f, b_i, b_c, b_o$ denote the corresponding bias terms. $\sigma$ is the sigmoid function, and $h_{t-1}$ and $h_t$ are the output states of the previous and current time steps. BiLSTM extends conventional LSTM by processing data in both forward and reverse directions using two LSTM networks, enhancing performance on tasks requiring both retrospective and prospective context.

The Encoding module commences by initializing the hidden state $h_0$. It then employs an embedding function $f_0(\cdot)$, parameterized by $\theta$, to process the input data and encode the hidden state information across all time steps, projecting it into the embedding space. The specific computation process, given the historical data and additional features, is as follows:

$$\begin{aligned} h_t &= f_0(x_t, h_{t-1}) \in \mathbf{R}^D, \quad t = 1, 2, \ldots, l \\ h'^i_t &= f_0(x'^i_t, h'^i_{t-1}) \in \mathbf{R}^D, \quad t = 1, 2, \ldots, l. \end{aligned} \tag{10}$$

where $h'^i_{t-1}$, $h'^i_t$, $h_t$, and $h_{t-1}$ represent the hidden states at the previous and current time steps, respectively; $D$ denotes the dimension of the hidden state. By performing computations at each time step within the input time window, the hidden states for all time steps within the window can be derived as $H'_t = \{h'^i_1, h'^i_2, \ldots, h'^i_l\}_{i=1}^N$ and $H_t = \{h_1, h_2, \ldots, h_l\}$.

(3) Attention Module: The attention mechanism enhances the extraction of key features by assigning weights based on feature effectiveness. The attention mechanism requires three inputs: Query (Q), Key

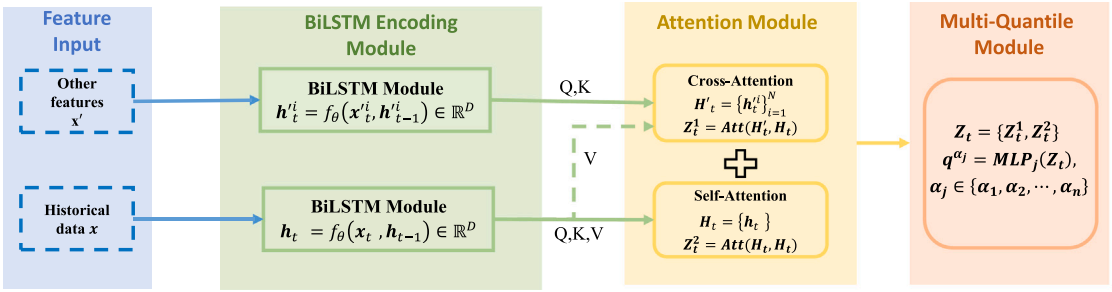



**Fig. 3.** The architecture of forecasting module.

(K), and Value (V) (as shown in Eq (11)). Based on this, we employ two attention modules to separately extract key historical features and additional features, thereby adequately modeling historical information.

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{Q \cdot K}{\sqrt{d_k}}\right) \cdot V. \tag{11}$$

$$Z_t^1 = \text{Cross-Attention}(H_t, H_t, H'_t) \tag{12a}$$

$$Z_t^2 = \text{Self-Attention}(H_t, H_t, H_t). \tag{12b}$$

(4) Multi-Quantile Module: The prediction output layer utilizes MLPs to predict a spectrum of quantiles concurrently.

$$q^{\alpha_j}_{\text{prediction}} = MLP_j(Z_t) \quad \alpha_j \in \{\alpha_1, \alpha_2, \ldots, \alpha_n\}. \tag{13}$$

In the equation, $q^{\alpha_j}_{\text{prediction}}$ represents the $j$-th predicted quantile. The vector $Z_t = \text{concat}(Z_t^1, Z_t^2)$. Here, $n$ denotes the total number of quantiles predicted, $\alpha_j$ indicates the $j$-th quantile value, and $MLP_j(\cdot)$ signifies the $j$-th Multilayer Perceptron used in the quantile prediction process.

The quantile loss function is a measure of the difference between predicted and actual values, commonly used in probabilistic forecasting and quantile regression. It evaluates model performance by calculating the quantile differences between predictions and actual values. The formula for the quantile loss function is given by:

$$L(\tau) = \max\left(\tau(y - \hat{y}), (\tau - 1)(y - \hat{y})\right). \tag{14}$$

Where $L(\tau)$ is the loss at quantile $\tau$, $y$ is the actual value, and $\hat{y}$ is the predicted value.

To address the common issue of quantile crossing in current quantile prediction models, this paper proposes a novel joint quantile loss $L_{JQR}$ based on multi-task learning. The model employs a Gaussian likelihood framework to estimate the quantile $y_{t(\alpha)}$ at the $\alpha$-th quantile point, defined as:

$$p(y_{t(\alpha)}|\hat{y}_{t(\alpha)}) = \mathcal{N}(\hat{y}_{t(\alpha)}, \sigma^2). \tag{15}$$

The losses for different quantiles are combined via a weighted sum. The joint quantile regression loss $L_{JQR}$ is derived from the negative log-likelihood of multiple quantiles:

$$L_{\text{sum}} = \sum_{e=1}^{E} \omega_e L_e, \tag{16a}$$

$$\mathcal{G} \propto \sum_{j=1}^{n}\left(\frac{1}{2\sigma_j^2}\right)\sum_{t=1}^{T}\rho_{\alpha_j}(y_t - \hat{y}_{t(\alpha_j)}) = L_{JQR},. \tag{16b}$$

where $E$ is the number of tasks, $\omega_e$ is the weight, and $L_e$ is the task loss.

### 3.2. DDQN learning

Given the high computational complexity of directly backpropagating gradients using a decision optimization problem as the loss function, our method leverages DDQN for decision optimization and probabilistic interval forecasting, capitalizing on the ability of reinforcement learning to generate optimal actions through policy gradients dynamically. The specific details are provided in Algorithm 1. It is important to note that our method is RL-free and can be integrated with other discrete reinforcement learning methods.

In the DDQN algorithm, the chosen action is evaluated within the target Q-network $\hat{Q}(s_{t+1}, a_{t+1}; \theta')$, and its update function is shown in Equation:

$$Y^{DDQN} = r + \gamma\hat{Q}\left(s_{t+1}, \arg\max_{a_{t+1}} Q(s_{t+1}, a_{t+1}; \theta); \theta'\right). \tag{17}$$

The update rule for the parameters $\theta$ in the DDQN algorithm is derived from the loss function, which quantifies the difference between the target and the predicted Q-values. The goal is to minimize the loss, leading to the update of the network parameters. The loss function is given by the following:

$$L(\theta) = \text{E}_{s_t, a_t, r, s_{t+1}}\left[\left(Y^{DDQN} - Q(s_t, a_t; \theta)\right)^2\right]. \tag{18}$$

To update the parameters $\theta$, we compute the gradient of the loss function with respect to $\theta$ and perform gradient descent. The update rule for $\theta$ is:

$$\theta \leftarrow \theta - \alpha\nabla_\theta L(\theta) \tag{19}$$

where $\alpha$ is the learning rate.

The gradient of the loss with respect to the parameters $\theta$ is computed as follows:

$$\nabla_\theta L(\theta) = \nabla_\theta\left[\left(Y^{DDQN} - Q(s_t, a_t; \theta)\right)^2\right]. \tag{20}$$

This gradient update adjusts $\theta$ to minimize the temporal difference error and improve the performance of the Q-network.

## 4. Experiment and case studies

### 4.1. Experimental setup

This study uses real-time wind power data, as published by Elia, the transmission system operator for Belgium. The dataset(Dataset 1), with a temporal resolution of 15 min, spans from January 2020 to December 2023.We also added another wind dataset (Dataset 2) to further validate the forecasting module [44].The proposed closed-loop method is also verified using the IEEE 6-bus and 30-bus cases.we assume that wind power generation does not exhibit spatial correlation. This assumption is based on the fact that transmission grids span vast geographical areas, and wind farms situated at different nodes possess relatively independent geographical conditions.The experiments are carried out using Python 3.9 and PyTorch on an RTX 4090 GPU.

(1) Forecasting module setting:In forecasting tasks,we compare the forecasting module proposed in this paper with the baseline models (Gaussian Process, BiLSTM, GRU, TCN, and Transformer-based [45]). The hyperparameters for our model are presented in Table 1. Our model, along with the comparative models, was utilized to forecast

**Algorithm 1** Decision Optimization PI Method based on DDQN

**Require:** Batch size $B$, learning rates $\eta_\alpha, \eta_Q$, parameters $\tau, \gamma$, NCP $(1-\beta)$, exploration rate $\epsilon_0$, target network update frequency $C$

Initialize the RL agent's parameters with random weights and pre-train the model $M$
Initialize the target network $Q'$ with the same weights as $Q$
Initialize exploration rate $\epsilon \leftarrow \epsilon_0$
**for** $t = 1, 2, \ldots T$ **do**
  Given the input state $\mathcal{X}_{t-1}$, with probability $\epsilon$, the agent selects a random action; otherwise, the agent selects $\alpha_t^* = \arg\max Q(\mathcal{X}_{t-1}, \alpha; Q)$
  Execute action in the environment: Select the quantile predictors $\hat{\alpha}_i$ and $\hat{\alpha}_i + 1 - \beta$ from the set $\{\alpha_1, \ldots, \alpha_i\} = M(\mathcal{X}_{t-1}, w)$
  Reveal the true value $y_t$
  Calculate the reward as defined in Eq (5) and update the input feature $\mathcal{X}_{t-1}$ for the next time-step based on the true value $y_t$
  Store the transition $(\mathcal{X}_{t-1}, r_t, \alpha_t^*, \mathcal{X}_t)$ in the replay buffer $D_t^Q$
  Sample a random batch of transitions from $D_t^Q$
  **for** each transition $(\mathcal{X}_j, r_j, \alpha_j^*, \mathcal{X}_{j+1})$ in the batch **do**
    Use the Deep Q-network $Q$ to select the action: $\alpha' = \arg\max Q(\mathcal{X}_{j+1}, \alpha; Q)$
    Use the target Q-network $Q'$ to evaluate the target Q-value: $y_j = r_j + \gamma Q(\mathcal{X}_{j+1}, \alpha'; Q')$
    Compute the loss: $\mathcal{L}$
    Update the Deep Q-network $Q$ using gradient descent with loss $\mathcal{L}$
  **end for**
  Every $C$ steps, update the target Q-network: $Q' \leftarrow Q$
  Decay the exploration rate: $\epsilon \leftarrow \max(\epsilon_{\min}, \epsilon \cdot \epsilon_{\text{decay}})$
**end for**

**Table 1**
Forecasting model hyperparameters configuration.

| Hyperparameter | Value |
|---|---|
| No. of neurons BiLSTM encoding layer | 64 |
| No. of BiLSTM network layers | 2 |
| No. attention heads | 4 |
| Learning rate | 0.01 |
| Batch size | 64 |
| Epochs | 50 |

wind power output of one point(15 min) at multi-quantiles: 0.05, 0.1, 0.2, 0.3, 0.4, 0.6, 0.7, 0.8, 0.9, and 0.95. Considering the dispatching issue, a prediction horizon of 15 min is adopted for intraday dispatch scheduling.Subsequently, we established confidence intervals for these forecasts corresponding to confidence levels of 20%, 40%, 60%, 80%, and 90%, which were calculated based on the quantile ranges [0.1, 0.9], [0.2, 0.8], [0.3, 0.7], [0.4, 0.6], and [0.05, 0.95], respectively. Following the construction of these intervals, we proceeded to calculate the evaluation metrics(Eq. (21)-Eq. (27)). The details of metrics can be seen in Appendix.

(2) DDQN setting:In decision tasks, the network hyperparameters of the DDQN algorithm are detailed in Table 2.The training parameter settings for the DDQN are as follows: The learning rate for the Primary-network, denoted as $\eta_\alpha$, is set to 0.001. Similarly, the learning rate for the Q-network, $\eta_Q$, is also set to 0.001. The discount factor, $\gamma$ is 0.9, and the target network update rate, $\tau$, is 0.1. The exploration rate starts at $\epsilon_0 = 0.9$ and decays by a factor of $\epsilon_{decay} = 0.99$ until it reaches a minimum value of $\epsilon_{min} = 0.01$. In decision tasks, we evaluated four benchmark methods: M1:Prediction-then-Optimization [43], which involves prediction followed by an optimization algorithm for decision-making; M2:Naive [43], which utilizes the previous data point as the predicted value for optimization, and two decision-oriented M3 [40] and M4 [41] forecasting methods.

**Table 2**
DDQN hyperparameters.

| Hyperparameter | Value |
|---|---|
| Batch size | 64 |
| No. of hidden layers | 2 |
| No. of neurons in the first hidden layer | 64 |
| No. of neurons in the second hidden layer | 128 |
| Learning rate | 0.001 |

To handle missing values within the dataset, we employ the strategy of filling them with the average of the preceding and succeeding data points. Furthermore, to mitigate the risk of gradient explosion, we apply Max-Min normalization to scale the data to the interval [0, 1]. Divide the data set in a 6:1:3 ratio to train, valid, and test.

### 4.2. Comparison of forecasting module

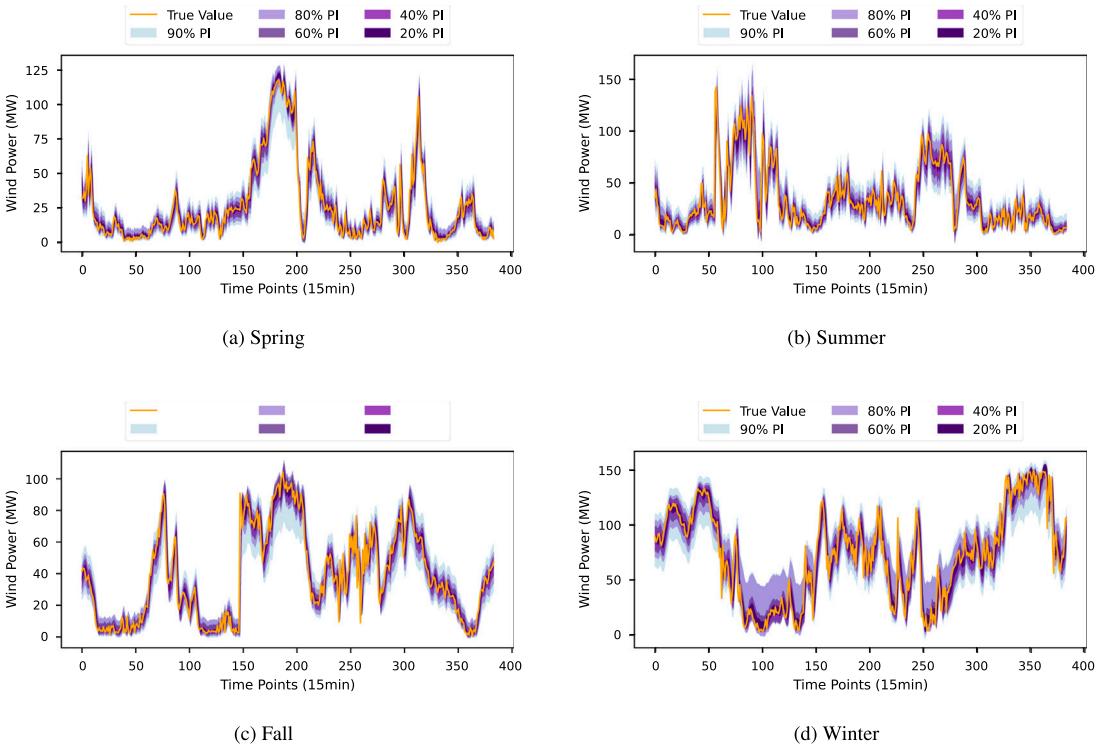
#### 4.2.1. Results and analysis

Table 3 shows the full forecasting results for two datasets. From Table 3, the forecasting model proposed in this paper demonstrates superior performance compared to the baseline models in one-point-ahead prediction. According to baseline results, it is evident that the BiLSTM/GRU models outperform the Autoformer/Transformer models in short-term prediction. This advantage is attributed to the Markovian modeling approach of BiLSTM/GRU, which is more effective in capturing short-term features.Compared to the BiLSTM model, our proposed model consistently demonstrates significant improvements across both datasets and all evaluation metrics. On Dataset1, our model reduces $|E_{ACD}|$ from 0.0438 to 0.0250, representing a 42.92% improvement. Similar enhancements are observed in $E_{AW}$, with a 22.11% decrease (from 31.998 to 24.716), and a 50.34% improvement in $E_{AIS}$ (from −0.9724 to −0.4835). The probabilistic metric $E_{MPICD}$ is also improved by 10.94%. For Dataset2, the improvements are even more pronounced. $|E_{ACD}|$ is reduced by 55.43% (from 0.0438 to 0.0195), while $E_{AW}$ is lowered by 61.48% (from 32.004 to 12.340). $E_{AIS}$ is enhanced by 41.17%, and $E_{MPICD}$ sees a 46.96% improvement. These enhancements are attributed to the effective use of the Attention mechanism, which improves the modeling of historical features.

Tables 4 and 5 show details of the mean Absolute Coverage Deviation (ACD) and Average Width (AW) for Dataset1. Table 4 shows significant ACD variation among models at different confidence intervals. BiLSTM-JQR and GRU-JQR exhibit notable fluctuations, particularly at the 20% and 60% intervals, indicating inconsistent coverage. For example, BiLSTM-JQR's ACD ranges from 4.2% at 20% confidence to −6.3% at 60%, while GRU-JQR's ACD varies from 3.8% to 6.7% across the same intervals. TCN-JQR shows a bias, with an underestimation of −9.4% at the 80% confidence interval. In contrast, our proposed model maintains ACD values close to zero, with the highest deviation being 2.5% at the 90% confidence interval, demonstrating consistent coverage. Table 5 reveals that our model achieves significantly lower AW values across all confidence intervals compared to baseline models. Specifically, at the 20% confidence interval, our model's AW value is 7.79, which is 37% lower than BiLSTM-JQR's and 73% lower than TCN-JQR's. At the 90% confidence interval, our AW value is 45.20, which is 25% lower than GRU-JQR's and 39% lower than TCN-JQR's. These results highlight our model's superior accuracy and sharpness in probabilistic forecasting. Fig. 4 visualizes the forecasting result of the proposed forecasting module across different seasons.

#### 4.2.2. Impact of quantile non-cross

This section aims to verify the effectiveness of the proposed quantile non-cross loss function.The Table 6 presents the $E_{qir}$ values(Eq. (23)) for four models — BiLSTM, GRU, TCN, and Our proposed — under two loss functions: $QR - E_{qir}$ and $JQR - E_{qir}$.$QR$(Eq. (14)) means the quantile loss and $JQR$(Eq. (16b)) means the non-cross quantile




(a) Spring (b) Summer

(c) Fall (d) Winter

**Fig. 4.** Seasonal Wind Power Prediction Intervals(4-day ahead).

**Table 3**
Comparison of prediction performances on Dataset1 and Dataset2.

| Datasets | Model | Metrics | | | |
|---|---|---|---|---|---|
| | | $\lvert E_{ACD}\rvert$ | $E_{AW}$ | $E_{AIS}$ | $E_{MPICD}$ |
| DataSet1 | Gaussian | 0.0769 | 52.341 | −2.3712 | 0.0551 |
| | GRU | 0.0498 | 35.166 | −1.1028 | 0.0581 |
| | BiLSTM | 0.0438 | 31.998 | −0.9724 | 0.0573 |
| | TCN | 0.0906 | 45.734 | −2.0723 | 0.0699 |
| | Autoformer | 0.0581 | 31.266 | −1.7864 | 0.0704 |
| | Transformer | 0.0604 | 34.512 | −1.5371 | 0.0647 |
| | Our model | **0.0250** | **24.716** | **−0.4835** | **0.0503** |
| DataSet2 | Gaussian | 0.0740 | 28.912 | −2.0378 | 0.0612 |
| | GRU | 0.0392 | 15.892 | −0.7941 | 0.0531 |
| | BiLSTM | 0.0385 | 14.329 | −0.8569 | 0.0545 |
| | TCN | 0.1041 | 26.523 | −2.4155 | 0.0704 |
| | Autoformer | 0.0441 | 16.785 | −1.9234 | 0.0679 |
| | Transformer | 0.0432 | 18.913 | −1.6454. | 0.0614 |
| | Our model | **0.0195** | **12.340** | **−0.5043** | **0.0459** |

**Table 4**
$E_{ACD}$ of probabilistic forecasting results (Dataset1).

| Model | $E_{ACD}^{(20\%)}$ | $E_{ACD}^{(40\%)}$ | $E_{ACD}^{(60\%)}$ | $E_{ACD}^{(80\%)}$ | $E_{ACD}^{(90\%)}$ |
|---|---|---|---|---|---|
| BiLSTM-JQR | 0.042 | 0.049 | −0.063 | −0.025 | 0.042 |
| GRU-JQR | 0.038 | −0.051 | 0.067 | −0.039 | −0.054 |
| TCN-JQR | 0.062 | −0.099 | 0.091 | −0.094 | 0.107 |
| Our | **0.020** | **−0.045** | **0.034** | **0.011** | **0.025** |

**Table 5**
$E_{AW}$ of probabilistic forecasting results (Dataset1).

| Model | $E_{AW}^{(20\%)}$ | $E_{AW}^{(40\%)}$ | $E_{AW}^{(60\%)}$ | $E_{AW}^{(80\%)}$ | $E_{AW}^{(90\%)}$ |
|---|---|---|---|---|---|
| BiLSTM-JQR | 12.36 | 15.61 | 27.34 | 45.36 | 59.32 |
| GRU-JQR | 16.82 | 26.24 | 28.30 | 43.21 | 60.26 |
| TCN-JQR | 29.41 | 32.31 | 35.42 | 57.72 | 73.81 |
| Our | **7.79** | **16.28** | **20.12** | **34.19** | **45.20** |

**Table 6**
$E_{qir}$ and $E_{time}$ values for different loss ways.

| Model | $QR - E_{qir}$ | $JQR - E_{qir}$ | $QR - E_{time}$ | $JQR - E_{time}$ |
|---|---|---|---|---|
| BiLSTM | 0.074 | 0 | 1.21 | 2.32 |
| GRU | 0.055 | 0 | 0.83 | 1.36 |
| TCN | 0.103 | 0 | 6.81 | 10.71 |
| Transformer | 0.091 | 0 | 89.23 | 126.64 |
| Autoformer | 0.098 | 0 | 65.41 | 115.03 |
| Our | 0.085 | 0 | 8.20 | 12.05 |

*Note:* $E_{time}$ means the time (s) for each epoch.

**Table 7**
Hyperparameter sensitiveness performances on Dataset1.

| Model | Metrics | | | |
|---|---|---|---|---|
| | $\lvert E_{ACD}\rvert$ | $E_{AW}$ | $E_{AIS}$ | $E_{MPICD}$ |
| Our model | $0.0250 \pm 0.003$ | $24.716 \pm 2$ | $-0.4835 \pm 0.08$ | $0.0503 \pm 0.005$ |

function.From the result, under the $QR - E_{qir}$ loss, the TCN model has the highest error rate at 0.103, while the GRU model shows the lowest error at 0.055. The BiLSTM and Our models fall in between with error rates of 0.074 and 0.085, respectively. The results indicate that all models exhibit zero $E_{qir}$ under the $JQR - E_{qir}$ loss function, suggesting that $JQR$ can solve the quantile-cross issue.

### 4.2.3. Impact of hyperparameter

This section illustrates the robustness of our forecasting module to hyperparameters via random hyperparameter experiments and optimization. Specifically, we performed ten experiments using random seeds (111, 2025, 2024, 2222), varying BiLSTM layers from 2 to 4, and Attention heads between 4 and 8. The outcomes, presented in Table 7, demonstrate the forecasting model's robustness to hyperparameter variations. Subsequently, we employed hyperparameter tuning, targeting the $E_{ACD}$ metric. As shown in Table 8, Bayesian optimization was more time-efficient and Grid-Search method can achieve accurate result.

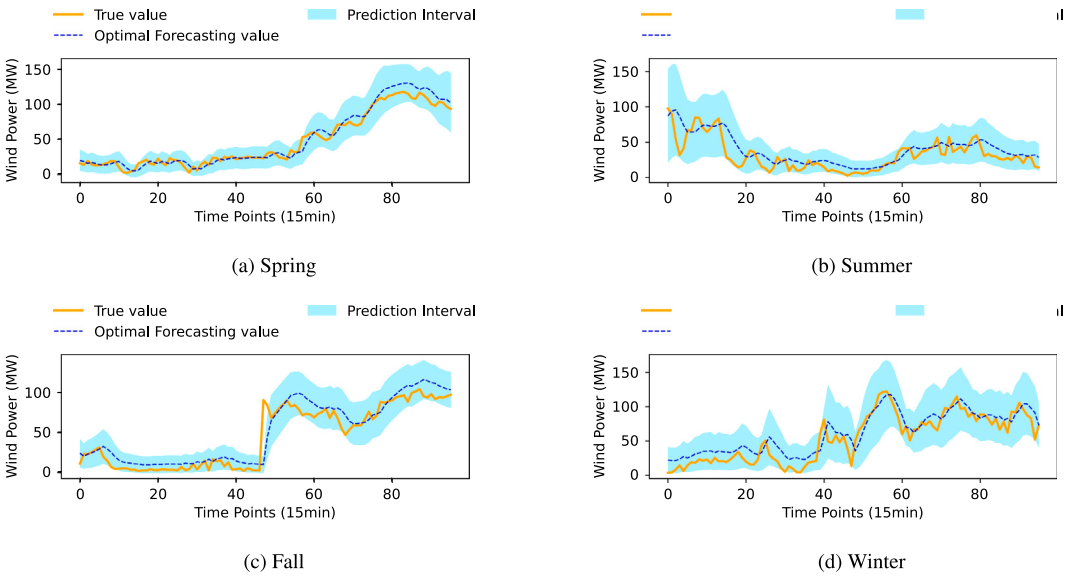



**Fig. 5.** The Wind Power Decision-Optimal Prediction Intervals(1-day ahead).

**Table 8**
Comparison of Grid-Search and Bayesian-Optimization.

| | $E_{ACD}/E_{AW}$ | Time (s) |
|---|---|---|
| Grid-Search | 0.0184/30.243 | 1264 |
| Bayesian-Optimization | 0.0196/28.904 | 532 |

**Table 9**
Decision results in test set (IEEE 6-bus).

| Model | Average cost ($) | Average risk |
|---|---|---|
| M1 [43] | 21 048 | 47.47 |
| M2 [43] | 21 354 | 66.00 |
| M3 [40] | 21 006 | 43.20 |
| M4 [41] | 20 741 | 39.28 |
| Our | 20 667 | 37.56 |

**Table 10**
Decision results in test set (IEEE 30-bus).

| Model | Average cost ($) | Average risk |
|---|---|---|
| M1 [43] | 35 218 | 119.22 |
| M2 [43] | 36 361 | 145.43 |
| M3 [40] | 36 204 | 138.76 |
| M4 [41] | 34 970 | 98.91 |
| Our | 34 703 | 92.69 |

**Table 11**
Comparison on averaged operating cost based on different RL approaches(IEEE 6-bus).

| Algorithm | DDQN | DQN | PPO | SAC |
|---|---|---|---|---|
| Average cost ($) | 20 667 | 21 032 | 20 930 | 22 458 |

### 4.3. Case study

#### 4.3.1. IEEE 6-bus

In this study, the action space $|\mathcal{A}|$ is defined as 10. The proposed method's efficacy is demonstrated through experiments on the IEEE 6-bus system. The setup includes thermal power units at nodes 1, 2, 3 and wind power units at node 4. In the numerical experiment, the parameters $\gamma_s$ and $\gamma_e$ are set to 50 and 0.5, respectively, with a wind power capacity of 150 MW. The decision time for real-time dispatch is set at 15 min.

Table 9 presents the results of the IEEE 6-bus decision tasks. Specifically, M1 and M2 are traditional Prediction-then-Optimization approaches. Compared with decision-oriented methods (M3, M4 and Our), they achieve higher average costs and risks. M3 which is restricted to linear models, performs worse than M4, which employs Bayesian methods. Meanwhile, the method proposed in this paper, which utilizes reinforcement learning, outperforms the baseline models due to its powerful policy learning capability. The proposed approach achieves a reduction of 0.36% in Average Cost and a decrease of 4.38% in Average Risk. Fig. 5 shows the visualization of decision-optimal results.

#### 4.3.2. IEEE 30-bus

We also validated the proposed framework using the IEEE 30-bus system, integrating two 150MW wind farms to simulate a power system with significant renewable energy penetration. The forecasting module was trained individually for each wind farm to account for location-specific wind variability. The number of forecasting modules, $|\mathcal{A}|$ is set to 10 for each model, ensuring adaptability to multiple wind farms while maintaining accuracy and reliability.

Table 10 compares the decision results of different models on the IEEE 30-bus test system. The proposed model demonstrates superior performance in both average cost and average risk value.Compared to the state-of-the-art method M4, the proposed approach achieves a reduction of 0.76% in Average Cost and a decrease of 6.29% in Average Risk.This indicates that the method proposed in this paper possesses scalability. This scalability further highlights its potential for broader implementation, offering enhanced efficiency and reliability in renewable energy integration.

#### 4.3.3. Comparison of RL algorithms

In this section, we compare the performance of the DDQN algorithm employed in this paper with those of DQN, Proximal Policy Optimization(PPO), and Soft Actor-Critic(SAC). The results ( Table 11) presented in the table indicate that DDQN, DQN, and PPO, due to their advantages in handling discrete decision spaces, significantly reduce operational costs compared to traditional methods. In contrast, SAC designed for continuous spaces, exhibits suboptimal policy learning outcomes. These findings not only demonstrate the superior learning effectiveness of DDQN but also validate that the framework proposed in this paper can accommodate other RL algorithms.

**Table 12**
Comparison on train and inference time (s) (IEEE 6-bus).

| Algorithm | Our | M3 | M4 | M2 |
|---|---|---|---|---|
| Train/Inference (s) | 1865/1.3 | 327/1.5 | 286/1.7 | –/2.6 |

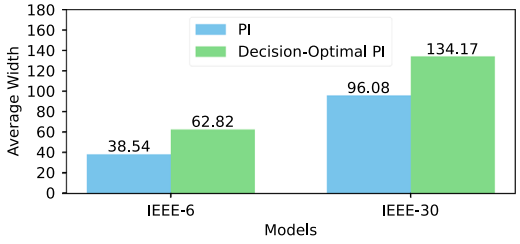



**Fig. 6.** The Average width of Forecasting results.

### 4.4. Discussion

This section primarily focuses on the discussion of the algorithmic complexity of the proposed method and the inconsistency in Forecasting-Decision.The proposed framework's complexity is driven by the BiLSTM module ($O(n \cdot d \cdot h)$) and attention mechanism ($O(n^2 \cdot d)$), with DDQN contributing $O(m \cdot t)$. The overall complexity is $O(n \cdot d \cdot h + n^2 \cdot d + m \cdot t)$. Variables $n$, $d$, $h$, $m$, and $t$ represent sequence length, feature dimension, hidden state size, training epochs, and decision time steps, respectively. Table 12 indicates that, although reinforcement learning requires longer training times, it can provides dynamic adaptability that other methods lack.

Fig. 6 presents a comparison of prediction quality between the decision-optimal prediction interval(NCP 90%) and the standard prediction interval. As illustrated in the table, the average width of the decision-optimal prediction interval is significantly larger than that of the standard prediction interval, indicating broader uncertainty estimates. This observation underscores the inconsistency of optimization objectives between decision-making and prediction tasks. While standard prediction intervals focus on minimizing statistical errors, decision-optimal intervals prioritize actionable insights, often requiring wider bounds to account for potential risks and uncertainties.

## 5. Conclusion

This paper introduces a reinforcement learning adaptive decision-making optimal interval prediction framework specifically designed for wind power prediction to mitigate the impact of wind power uncertainty on the economic dispatch of power systems. The problem is modeled as a two-layer optimization task, which incorporates the economic scheduling function, prediction intervals, and a decision-objective function aimed at minimizing grid power imbalance risk. The methodology employs a closed-loop framework powered by the DDQN algorithm. This framework is enhanced by a prediction module that leverages BiLSTM networks integrated with attention mechanism to refine historical data processing and boost prediction accuracy. The forecasting results are further optimized through reinforcement learning to determine the optimal quantile ratio and choices for the prediction interval. Experimental validation was conducted using real-world wind power data provided by the Belgian company Elia and tested on IEEE 6-bus and 30-bus cases. The test results prove the validity of the framework proposed in this paper.

The method proposed in this paper still has the following limitations: when applied to large power grids with high penetration of renewable energy, the computation time is excessively long. Moreover, this method does not take into account the impact of wind power data under extreme weather conditions on the security of the power grid.In the follow-up research, we will consider the spatial correlation of wind power generation and the topological correlation of the grid. Additionally, this study focuses solely on the decision-making risks associated with the uncertainties in new energy predictions affecting the power system. Future research will consider extreme weather potential risks arising from multiple uncertainties within the power system, causing historical data quality problems from extreme weather, and analyze the different constraints to enhance the operational efficiency of the power system comprehensively.

## CRediT authorship contribution statement

**Chenghan Li:** Writing – review & editing, Writing – original draft, Methodology, Investigation, Formal analysis, Conceptualization. **Ye Guo:** Writing – review & editing, Supervision, Project administration, Methodology, Funding acquisition, Conceptualization. **Yinliang Xu:** Writing – review & editing, Writing – original draft, Supervision, Resources, Methodology, Investigation, Funding acquisition, Conceptualization.

## 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.

## Acknowledgment

This work was supported by the National Key Research and Development Program, Grant No. 2024YFB4206500.

## Appendix

### A.1. Metrics

**Average Coverage Deviation (ACD)** is used to evaluate the accuracy of quantile forecasts by measuring the deviation between the actual coverage rate and the target coverage rate $\alpha$. It is defined as $E_{ACD}^{(\alpha)} = \alpha - b^{(\alpha)}$, where $b^{(\alpha)}$ is the average coverage rate, calculated as $b^{(\alpha)} = \frac{1}{T}\sum_{t=1}^{T} c_t^{(\alpha)}$. Here, $c_t^{(\alpha)}$ is an indicator function:

$$c_t^{(\alpha)} = \begin{cases} 1, & \text{if } y_t \le \hat{y}_{t(\alpha)} \\ 0, & \text{if } y_t > \hat{y}_{t(\alpha)}. \end{cases} \tag{21}$$

When the observed value $y_t$ is less than or equal to the predicted quantile $\hat{y}_{t(\alpha)}$, $c_t^{(\alpha)} = 1$; otherwise, it is 0. The parameter $b^{(\alpha)}$ represents the average value of the indicator function over the entire time series. The target coverage rate $\alpha$ is the desired proportion of observations below the predicted quantile. A value of $E_{ACD}^{(\alpha)}$ close to zero indicates better forecast reliability, while a positive value suggests under-coverage and a negative value suggests over-coverage.

**Average Width (AW) Sharpness:** The sharpness of a forecast is measured by the Average Width (AW) of the forecast interval, reflecting the concentration of the predicted probability distribution. A wider interval increases uncertainty and decision-making costs, reducing operational efficiency. The AW, denoted as $E_{AW}$, is calculated as:

$$E_{AW} = \frac{1}{T}\sum_{s=1}^{T}\left(q_{s(\alpha_{up})} - q_{s(\alpha_{down})}\right). \tag{22}$$

where $q_{s(\alpha_{up})}$ and $q_{s(\alpha_{down})}$ are the upper and lower bounds of the forecast interval at quantile points $\alpha_{up}$ and $\alpha_{down}$, respectively. A lower $E_{AW}$ indicates a more concentrated distribution and better forecast sharpness.

**Quantile Intersection Rate (QIR):** QIR measures the proportion of times predicted quantiles violate the monotonicity property of a cumulative distribution function (CDF). When a lower quantile's predicted

value exceeds a higher quantile's, it indicates a CDF violation. The QIR, denoted as $E_{qir}$, is calculated as:

$$E_{qir} = \frac{1}{T}\sum_{t=1}^{T} p(t). \tag{23}$$

where $p(t)$ is an indicator function:

$$p(t) = \begin{cases} 1, & \hat{y}_{t(\alpha_u)} < \hat{y}_{t(\alpha_v)} \\ 0, & \hat{y}_{t(\alpha_u)} \geq \hat{y}_{t(\alpha_v)}. \end{cases} \tag{24}$$

Here, $\hat{y}_{t(\alpha_u)}$ and $\hat{y}_{t(\alpha_v)}$ are predicted quantiles at time $t$ for quantile points $\alpha_u > \alpha_v$. If a violation occurs at time $t$, $p(t) = 1$; otherwise, $p(t) = 0$. A QIR close to 0 indicates the model satisfies CDF monotonicity, while a higher value indicates frequent violations.

**Average Interval Score (AIS):** Interval Score (IS) is a practical tool that provides a comprehensive consideration of coverage rate and interval width. The definition of the interval score $S^{(\alpha)}(x_i)$ is as follows:

$$S^{(\alpha)}(x_i) = \begin{cases} -2\alpha\zeta_i^{(\alpha)} - 4(L^{(\alpha)}(x_i) - p_i), & \text{if } p_i < L^{(\alpha)}(x_i) \\ -2\alpha\zeta_i^{(\alpha)}, & \text{if } p_i \in [L^{(\alpha)}(x_i), U^{(\alpha)}(x_i)] \\ -2\alpha\zeta_i^{(\alpha)} - 4(p_i - U^{(\alpha)}(x_i)), & \text{if } p_i > U^{(\alpha)}(x_i). \end{cases} \tag{25}$$

where $\zeta_i^{(\alpha)}$ is the width of the $i$-th PI:

$$\zeta_i^{(\alpha)} = U^{(\alpha)}(x_i) - L^{(\alpha)}(x_i), \tag{26a}$$

$$\overline{S^{(\alpha)}} = \frac{1}{T}\sum_{i=1}^{T} S^{(\alpha)}(x_i). \tag{26b}$$

When the target is not within the PI coverage interval, a certain penalty is given. A higher AIS value indicates better quality of the prediction intervals.

**Mean Prediction Interval Center Deviation (MPICD):** The MPICD is a measure used to assess the quality of prediction intervals, particularly in probabilistic forecasting. It evaluates how well the prediction intervals cover the actual observations. The MPICD is calculated as the average of the absolute differences between the midpoint of the prediction intervals and the actual observations, normalized by the number of test cases. The formula for MPICD is:

$$\text{MPICD} = \frac{1}{T}\sum_{i=1}^{T}\left|\frac{\hat{U}^{(\alpha)}(x_i) + \hat{L}^{(\alpha)}(x_i)}{2} - p_i\right|. \tag{27}$$

where $\hat{U}^{(\alpha)}(x_i)$ and $\hat{L}^{(\alpha)}(x_i)$ are the upper and lower bounds of the prediction interval for the $i$-th observation, respectively, and $p_i$ is the actual observation.

## Data availability

Data will be made available on request.

## References


[1] Guo Jianbo, Jing Yiran, Hou Weilin, Wang Tiezhu, Ma Shicong, He Guoqing. Demands and challenges of energy storage technology for future power system. Energy Internet 2024;1(2):116–22.

[2] Cheng Runkun, Yang Di, Liu Da, Zhang Guowei. A reconstruction-based secondary decomposition-ensemble framework for wind power forecasting. Energy 2024;308:132895.

[3] Xie Xiangmin, Ding Yuhao, Sun Yuanyuan, Zhang Zhisheng, Fan Jianhua. A novel time-series probabilistic forecasting method for multi-energy loads. Energy 2024;306:132456.

[4] Yakoub Ghali, Mathew Sathyajith, Leal Joao. Intelligent estimation of wind farm performance with direct and indirect 'point'forecasting approaches integrating several NWP models. Energy 2023;263:125893.

[5] Cabello-López Tomás, Carranza-García Manuel, Riquelme José C, García-Gutiérrez Jorge. Forecasting solar energy production in Spain: A comparison of univariate and multivariate models at the national level. Appl Energy 2023;350:121645.

[6] Zhao Changfei, Wan Can, Song Yonghua. Cost-oriented prediction intervals: On bridging the gap between forecasting and decision. IEEE Trans Power Syst 2021;37(4):3048–62.

[7] Zhao Changfei, Wan Can, Song Yonghua. Operating reserve quantification using prediction intervals of wind power: An integrated probabilistic forecasting and decision methodology. IEEE Trans Power Syst 2021;36(4):3701–14.

[8] Bertsimas Dimitris, Litvinov Eugene, Sun Xu Andy, Zhao Jinye, Zheng Tongxin. Adaptive robust optimization for the security constrained unit commitment problem. IEEE Trans Power Syst 2012;28(1):52–63.

[9] Qiu Haifeng, Gu Wei, Xu Yinliang, Wu Zhi, Zhou Suyang, Wang Jianhua. Interval-partitioned uncertainty constrained robust dispatch for AC/DC hybrid microgrids with uncontrollable renewable generators. IEEE Trans Smart Grid 2018;10(4):4603–14.

[10] Yang Mao, Jiang Yuxi, Xu Chuanyu, Wang Bo, Wang Zhao, Su Xin. Day-ahead wind farm cluster power prediction based on trend categorization and spatial information integration model. Appl Energy 2025;388:125580.

[11] Ge Chang, Yan Jie, Song Weiye, Zhang Haoran, Wang Han, Li Yuhao, et al. Middle-term wind power forecasting method based on long-span NWP and microscale terrain fusion correction. Renew Energy 2025;240:122123.

[12] Liu Chenyu, Zhang Xuemin, Mei Shengwei, Zhen Zhao, Jia Mengshuo, Li Zheng, et al. Numerical weather prediction enhanced wind power forecasting: Rank ensemble and probabilistic fluctuation awareness. Appl Energy 2022;313:118769.

[13] Zhang Hao, Yan Jie, Liu Yongqian, Gao Yongqi, Han Shuang, Li Li. Multi-source and temporal attention network for probabilistic wind power prediction. IEEE Trans Sustain Energy 2021;12(4):2205–18.

[14] Jaseena KU, Kovoor Binsu C. Decomposition-based hybrid wind speed forecasting model using deep bidirectional LSTM networks. Energy Convers Manage 2021;234:113944.

[15] Liu Ming-De, Ding Lin, Bai Yu-Long. Application of hybrid model based on empirical mode decomposition, novel recurrent neural networks and the ARIMA to wind speed prediction. Energy Convers Manage 2021;233:113917.

[16] Wang Lei, Wang Xinyu, Zhao Zhongchao. Mid-term electricity demand forecasting using improved multi-mode reconstruction and particle swarm-enhanced support vector regression. Energy 2024;304:132021.

[17] Neshat Mehdi, Nezhad Meysam Majidi, Mirjalili Seyedali, Piras Giuseppe, Garcia Davide Astiaso. Quaternion convolutional long short-term memory neural model with an adaptive decomposition method for wind speed forecasting: North aegean islands case studies. Energy Convers Manage 2022;259:115590.

[18] Yuzgec Ugur, Dokur Emrah, Balci Mehmet. A novel hybrid model based on empirical mode decomposition and echo state network for wind power forecasting. Energy 2024;300:131546.

[19] Lv Sheng-Xiang, Wang Lin. Multivariate wind speed forecasting based on multi-objective feature selection approach and hybrid deep learning model. Energy 2023;263:126100.

[20] Meng Anbo, Zhu Zibin, Deng Weisi, Ou Zuhong, Lin Shan, Wang Chenen, et al. A novel wind power prediction approach using multivariate variational mode decomposition and multi-objective crisscross optimization based deep extreme learning machine. Energy 2022;260:124957.

[21] Neshat Mehdi, Nezhad Meysam Majidi, Abbasnejad Ehsan, Mirjalili Seyedali, Groppi Daniele, Heydari Azim, et al. Wind turbine power output prediction using a new hybrid neuro-evolutionary method. Energy 2021;229:120617.

[22] Liu Tongchui, Pan Wenxia, Zhu Zhu, Liu Mingyang. Two-stage risk dispatch for combined electricity and heat system under extreme weather events. Int J Electr Power Energy Syst 2024;157:109812.

[23] Zhao Zhuoli, Xu Jiawen, Lei Yu, Liu Chang, Shi Xuntao, Lai Loi Lei. Robust dynamic dispatch strategy for multi-uncertainties integrated energy microgrids based on enhanced hierarchical model predictive control. Appl Energy 2025;381:125141.

[24] Chen Yuejiang, Xiao Jiang-Wen, Wang Yan-Wu, Luo Yunfeng. Non-crossing quantile probabilistic forecasting of cluster wind power considering spatio-temporal correlation. Appl Energy 2025;377:124356.

[25] Cui Wenkang, Wan Can, Song Yonghua. Ensemble deep learning-based non-crossing quantile regression for nonparametric probabilistic forecasting of wind power generation. IEEE Trans Power Syst 2022;38(4):3163–78.

[26] Cannon Alex J. Quantile regression neural networks: Implementation in r and application to precipitation downscaling. Comput Geosci 2011;37(9):1277–84.

[27] Al-Gabalawy Mostafa, Hosny Nesreen S, Adly Ahmed R. Probabilistic forecasting for energy time series considering uncertainties based on deep learning algorithms. Electr Power Syst Res 2021;196:107216.

[28] Liu Yanli, Wang Junyi, Liu Liqi. Physics-informed reinforcement learning for probabilistic wind power forecasting under extreme events. Appl Energy 2024;376:124068.

[29] Zhao Jing, Guo Yiyi, Lin Yihua, Zhao Zhiyuan, Guo Zhenhai. A novel dynamic ensemble of numerical weather prediction for multi-step wind speed forecasting with deep reinforcement learning and error sequence modeling. Energy 2024;302:131787.

[30] Gubareva Mariya, Shafiullah Muhammad, Teplova Tamara. Cross-quantile risk assessment: The interplay of crude oil, artificial intelligence, clean tech, and other markets. Energy Econ 2025;141:108085.

[31] Wadström Christoffer, Hedström Axel. Assessing the quantile dependence and interconnectedness of electricity utilisation across Swedish industrial sectors. Energy 2025;135196.
[32] Wang HZ, Wang GB, Li GQ, Peng JC, Liu YT. Deep belief network based deterministic and probabilistic wind speed forecasting approach. Appl Energy 2016;182:80–93.
[33] Li Dan, Zhang Yuanhang, Yang Baohua, Wang Q. Short time power load probabilistic forecasting based on constrained parallel-LSTM neural network quantile regression mode. Power Syst Technol 2021;45(4).
[34] Moreno-Pino Fernando, Olmos Pablo M, Artés-Rodríguez Antonio. Deep autoregressive models with spectral attention. Pattern Recognit 2023;133:109014.
[35] Luo Xing, Zhang Dongxiao, Zhu Xu. Deep learning based forecasting of photovoltaic power generation by incorporating domain knowledge. Energy 2021;225:120240.
[36] Carriere Thomas, Kariniotakis George. An integrated approach for value-oriented energy forecasting and data-driven decision-making application to renewable energy trading. IEEE Trans Smart Grid 2019;10(6):6933–44.
[37] Liu Yixin, Shi Haoqi, Guo Li, Xu Tao, Zhao Bo, Wang Chengshan. Towards long-period operational reliability of independent microgrid: A risk-aware energy scheduling and stochastic optimization method. Energy 2022;254:124291.
[38] Stratigakos Akylas, Camal Simon, Michiorri Andrea, Kariniotakis Georges. Prescriptive trees for integrated forecasting and optimization applied in trading of renewable energy. IEEE Trans Power Syst 2022;37(6):4696–708.
[39] Chen Xianbang, Yang Yafei, Liu Yikui, Wu Lei. Feature-driven economic improvement for network-constrained unit commitment: A closed-loop predict-and-optimize framework. IEEE Trans Power Syst 2021;37(4):3104–18.
[40] Wang Jingxing, Chung Seokhyun, AlShelahi Abdullah, Kontar Raed, Byon Eunshin, Saigal Romesh. Look-ahead decision making for renewable energy: A dynamic "predict and store" approach. Appl Energy 2021;296:117068.
[41] Donti Priya, Amos Brandon, Kolter J Zico. Task-based end-to-end model learning in stochastic optimization. Adv Neural Inf Process Syst 2017;30.
[42] Elmachtoub Adam N, Grigas Paul. Smart "predict, then optimize". Manag Sci 2022;68(1):9–26.
[43] Zhang Jialun, Wang Yi, Hug Gabriela. Cost-oriented load forecasting. Electr Power Syst Res 2022;205:107723.
[44] Ding Y. Inland-offshore wind farm dataset1. 2019.
[45] Sun Shilin, Liu Yuekai, Li Qi, Wang Tianyang, Chu Fulei. Short-term multi-step wind power forecasting based on spatio-temporal correlations and transformer neural networks. Energy Convers Manage 2023;283:116916.