symmetry

MDPI

Article

A High–Mid–Low Triangular Echo State Network with Attention and Improved Beaver Behavior Optimizer for Photovoltaic Power Forecasting

Ying Yao, Shuxian Lun * and Ming Li

College of Control Science and Engineering, Bohai University, Jinzhou 121013, China; 2025008063@bhu.edu.cn (Y.Y.); liming1@qymail.bhu.edu.cn (M.L.)
* Correspondence: lunshuxian@qymail.bhu.edu.cn

Abstract

Proper photovoltaic (PV) power forecasting supports grid stability, efficient dispatch, and economic performance. While most forecasting models use rigid time smoothing, which eliminates multi-scale volatility, in this study, we propose a High–Mid–Low Triangular Echo State Network (HMLT-ESN) optimized by a Quasi-Oppositional Learning Cyclonic Foraging Beaver Behavior Optimizer (QCBBO). HMLT-ESN partitions photovoltaic sequences into high-, mid-, and low-frequency components using Variational Mode Decomposition (VMD). Each component is processed by a sub-reservoir, and each sub-reservoir uses different activation functions: TreX for transients, tanh for periodicities, and sigmoid for trends. Rather than using a fixed leakage rate, we introduce an attention-driven diagonal matrix that adjusts each timescale blockwise according to its instantaneous activation energy. Recurrent topology is restricted to an equilateral triangular lattice that provides an analytically tractable spectral radius. The circle map achieves better coverage over the normal input domain and feature discrimination, while the QCBBO runs autonomously to avoid local optima. Tested on a 20 MWp plant in northern China, the model predicts 15 min power at 15 min resolution using 90 min windows. The HMLT-ESN reduces Mean Absolute Error (MAE) by 41.57% and Normalized Mean Square Error (NMSE) by 51.73% compared to ESN, acting as a robust dispatch tool for renewable-heavy power systems.

Keywords: photovoltaic power forecasting; echo state network; variational mode decomposition; beaver behavior optimization; multi-frequency feature learning

Check for updates

Academic Editor: Vasilis K. Oikonomou

Received: 13 October 2026
Revised: 12 November 2026
Accepted: 16 November 2026
Published: 18 November 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

The rapid depletion of fossil fuels and worsening environmental pollution have made photovoltaic (PV) power generation a key element in the global energy transition. Photovoltaic power offers several advantages: it is clean and abundant, and its costs have dropped sharply in recent years [1], with China being the largest producer and user of photovoltaic systems. Photovoltaic output is dependent on weather conditions, as cloud cover, humidity, and solar irradiance are subject to uncertainty and volatility. For grid operators, this volatility causes problems with real-time balancing and the risk of unsafe operation. In ultra-short-term forecasting, better accuracy can directly reduce the cost of frequency regulation services, which plants have to purchase for deviations in outputs, and can also reduce overall operational costs [2].

PV power prediction methods are categorized into three main types: physical models [3], statistical methods [4], and Artificial Intelligence-based (AI-based) approaches.

Physical models use physical formulas, solar radiation transmission models, and numerical weather forecasts to estimate potential output. In stable weather conditions, these models have high accuracy, fast computation times, and clear physical interpretability. However, they depend heavily on forecast accuracy and require detailed plant parameters, and their local correction ability is limited by fast-changing events such as cloud transients.

Statistical methods, such as regression and autoregressive models, extract empirical relationships from historical data and are simple and effective when data patterns and trends are stable. However, they do not capture nonlinear properties and are not scaled well for large datasets. With the rapid advancement of machine learning and deep learning, AI-based methods are now standard in PV forecasting because they automatically learn complex nonlinear relationships without hand-defined physical formulas [5]. Among these, the "decomposition–prediction–combination" strategy has proven particularly effective [6].

Deep learning architectures have recently been explored by Zi et al. [7], who proposed an adaptive sparse self-attention Transformer with a dual-branch attention mechanism to reduce computational overhead. However, since the standard Transformer still processes all time steps uniformly, it fails to distinguish rapid changes from slow changes, while also increasing noise and masking important features. Liu et al. [8] used a deep belief network with a Takagi–Sugeno fuzzy model, which handles high-dimensional meteorological inputs effectively but lacks explicit multi-scale decomposition. High-frequency noise is not distinct from low-frequency patterns, and performance decreases under rapidly changing weather. Wadood et al. [9] proposed a fractional-order Whale Optimization Algorithm for support vector regression (SVR) parameter tuning. While their optimizer achieved a better exploration–exploitation trade-off, SVR is shallow and cannot take into account long-range dependencies in PV sequences. In addition, optimization is independent from internal dynamics. Zhang et al. [10] proposed a combination of Ensemble Empirical Mode Decomposition (EEMD), Variational Mode Decomposition (VMD) and a Bidirectional Gated Recurrent Unit (BiGRU). Although their two-stage decomposition scheme works reasonably well, all decomposed components are processed by the same BiGRU structure. High-frequency transients and low-frequency trends get the same gating update, meaning that their model lacks frequency-specific processing. Sun et al. [11] combined VMD with black-winged kite optimization and backpropagation (BP) neural networks. While their model incorporates similar-day clustering, BP is a feedforward network without recursive memory. Random weight initialization leads to poor input space coverage [12], and the clustering module is independent of internal reservoir dynamics.

Unlike gradient-descent-based neural networks, which require full backpropagation through time, Echo State Networks train only the output weights, while the reservoir weights remain fixed after random initialization. In this way, training costs are reduced and the vanishing gradient is avoided. This efficiency is crucial for photovoltaic forecasting, which needs to be trained frequently to adapt to weather conditions. ESNs excel at capturing temporal dynamics, as their rich reservoir states form a high-dimensional nonlinear expansion of input history.

However, the standard ESN has some shortcomings: it processes all inputs uniformly and applies a single leakage rate for all frequency bands, while the spectral radius of a randomly connected reservoir must be computed numerically for each realization and varies between realizations [13].

Motivated by these limitations, in this study, we propose a High–Mid–Low Triangular Echo State Network (HMLT-ESN).

The main contributions of this paper are summarized as follows:

1. Attention-driven diagonal leakage adapts reservoir timescales depending on activation energy.

2. VMD with TreX/tanh/sigmoid sub-reservoirs handles multi-scale PV volatility.
3. The circle map provides a more uniform coverage of the normal input area and feature discrimination.
4. An equilateral triangular lattice provides an analytically tractable spectral radius and stable reservoir dynamics.
5. QCBBO integrates opposition-based learning and spiral local search, which support robust hyperparameter selection under different operating conditions.

The rest of this paper is organized as follows: Section 2 reviews ESN, VMD, and the QCBBO algorithm; Section 3 describes the proposed HMLT-ESN architecture, QCBBO optimization strategy, experimental setup, dataset description, and evaluation metrics; Section 4 presents our experimental results, including scene clustering, multi-scale decomposition, prediction performance under different weather conditions, ablation studies, and baseline models; and Sections 5 and 6 summarize the results and propose directions for future work.

2. Theoretical Background

2.1. Echo State Network

The Echo State Network (ESN) is a special type of recurrent neural network introduced by Jaeger et al. in 2001 [14]. While recurrent neural networks often have local minima and complex training algorithms, ESN holds weights fixed after initialization, and learning weights only requires training the output layer weights. The ESN consists of three main components: the input layer, the reservoir, and the output layer. The reservoir is a directed graph composed of a large, sparse, and randomly connected network of nodes. The internal connection weights of the reservoir are randomly generated and remain fixed; not all nodes connect to every other node, and the sparsity of connections promotes the emergence of rich dynamic characteristics. The structure of an Echo State Network is illustrated in Figure 1.

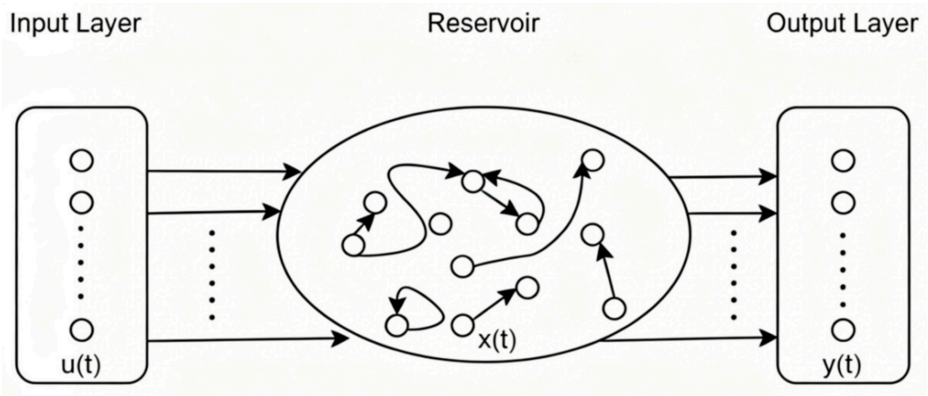


Figure 1. Echo State Network structure.

A typical update equation for a standard Echo State Network is as follows:

$$x(n+1) = f(W_{in}u(n+1) + Wx(n) + W_{back}y(n)) \quad (1)$$

where $f = (f_1, ..., f_N)$ is the internal unit's output functions, and n, $u(n+1)$, $x(n)$,$x(n+1)$, $y(n)$, W_{in}, W, and W_{back} denote discrete time, the input vector, a vector of reservoir neuron activations and its update, the output vector, the connection weight matrix from the input layer to the reservoir, the weight matrix among neurons within the reservoir, and the inverse weight matrix from the output layer to the reservoir, respectively.

The output is computed according to Equation (2):

$$y(n+1) = f_{out}(W_{out}(u(n+1), x(n+1), y(n))) \quad (2)$$

where $f_{out} = (f_{out,1}, f_{out,2}, \ldots, f_{out,L})$ is the output unit's output functions and $u(n+1)$, $x(n+1)$, $y(n)$, and W_{out} are the concatenation of the input, internal, and previous output activation vectors, and the connection weight matrix from the reservoir to the output layer.

Two steps are used to solve W_{out} during the training process of the Echo State Network: one is to use the pseudo-inverse matrix in linear regression to derive the matrix of the output connection weights, and the other is to use the pseudo-inverse method to add the regularization term.

1. Pseudo-inverse method

   The output connection weight matrix is expressed using Equation (3):

$$\begin{aligned} X^{+} &= \left(XX^{T}\right)^{-1}X^{T} \\ W^{out} &= X^{+}Y \end{aligned} \quad (3)$$

where the pseudo-inverse matrix of X is X^{+}, and X is the state matrix.

2. Ridge regression

   The output connection weight matrix is denoted by Equation (4):

$$W^{out} = X^{T}\left(XX^{T} + \psi I_N\right)^{-1}Y \quad (4)$$

where N denotes N dimensions, and ψ is a regularization factor [15].

2.2. Variational Mode Decomposition

VMD decomposes the original data into several Intrinsic Mode Functions (IMFs), each exhibiting unique spectral characteristics. It minimizes the sum of the estimated bandwidths of these modal components by solving a variational problem, as presented in Equation (5):

$$\begin{aligned} \min_{\{u_k\},\{\omega_k\}} \sum_{k=1}^{K} \quad & \left\| \partial_t \left[\left(\delta(t) + \frac{j}{\pi t} \right) u_k(t) \exp(-j\omega_k t) \right] \right\|_2^2 \\ \text{s.t.} \ & \sum_{k=1}^{K} u_k(t) = f(t) \end{aligned} \quad (5)$$

where $f(t)$ is the original sequence, K is the number of decomposed modes, $u_k(t)$ is the $k-$th modal function, ω_k denotes the center frequency of the k−th modal function, t denotes the time, and $\delta(t)$ is the impulse function. In order to solve the problem of the constrained variational, it is transformed into the problem of an unconstrained variational using the Lagrange multiplier method, as illustrated in Equation (6):

$$\begin{aligned} I(u_k(\theta), \omega_k, \lambda) = \alpha \sum_{k=1}^{K} \quad & \left\| \partial_t \left[\left(\delta(t) + \frac{j}{\pi t} \right) u_k(t) \exp(-j\omega_k t) \right] \right\|_2^2 \\ & + \left\| f(t) - \sum_{k=1}^{K} u_k(t) \right\|_2^2 + \left(\lambda(t), f(t) - \sum_{k=1}^{K} u_k(t) \right) \end{aligned} \quad (6)$$

where λ is the Lagrange multiplier and α is the quadratic penalty factor. By solving Equation (6) iteratively in a loop, VMD can decompose the time series data into sub-signals of different frequencies, which helps to accurately capture changes in different frequency components and thus improves the accuracy of the prediction model [16].

2.3. Quasi-Oppositional Learning Cyclonic Foraging Beaver Behavior Optimizer

In 2025, the Beaver Behavior Optimizer (BBO) was introduced as a biologically inspired metaheuristic that mimics the cooperative and exploratory behaviors observed in beaver dam-building and foraging [17].

In the algorithm, each beaver represents a candidate solution, and its position is updated through information exchange and adaptive perturbations.

In the exploration phase, the population is divided into "architects" and "explorers" based on fitness. Architects learn from each other, while explorers learn from architects with additional random disturbances.

In the exploitation phase, individuals simultaneously learn from a randomly selected peer and the global best solution, enhancing local refinement. The dynamic factor E, a sinusoidal function, controls the balance between exploration and exploitation, as illustrated in Equation (7):

$$E = \sin\left(\frac{\pi}{2} \cdot \frac{t}{T_{max}}\right) \tag{7}$$

where E is the dynamic balance factor that adaptively controls the trade-off between exploration and exploitation, t is the current iteration index, and T_{max} is the maximum number of iterations. In early iterations, E is small to promote exploration; later, E grows to shift the emphasis toward exploitation.

Despite this well-structured framework, the convergence trajectory of a standard BBO often exhibits premature stagnation. To overcome this limitation, we augment BBO using the two enhancement modules described below.

2.3.1. Quasi-Oppositional Learning

Conventional opposition-based learning (OBL) generates the opposite point of a given solution to probe the dual space; however, the opposite point may fall far outside the feasible region. Quasi-oppositional learning confines the opposite point within the interval between the center C and the opposite point X_0, yielding more plausible candidates. For the $j-th$ dimension of the $i-th$ individual $x_{i,j}$, we define

$$C_j = \frac{lb_j + ub_j}{2},\ X_{0,j} = lb_j + ub_j - x_{i,j} \tag{8}$$

The quasi-opposite solution is then computed as follows:

$$q_{ij} = \begin{cases} C_j + \left(X_{0,j} - C_j\right) \cdot \text{rand}, & \text{if } x_{i,j} < C_j \\ C_j + \left(C_j - X_{0,j}\right) \cdot \text{rand}, & \text{otherwise} \end{cases} \tag{9}$$

If $f(q_i) < f(x_i)$, then q_i replaces x_i. Quasi-Opposition-Based Learning (QOBL) enhances population diversity without incurring a notable increase in computational overhead, effectively assisting the algorithm in escaping local optima.

Particle Swarm Optimization (PSO) converges slowly toward local optimizations, the Genetic Algorithm (GA) converges slowly due to crossover and mutation, and the standard Whale Optimization Algorithm (WOA) does not have opposition-based initialization and population diversity is poor early in the process. QCBBO quasi-oppositional strategy guarantees quality starting points, while cyclonic foraging mimics spiral contraction around prey to refine local exploitation. Combining these two mechanisms accelerates convergence and global search, which is useful for multimodule coupled architectures with high-dimensional hyperparameter spaces.

2.3.2. Cyclone Foraging Strategy

This strategy is inspired by the spiral foraging behavior of humpback whales and is adaptively modified to suit the BBO framework. At each iteration, a dynamic parameter β is generated for every individual, as illustrated in Equation (10):

$$\beta = 2 \cdot \exp\left(r_g \cdot \frac{T_{max} - t + 1}{T_{max}}\right) \cdot \sin(2\pi r_g) \tag{10}$$

where $r_g \sim U(0,1)$. The value of β adjusts adaptively over the course of iterations: it remains relatively large in early stages to encourage global exploration and gradually shrinks in later

stages to facilitate refined exploitation. Subsequently, the new position of each individual is determined as follows:

For the first individual,

$$x_{new} = x_{best} + rand \cdot (x_{best} - x_i) + \beta \cdot (x_{best} - x_i) \quad (11)$$

For other individuals,

$$x_{new} = x_{best} + rand \cdot (x_{i-1} - x_i) + \beta \cdot (x_{best} - x_i) \quad (12)$$

where x_{new} is the newly generated position of the current individual; x_{best} denotes the global best position found so far; x_i is the current position of the $i-th$ individual; x_{i-1} is the position of the previous individual; rand is a uniformly distributed random number in [0,1]; and β is a positive scaling coefficient that controls the influence of the global best.

This design enables each individual to move toward the best solution while also referencing the previous individual's position, thereby generating a cyclone-like search trajectory that facilitates more thorough exploration in the vicinity of the optimum [18].

3. Proposed Methodology

This section details the proposed HMLT-ESN framework for ultra-short-term PV power forecasting.

3.1. Overall Framework

The proposed forecasting system consists of four main stages:

1. Data Preprocessing and VMD-Based Signal Decomposition: The raw power series and the nine meteorological/NWP variables are cleaned and normalized, and are then decomposed by VMD into high-, mid-, and low-frequency modes, which form a 12-dimensional input.
2. Scene classification with K-means clustering: K-means is applied to the meteorological features to assign each time step a scene label.
3. Construction of sliding-window samples: A window of $L_{in} = 6$ and horizon $L_{out} = 1$ produces input–target pairs, each carrying the scene label of its target step.
4. Training of scene-specific HMLT-ESN models, with hyperparameters optimized by QCBBO: One HMLT-ESN sub-model is trained per scene, while the three shared hyperparameters are optimized once by QCBBO and used across all sub-models.

This is a feedforward workflow: the Stage 1 output features are passed to both Stage 2 and Stage 3; Stage 2 assigns scene labels to the Stage 3 samples; these labels determine which sub-model is used in Stage 4; the Stage 3 samples serve as the training and test data for Stage 4; and no stage feeds information back to an earlier stage. Figure 2 summarizes the overall workflow.

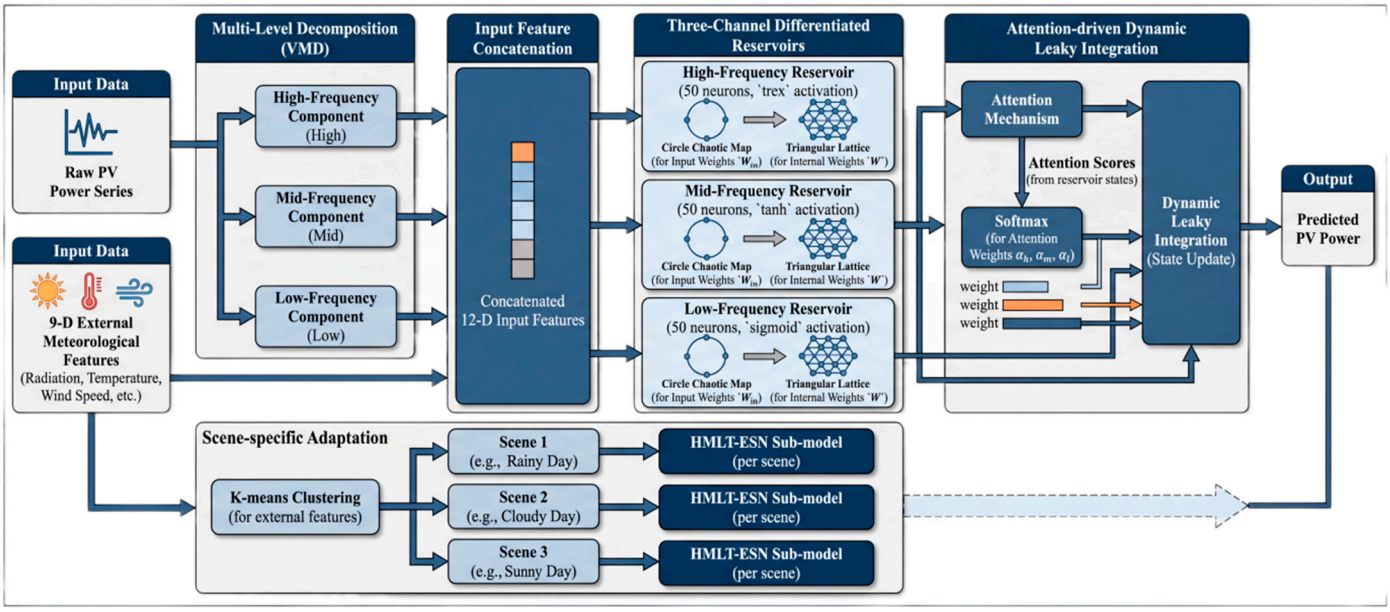


Figure 2. Overall architecture of the proposed HMLT-ESN.

3.2. Experimental Setup and Data Description

The experiments were run in Matlab 2021 on a Windows 11 system. The computer was a 12th Generation Intel Core i5-12400F 2.5 GHz (6 cores, 12 threads) with 16 GB RAM and an NVIDIA GeForce GT 730 graphics card. The system was an Inspiron 3910 64 bit x64 model.

The photovoltaic power dataset used was PVOD v1.0 (Photovoltaic Output Dataset version 1.0) available from Science Data Bank (http://www.scidb.cn) under DOI:10.11922/sciencedb.01094 (CSTR:31253.11.sciencedb.01094). A 20 MWp photovoltaic plant, Station01, is located in northern China (117.46° E, 38.18° N) with a temperate continental monsoon climate. Polycrystalline silicon panels (270 Wp, 74,000 units) are installed at 33° tilt facing south, with 128 minverters rated at 500 kW each. Global horizontal irradiance measurements, diffuse horizontal irradiance, temperature, wind speed, numerical weather prediction variables, and active power are measured. These variables were recorded using TBQ-2 pyranometers with 5% accuracy. Data were collected from 30 June 2018 to 13 June 2019 at 15 min intervals. This 11.5-month period covers a wide range of seasonal and weather conditions, supporting the training and evaluation of robust photovoltaic power forecasting models. Datasets were cleaned into training and test sets, at a ratio of 8:2. Training sets supported model development, while test sets supported final performance evaluation. The sample size was 33,402: 26,721 samples were assigned to training and 6681 samples were assigned to test sets, divided chronologically. Power values below zero were set to zero.

Since raw PV data typically contain missing values and outliers that degrade model performance [19], samples with missing entries in either features or targets were discarded. The remaining data were normalized via z-score:

$$\tilde{x} = (x - \mu_x)/\sigma_x, \tilde{p} = (p - \mu_p)/\sigma_p \quad (13)$$

where x and p denote the normalized feature and power targets, respectively; μ_x and σ_x are the mean and standard deviation of the original feature variable; and μ_p and σ_p are the mean and standard deviation of the original power variable.

The normalized power was decomposed by VMD into three modes (K = 3), with penalty factor $\alpha = 2000$ and convergence tolerance of 10^{-6}. VMD was chosen over Empirical Mode Decomposition (EMD) to avoid mode mixing. Modes were sorted by center frequency into high-, mid-, and low-frequency components and re-normalized [20]. The training and testing

VMD used the training and test power series, respectively, and all predictors were limited to the six preceding steps, so no future observations entered the one-step-ahead prediction.

A sliding window with length $L_{in} = 6$ and horizon $L_{out} = 1$ generated supervised samples. Each input comprised six steps of 12-dimensional features (three VMD modes plus nine meteorological variables). K-means clustering of meteorological features was used to assign scene labels to each sample.

Table 1 lists all nine input variables available at forecast origin t. Four observation-based variables were measured by the on-site sensors up to time t; five NWP-based variables were forecast fields issued before time t. No variable measured after time t was used as input.

Table 1. Input variables and their availability at the forecast origin.

Variable	Description	Unit	Source	Available at Forecast Origin t
lmd_diffuseirrad	Diffuse horizontal irradiance (measured)	W/m^2	Observation	Yes (measured up to t)
lmd_temperature	Air temperature (measured)	°C	Observation	Yes (measured up to t)
lmd_totalirrad	Global horizontal irradiance (measured)	W/m^2	Observation	Yes (measured up to t)
lmd_windspeed	Wind speed (measured)	m/s	Observation	Yes (measured up to t)
nwp_directirrad	Direct normal irradiance (forecast)	W/m^2	NWP forecast	Yes (issued before t)
nwp_globalirrad	Global horizontal irradiance (forecast)	W/m^2	NWP forecast	Yes (issued before t)
nwp_humidity	Relative humidity (forecast)	%	NWP forecast	Yes (issued before t)
nwp_temperature	Air temperature (forecast)	°C	NWP forecast	Yes (issued before t)
nwp_windspeed	Wind speed (forecast)	m/s	NWP forecast	Yes (issued before t)

3.3. Proposed HMLT-ESN Model

The complete model consists of six components:

1. Signal decomposition using VMD.
2. Frequency-specific sub-reservoirs with TreX, tanh, or sigmoid activations.
3. Attention-based dynamic leakage modulation.
4. The circle map-based initialization for input weights.
5. Triangular lattice topology.
6. QCBBO for hyperparameter optimization.

3.3.1. The Circle Map-Based Initialization for Input Weights

Input weights in standard ESNs are randomly initialized from uniform or Gaussian distributions. Steiner et al. argued that purely random initialization is suboptimal for Echo State Networks [21] as it offers no guarantee of coverage of the input domain [22], potentially yielding redundant projections that diminish reservoir discriminability [23].

Logistic sampling is dense near boundaries and sparse in central regions. The tent map is periodic, which is detrimental to effective randomness. The circle map has a uniform distribution across the whole domain but is computationally simple.

The circle map—a chaotic system with proven ergodic properties—generates input weights in lieu of random initialization. This represents the first application of the circle map to reservoir input weights. The circle map is

$$x_{n+1} = \mathrm{mod}\left(x_n + b - \frac{a}{2\pi}\sin(2\pi x_n), 1\right) \tag{14}$$

where x_n denotes the $n-th$ chaotic state variable; n is the iteration index; a and b are control parameters that govern the nonlinear dynamics of the map; and mod confines the output to the unit interval $[0,1)$, ensuring the boundedness of the chaotic sequence.

The parameters are set to $a = 0.5$ and $b = 0.2$. Iterating $N \times M$ times and reshaping yields

$$W_{in} = 2 \cdot \mathrm{reshape}(x, [N, M]) - 1 \tag{15}$$

where N is the number of reservoir neurons; M is the input dimension; and reshape rearranges the sequence into an $N \times M$ matrix. The resulting matrix W_{in} serves as the input weight matrix connecting the M input features to the N reservoir neurons.

Chaotic ergodicity ensures superior coverage of $[-1,1]$ compared with random draws, enhancing input feature discriminability while maintaining reservoir richness.

3.3.2. Weight Triangular Lattice Topology Reservoir

Internal connectivity in a standard ESN is typically a sparse random graph, ignoring the locality prior inherent to meteorological fields [24], where correlations decay with spatiotemporal distance [25]. In this paper, each sub-reservoir is structured as an equilateral triangular lattice, as illustrated in Figure 3, where each neuron connects only to its three nearest neighbors [26]. The coordinates are as follows:

$$(x_i, y_i) = \left(c + 0.5(r \bmod 2), \frac{\sqrt{3}}{2} r\right) \tag{16}$$

Connections are established when the Euclidean distance is less than 1.0, with weights drawn from $N(0,0.5^2)$ then spectrally normalized to spectral radius ρ. A standard ESN uses random sparse connectivity instead of distance-decay connectivity. Unlike random or fully connected topologies, the triangular lattice reduces redundancy and enhances stability. Unlike a grid or circle, the triangular lattice has a greater connectivity density and more local information flow.

The proposed triangular lattice reservoir topology forms bidirectional connections only between geometrically adjacent neurons, offering several advantages over traditional randomly connected and grid-structured Echo State Networks. Firstly, this restricts long-range random couplings. Table 1 shows that replacing the triangular lattice with random topology increases MAE by 3.73% and NMSE by 7.07%, suggesting that the structured connectivity improves reservoir stability. Secondly, the regular, constrained connectivity eliminates unregulated long-range random couplings that often induce ill-conditioned weight matrices. The degree of matrix ill-conditioning can be quantified using the condition number, as illustrated in Equation (17):

$$\kappa(W) = \frac{|\lambda_{max}(W)|}{|\lambda_{min}(W)|} \tag{17}$$

where W is the reservoir internal weight matrix, and $\lambda_{max}$and λ_{min} are the eigenvalues with maximum and minimum moduli, respectively. Larger $\kappa(W)$ indicates more severe ill-conditioning. Compared with fully connected reservoirs and unstructured random graphs, this ordered local connectivity simplifies spectral radius tuning. Reservoir weight matrices require spectral normalization:

$$W_{scale} = \rho \cdot \frac{W}{|\lambda_{max}(W)|} \tag{18}$$

where ρ is the target spectral radius and W_{scale} is the scaled weight matrix. Weight matrices constructed under fixed neighborhood constraints exhibit more stable eigenvalue distributions, reducing the risk of divergent reservoir states during spectral normalization.

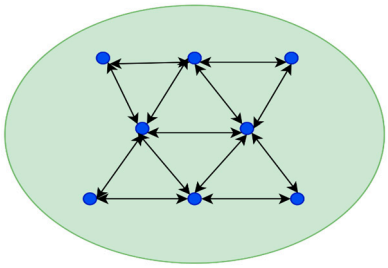


Figure 3. Topology of a triangular lattice reservoir.

3.3.3. Frequency-Specialized Reservoir Partitioning and Frequency-Adapted Activation Function

A standard ESN uses only the tanh activation function, which performs poorly when the frequency content varies widely. Such a model cannot capture multi-scale photovoltaic power dynamics: cloud transients, medium-frequency diurnal cycles, and low-frequency seasonalities each require different nonlinear processing.

In this paper, we partition the reservoir into three parallel sub-reservoirs—high($N_h = 50$), medium($N_m = 50$), and low($N_l = 50$)—each receiving its corresponding VMD mode. Each sub-reservoir adopts a frequency-adapted activation function: TreX for the high-frequency component, tanh for the mid-frequency component, and sigmoid for the low-frequency component.

The TreX function is defined as follows [27]:

$$\begin{aligned} f_{high}(x) &= x \cdot \tan h\left(\frac{x}{1+\exp(-0.5x)} + 0.5\right) \ (x > 0) \\ f_{high}(x) &= x \cdot \tan h\left(\frac{x}{1+\exp(-0.5x)} + 0.5\right) \cdot \exp(0.5x) \ (x \le 0) \end{aligned} \tag{19}$$

TreX enables sharp responses to transients while suppressing oscillations. The tanh function provides balanced saturation for periodic patterns, while the sigmoid function captures smooth trends monotonically.

The overall reservoir state is $x(t) = \left[x_h(t)^\top, x_m(t)^\top, x_l(t)^\top\right]^\top \in R^{N_h+N_m+N_l}$.

3.3.4. Dynamic Diagonal Blockwise Leakage Modulation by Attention

A standard leaky ESN uses a fixed scalar leakage rate $\alpha \in (0,1)$ and same-time smoothing on all neurons regardless of frequency [28]. Unlike multi-scale PV power sequences, where high-frequency components need quick updates while low-frequency trends need slow integration, we extend the scalar leakage to a time-varying diagonal blockwise matrix [29]:

$$\Lambda(t) = \begin{bmatrix} \alpha_h(t) & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & \alpha_h(t) & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \cdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & \alpha_h(t) & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & \alpha_m(t) & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \ddots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & \alpha_m(t) & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & \alpha_l(t) & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \cdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots & \alpha_l(t) \end{bmatrix} \tag{20}$$

or, equivalently, in compact block form:

$$\Lambda(t) = \begin{bmatrix} \alpha_h(t)I_{N_h} & 0 & 0 \\ 0 & \alpha_m(t)I_{N_m} & 0 \\ 0 & 0 & \alpha_l(t)I_{N_l} \end{bmatrix} \tag{21}$$

where I_{N_k} denotes the identity matrix and N_k is the neuron count of the $k-$ th sub-reservoir, with $k \in \{h,m,l\}$.

A separate leakage rate for each neuron would increase computation and the number of trainable parameters, so we assign one rate to all neurons in a block. The leakage coefficients $\alpha_k(t)$ are dynamically generated via an attention mechanism. First, the candidate state of each sub-reservoir is computed as

$$\tilde{x}_k(t) = f_k\left(W_k^{in}u(t) + W_k x_k(t-1)\right) \tag{22}$$

The activation energy is then defined as the mean absolute value of the candidate state:

$$e_k(t) = \frac{1}{N_k}\sum_{i=1}^{N_k} |\tilde{x}_{k,i}(t)| \tag{23}$$

where $\tilde{x}_{k,i}(t)$ is the $i-$ th element of $\tilde{x}_k(t)$. This energy reflects the instantaneous response strength to the current input.

Subsequently, the leakage coefficients are obtained via a softmax attention operator as illustrated in Equation (24):

$$\alpha_k(t) = \frac{\exp(e_k(t))}{\sum_{j\{h,m,l\}}\exp\left(e_j(t)\right)} \tag{24}$$

which ensures $\sum_k \alpha_k(t) = 1$ and implements a competitive allocation mechanism.

The final state update incorporating attention-driven leakage is

$$x_k(t) = (1-\alpha_k(t)) \cdot x_k(t-1) + \alpha_k(t) \cdot \tilde{x}_k(t) \tag{25}$$

This mechanism introduces no additional trainable parameters, as $\alpha_k(t)$ is computed directly from the candidate reservoir states, incurring negligible computational overhead while enabling frequency-adaptive updates [30]. This mechanism resolves conflicting timescale requirements, enabling frequency-adaptive update rates: high-energy sub-reservoirs update rapidly, while low-energy ones integrate slowly, as illustrated in Figure 4.

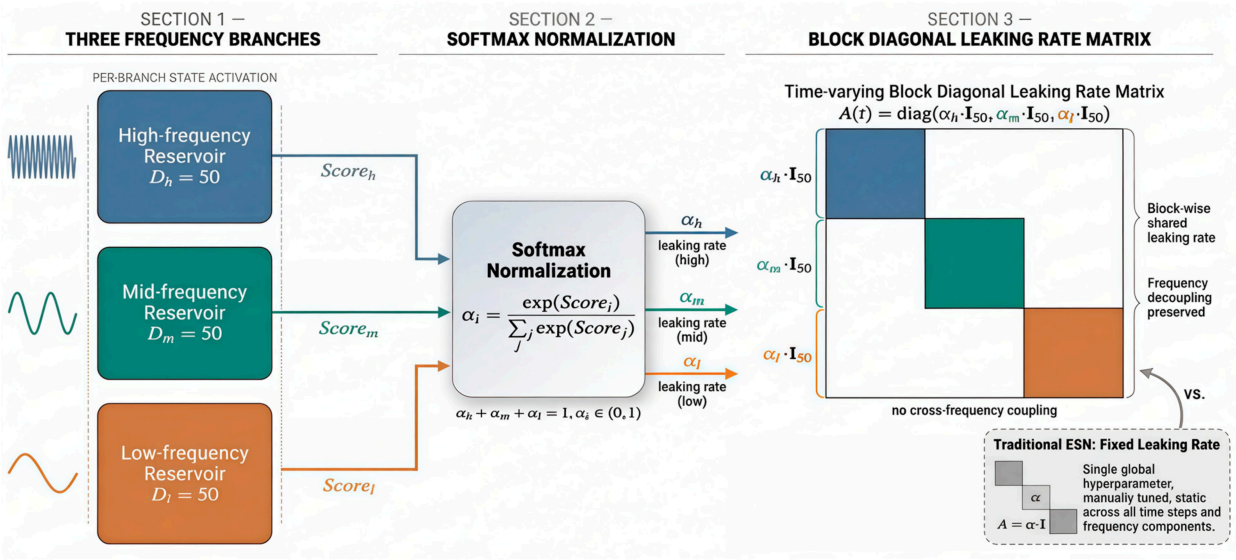


Figure 4. Attention-based leakage modulation architecture.

3.3.5. Procedure of the QCBBO Algorithm

The flowchart of the proposed QCBBO algorithm, shown in Figure 5, was prepared with the assistance of DeepSeek-V4.1-Flash.

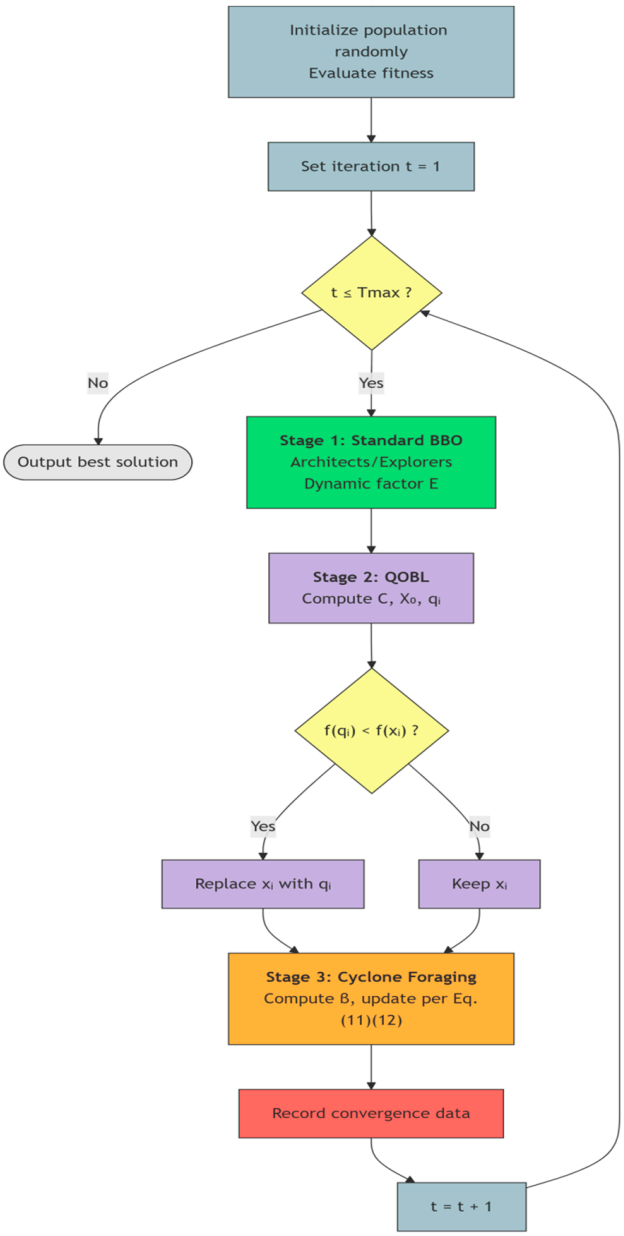


Figure 5. Overall procedure of QCBBO.

Candidate solutions are generated randomly based on the fitness function. The main loop involves three steps: first, the original BBO simulates beaver dam-building to update; second, quasi-oppositional learning produces quasi-opposite solutions to expand the search space and escape local optimization; and third, cyclone foraging searches spirally around the current best solution. After each loop, the convergence curve is recorded and the loop will repeat until the termination condition is satisfied. The optimal solution is the output. This workflow combines bioinspired schemes with two enhancements, maintaining a dynamic balance between global exploration and local exploitation.

3.3.6. Assessment Metrics

This research uses three metrics to assess model performance: Normalized Mean Square Error (NMSE), Mean Absolute Error (MAE), and the coefficient of determination (R^2). Their definitions are as follows:

$$\begin{aligned} NMSE &= \frac{\sum_{j=1}^{n}(y_j - y'_j)^2}{\sum_{j=1}^{n}(y_j - \overline{y})^2}, \\ MAE &= \frac{1}{n}\sum_{j=1}^{n}|y_j - y'_j|, \\ R^2 &= 1 - \frac{\sum_{j=1}^{n}(y_j - y'_j)^2}{\sum_{j=1}^{n}(y_j - \overline{y})^2} \end{aligned} \tag{26}$$

where y_j is the true value, y'_j is the prediction, and $\overline{y}$ is the mean. We removed scale effects using NMSE to make comparisons between datasets fairer. MAE is less sensitive to outliers and more representative of the typical error magnitude. The proportion of variance explained by the model is denoted as R^2: the closer the value is to 1, the better the fit. This should be weighed along with the error metrics to avoid overfitting. Together, these three values provide a balanced view of prediction quality.

4. Results

This section presents our experimental results and analysis of the proposed HMLT-ESN model, including clustering, multi-scale decomposition, prediction performance, ablation, and comparison.

4.1. Clustering Based on Day Similarity

Based on typical weather conditions, we classified all samples into sunny days, cloudy days, and rainy/snowy days. Clustering was matched with sunny, cloudy, and rainy/snowy days using centroid features: higher global irradiance and a lower diffuse fraction means a sunny day, and the opposite is a cloudy/snowy day. The clustering results are shown in Figure 6.

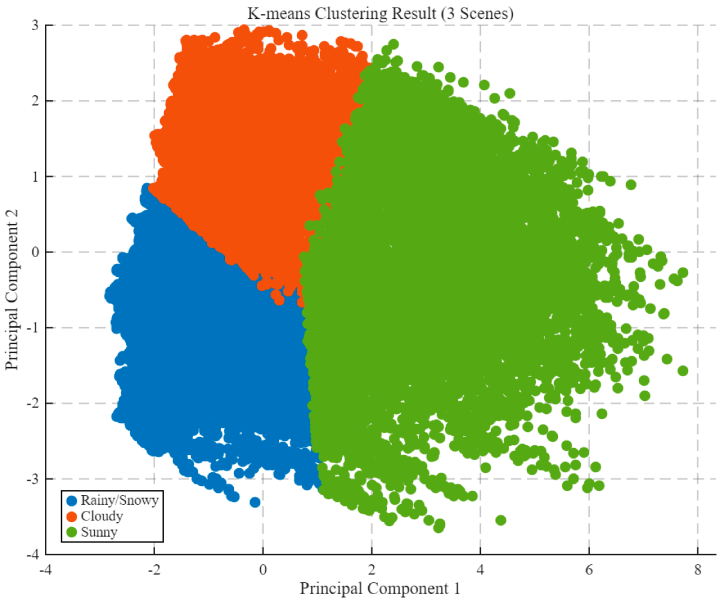


Figure 6. Results of clustering.

4.2. Multi-Scale Decomposition of Historical PV Power

Historical photovoltaic power data vary across multiple frequency ranges due to rapid cloud movement, daily irradiance, and slower airflow. Figures 7 and 8 show the original photovoltaic power series along with the Intrinsic Mode Functions. Figure 8 shows the decomposed signal split into high-frequency, mid-frequency, and low-frequency Intrinsic Mode Functions. The horizontal axis represents the total time steps of the dataset, while the vertical axis represents the amplitude of each decomposed component.

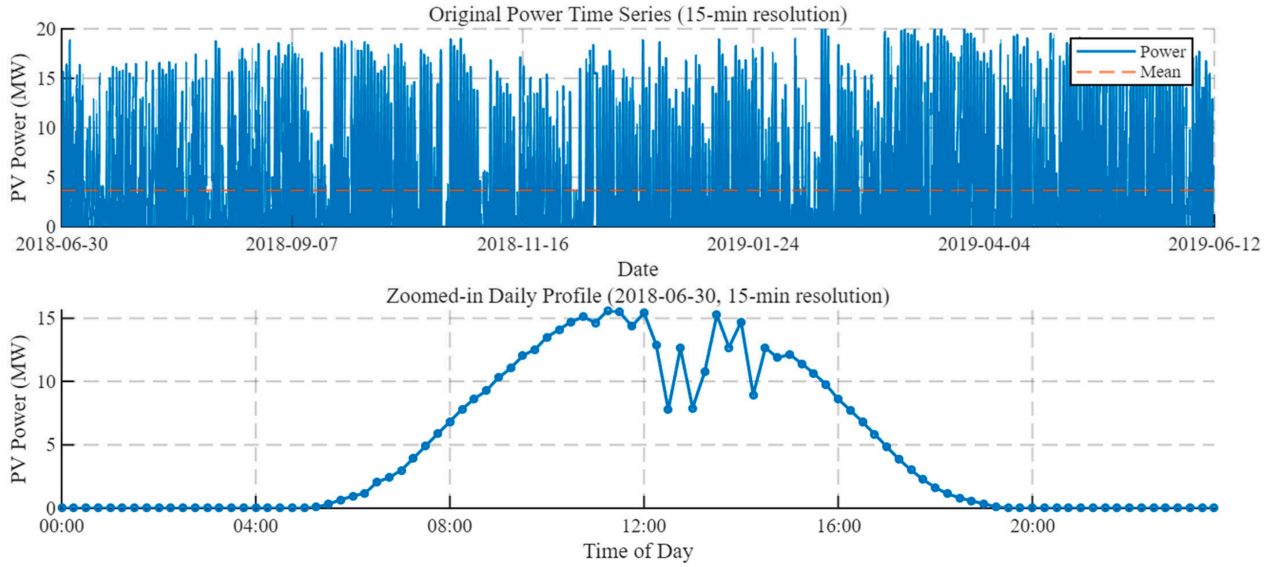


Figure 7. Original PV power series.

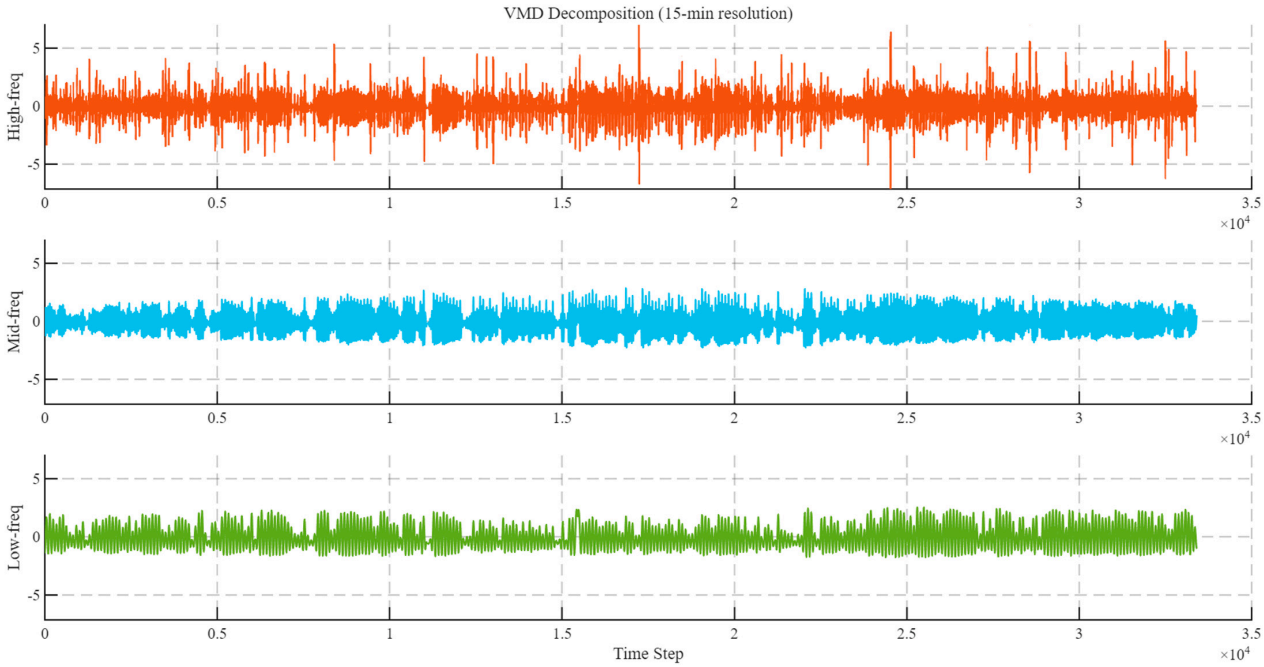


Figure 8. Multi-scale decomposition of historical PV power using Variational Mode Decomposition (K = 3).

4.3. Power Prediction Using HMLT-ESN

Figure 9 shows the prediction of the HMLT-ESN model using photovoltaic test data. The predicted curve, shown as orange dots, closely follows the solid black actual power curve over all samples.

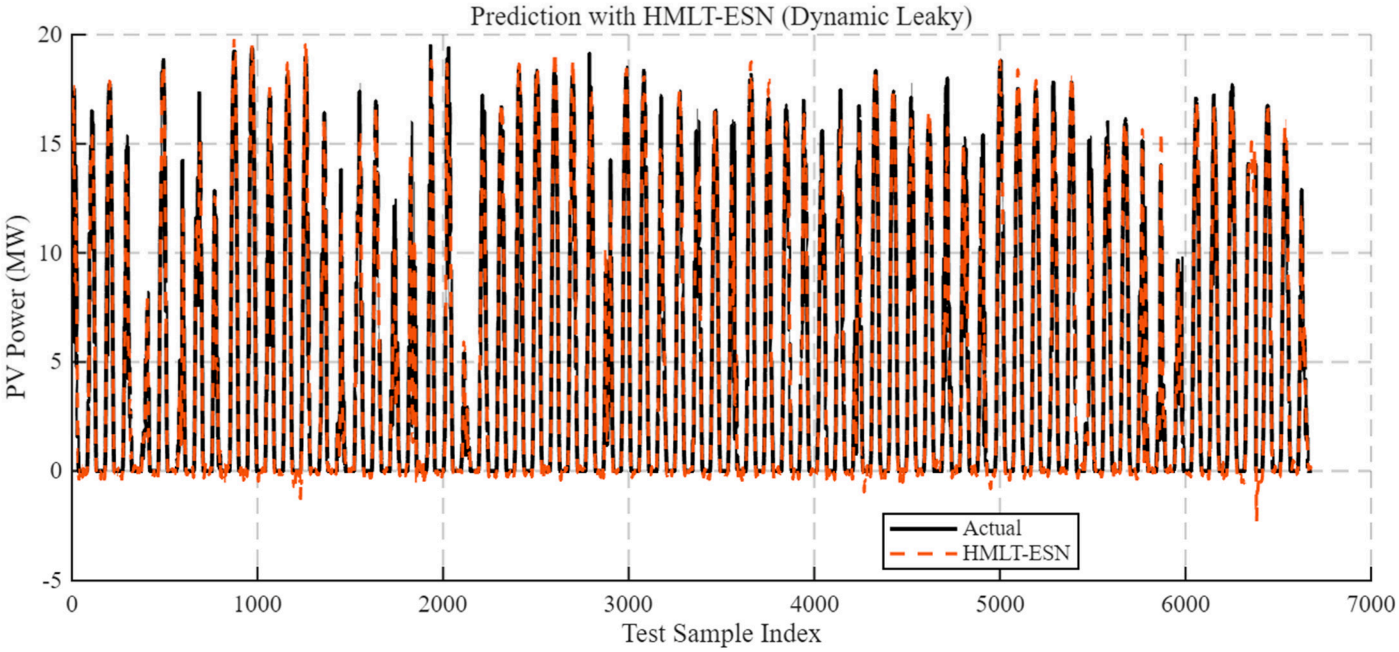


Figure 9. PV power fitting curves over all test samples.

Figure 10 plots the histogram of the residuals. Most samples are distributed around zero residuals with a strong central peak, suggesting a low overall prediction bias.

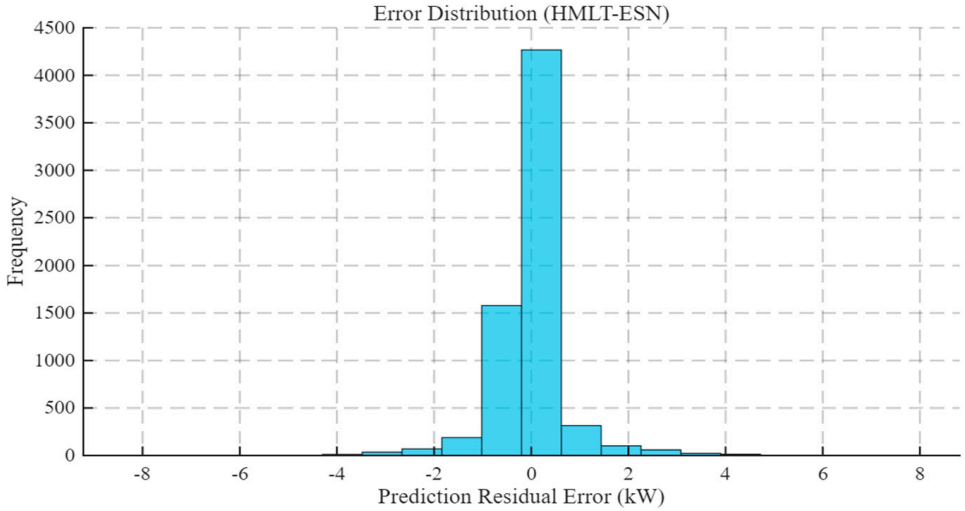


Figure 10. Histogram of global prediction residual errors.

Figure 11 shows the convergence curve of QCBBO hyperparameter tuning for the HMLT-ESN. The validation RMSE decreased sharply after five runs and quickly reached the optimal value, demonstrating the high efficiency of QCBBO's parameter optimization.

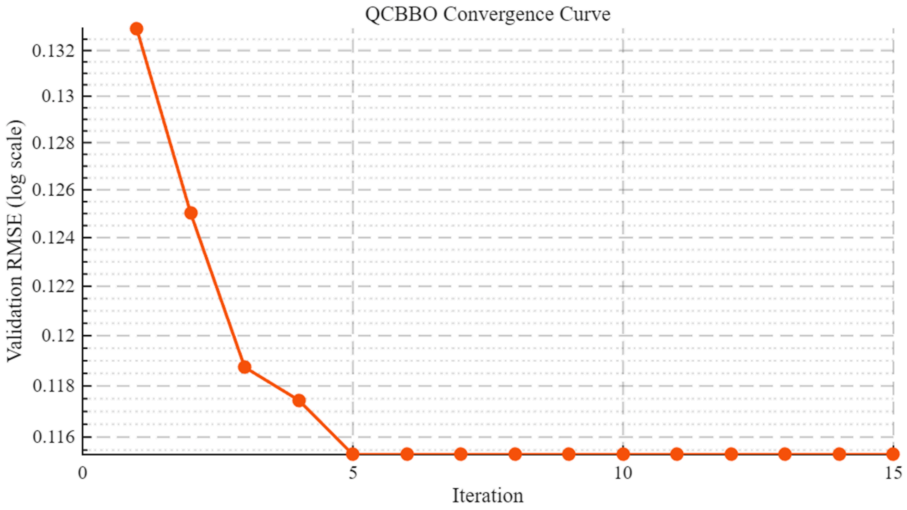


Figure 11. Convergence curve of QCBBO hyperparameter tuning for HMLT-ESN.

Figure 12 shows the actual and predicted power curves for three weather scenarios. In all cases, the predicted power curve matched the actual power curve.

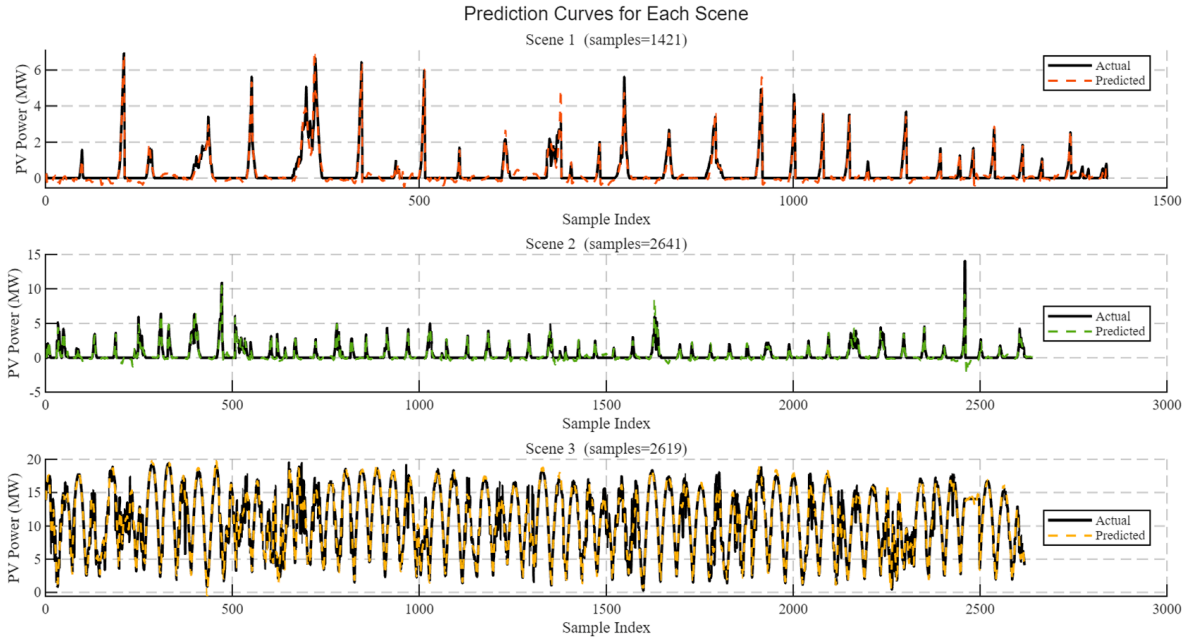


Figure 12. Scene-wise actual and predicted power curves for three weather categories.

Figure 13 shows the residuals of each scene at the 20 MWp plant. The absolute MAE is 0.153 MW, 0.221 MW, and 0.714 MW, respectively, with nMAE values of 0.000765%, 0.01055%, and 0.02377%. Absolute errors increase with more complex weather conditions, but the nMAE is less than 0.044% in all scenes, with residuals centered around zero. Figure 14 shows a numerical comparison of MAE and NMSE in the three scenes.

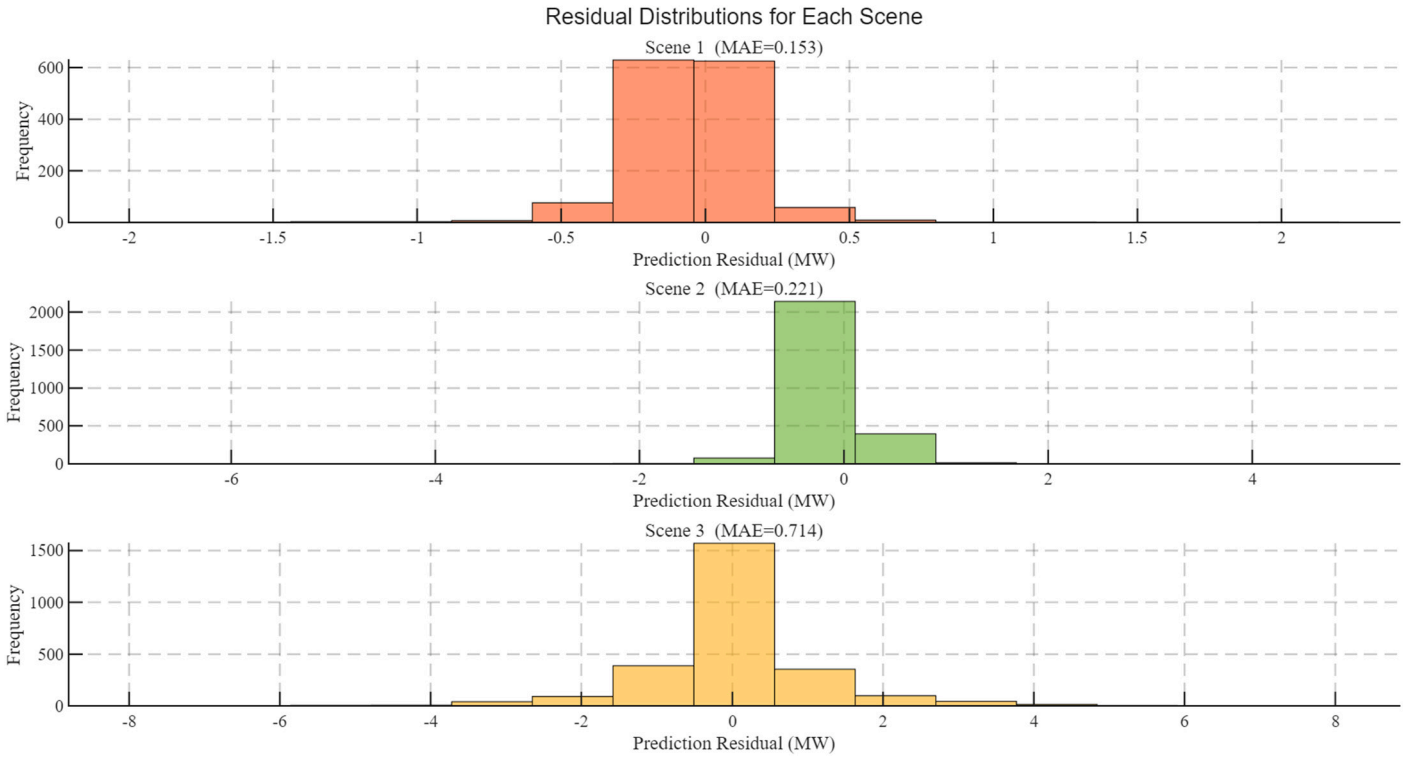


Figure 13. Residual frequency histograms with labeled MAE values for each scene.

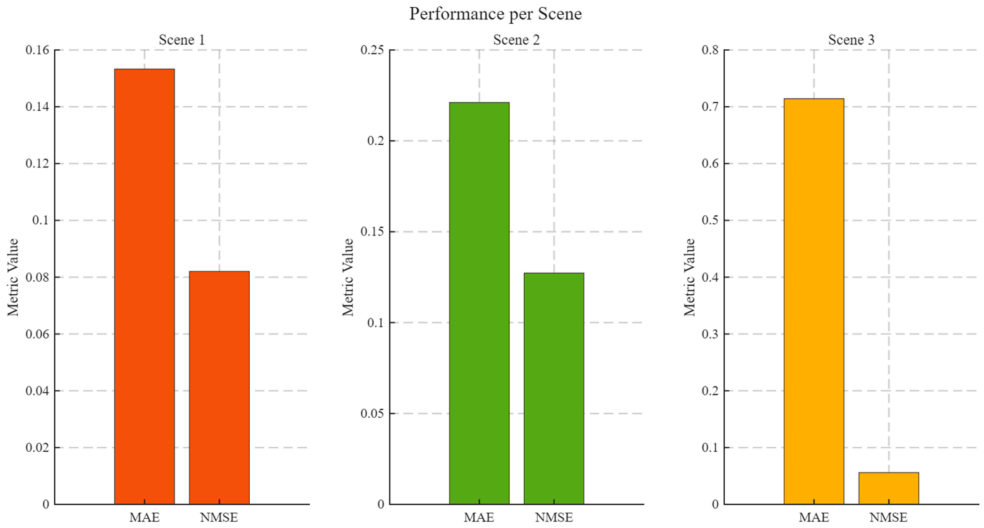


Figure 14. Quantitative comparison of MAE and NMSE metrics for Scene 1, Scene 2, and Scene 3.

4.4. Comparative Analysis

In this study, we performed ablation experiments quantitatively to analyze the contribution of each of the core components in the proposed HMLT-ESN model. Table 2 summarizes the quantitative results, which are also shown in bar graphs and prediction curve comparison diagrams in Figures 15 and 16. The standard ESN was trained with fixed hyperparameters and was not tuned by the QCBBO, while the proposed HMLT-ESN was optimized using the QCBBO.

Table 2. Quantitative comparison of prediction accuracy on the test set.

Modeling	MAE	NMSE	R^2
HMLT-ESN (proposed)	0.400	0.016	0.984
Standard ESN	0.685	0.033	0.967
HMLT-ESN without optimization	0.571	0.025	0.975
HMLT-ESN with random topology	0.415	0.017	0.983
HMLT-ESN with tanh activation	0.423	0.017	0.983
HMLT-ESN without circle mapping	0.441	0.019	0.981
Small-World ESN	0.625	0.028	0.972
Cycle ESN	0.622	0.029	0.971
SVR	1.120	0.091	0.909
LSTM	0.510	0.021	0.980

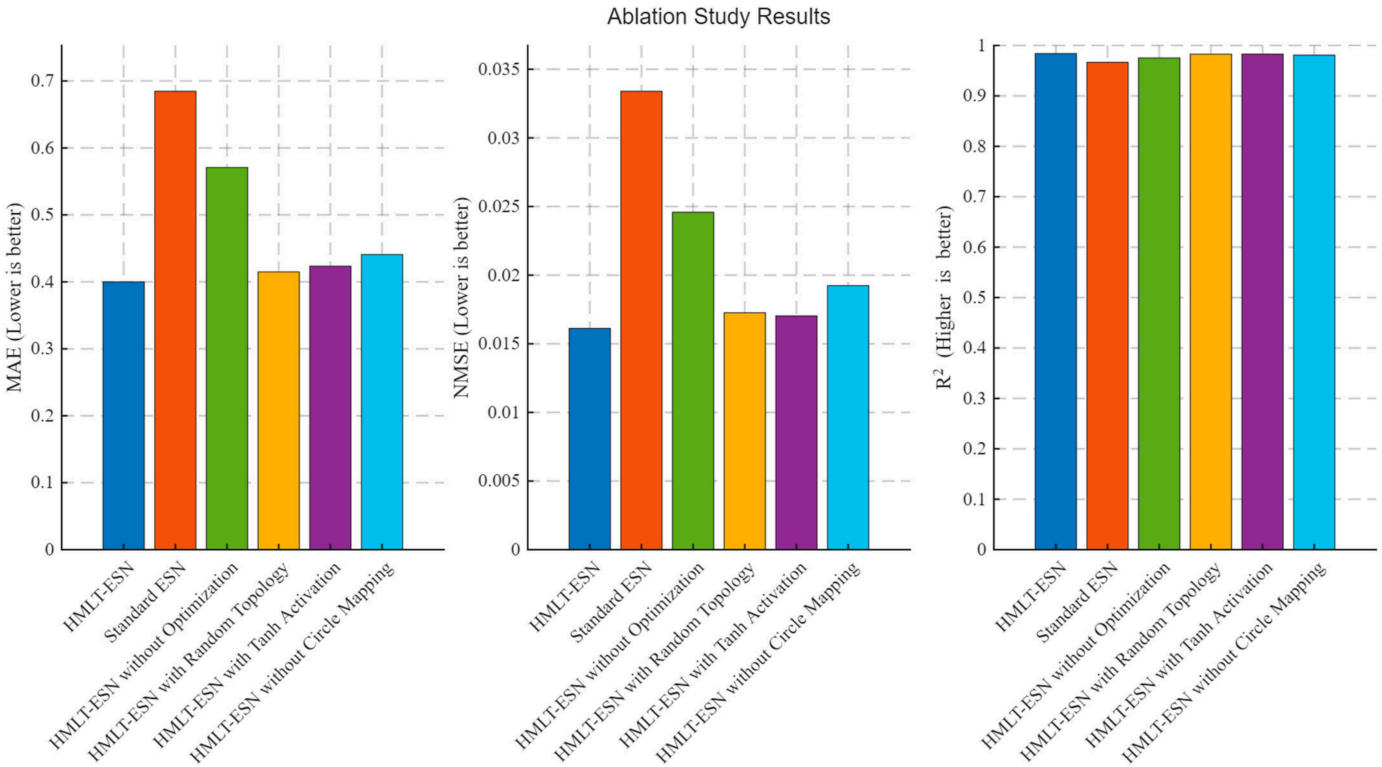


Figure 15. Ablation study results.

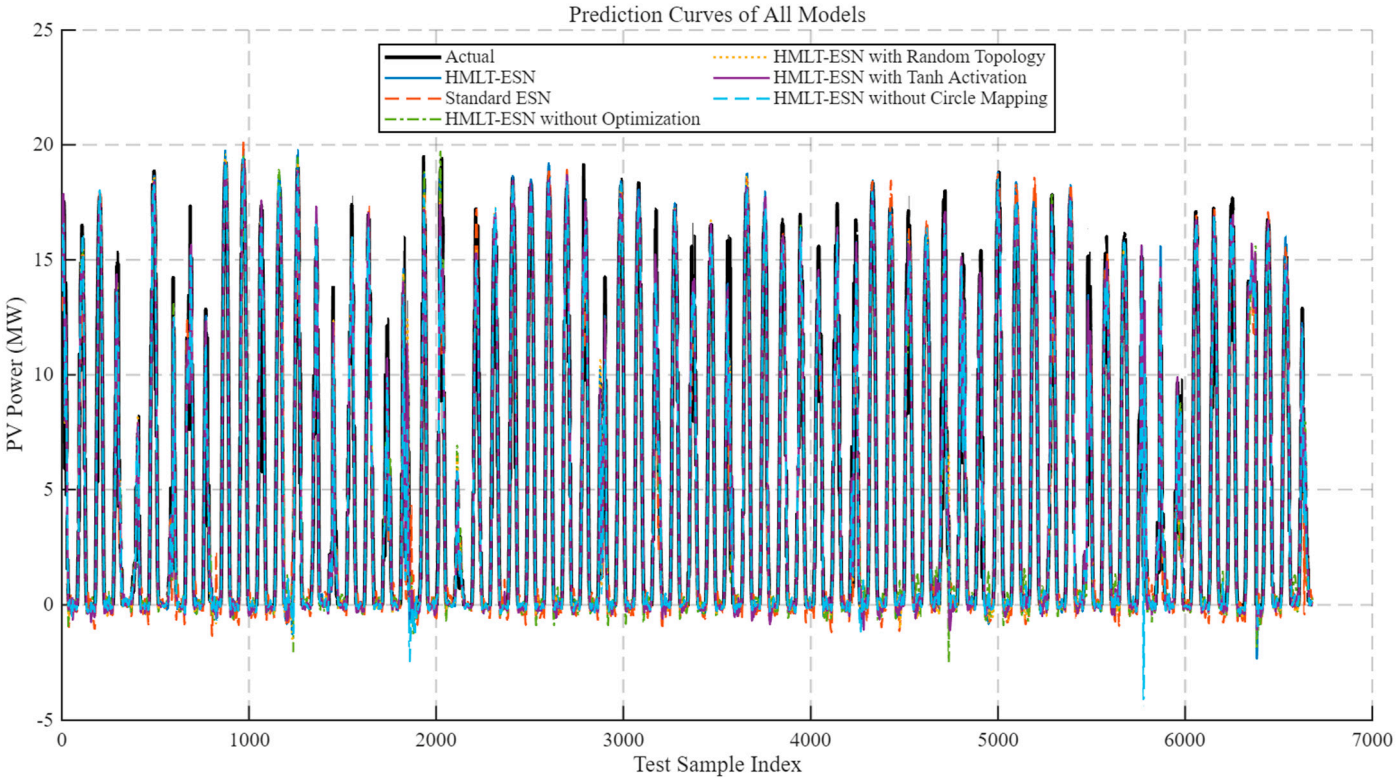


Figure 16. Prediction curves of all models.

Compared with the standard ESN model, HMLT-ESN reduced the Mean Absolute Error by 41.57% and the Normalized Mean Square Error by 51.73%, while R^2 increased by 1.79%. When the QCBBO optimization module was removed, the Mean Absolute Error increased to 0.57077 and the Normalized Mean Square Error rose to 0.024585. These values represent increases of 42.69% and 52.50%, respectively, compared to the full HMLT-ESN, while R^2 decreased by 0.8%. When random reservoir connections were used instead of the triangular lattice topology, MAE and NMSE increased by 3.73% and 7.07%, and R^2 decreased slightly by 0.12%. When the custom activation function was replaced by the standard Tanh activation, MAE and NMSE increased by 5.86% and 5.68%, respectively, while R^2 decreased by 0.09 percentage points. Eliminating the circle map module led to a surge in MAE and NMSE by 10.20% and 19.34%, while R^2 decreased by 0.32 percentage points, indicating that the circle map provides high-quality initial reservoir weights.

Table 2 and Figure 17 provide comparisons of the five models. The proposed HMLT-ESN has the lowest MAE and NMSE and highest R^2. Compared with LSTM, it reduces MAE by 22% and NMSE by 23%, and increases R^2 by 0.48 points. Compared with the Small-World ESN, it lowers MAE by 36% and NMSE by 41%, and raises R^2 by 1.15 points. Compared with Cycle ESN, it decreases MAE by 35.7% and NMSE by 45.0%, and improves R^2 by 1.32 points. Compared with SVR, it cuts MAE by 64.3%, reduces NMSE by 82.4%, and boosts R^2 by 7.53 points.

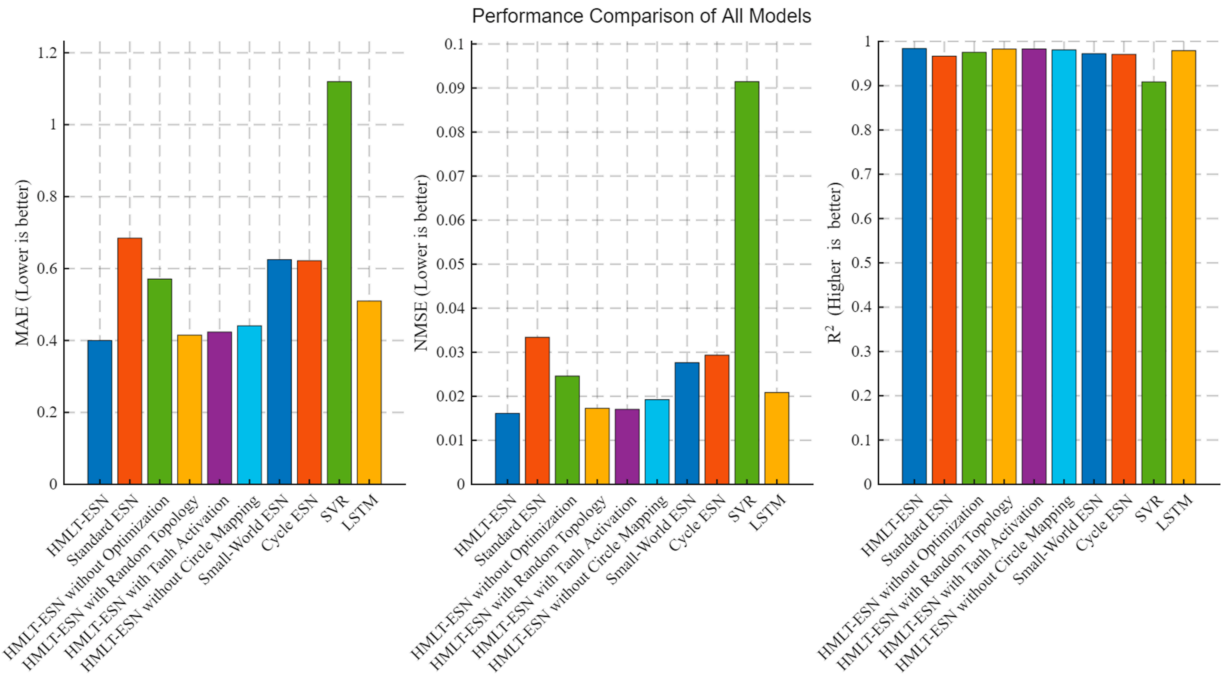


Figure 17. Performance metrics of all compared models.

Figure 18 shows the prediction results for each test sample. The prediction of the HMLT-ESN is close to the real value of the model, while the other models show substantial errors, with values that are much higher or much lower than the real values.

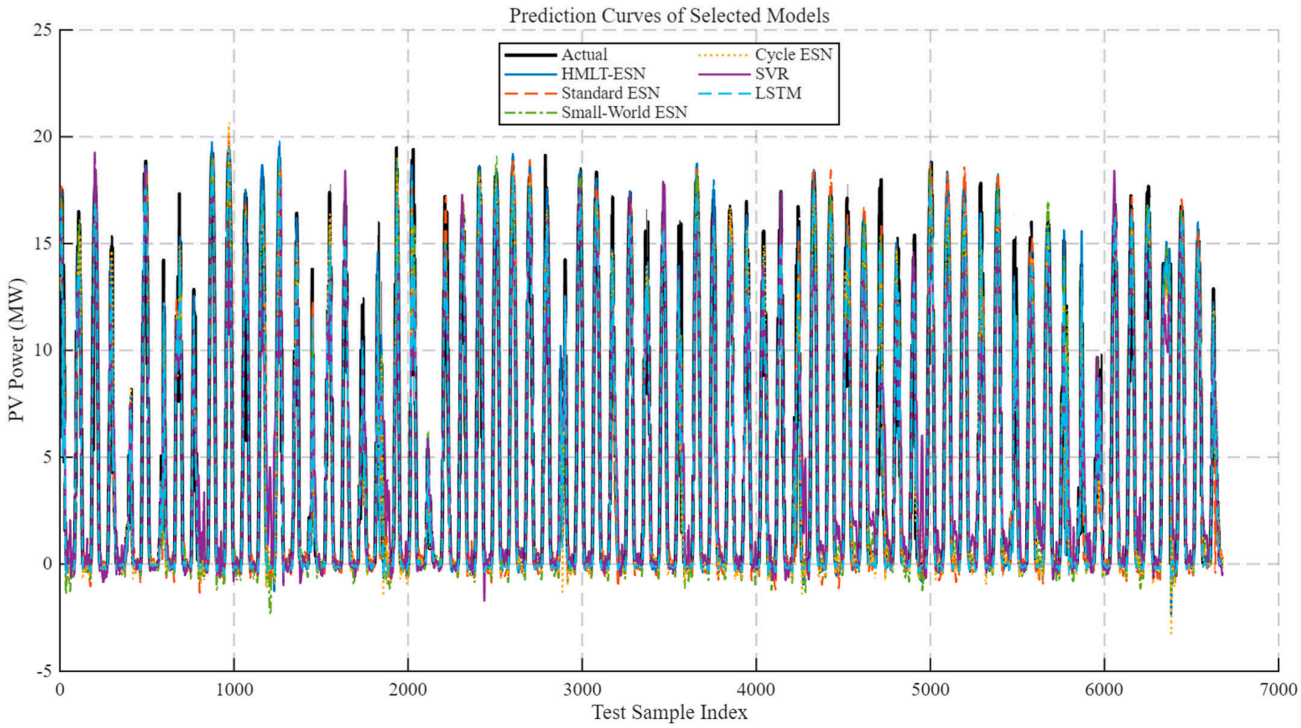


Figure 18. Multi-model prediction comparison on test samples.

Overall, the proposed model outperforms the other four models in terms of both error and R^2, while also tracking the actual power much more closely, indicating that the proposed model is more accurate and reliable.

Figure 19 compares the wall-clock time of the offline training stages and the online inference stages on a log scale, showing that the online prediction time for a single sample was only about 0.1 ms.

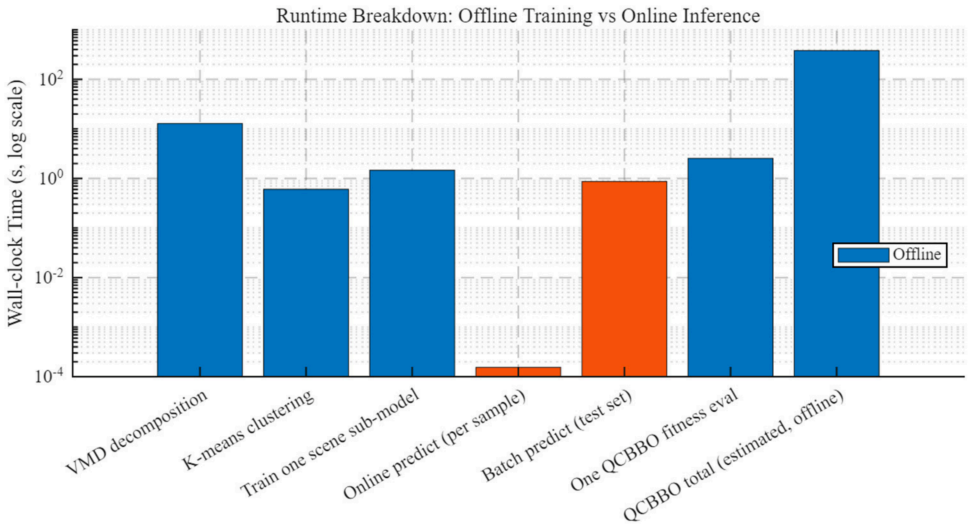


Figure 19. Runtime breakdown.

5. Discussion

The experimental results presented in Section 4 show that the proposed HMLT-ESN framework provides better forecasting performance in all weather scenarios. In this section, we interpret these results and discuss how key innovations contribute to these findings.

The QCBBO module is required for the performance of the HMLT-ESN, as it manages the high-dimensional and complex hyperparameter space of the network. The model includes many interactions, including those of the spectral radius, input scaling, and regularization coefficients.

The triangular lattice reservoir topology represents a key innovation, and our results show the importance of structured connectivity.

The frequency-specific activation function and the circle map provide important model performance enhancements. Our results show that frequency-matched activations are important for multi-scale modeling: VMD splits PV power into high-frequency transients, mid-frequency diurnal cycles, and low-frequency seasonalities. Removing the circle map increases MAE by 10.20% and NMSE by 19.34%. The results also show that input weight initialization plays an important role in reservoir discriminability. Standard random initialization does not guarantee adequate coverage, which may lead to redundant projections. The circle map's existing ergodicity ensures a uniform coverage of the normal input space.

The ablation results show that each module has different and complementary advantages. Figures 17 and 18 compare our proposed HMLT-ESN with many popular benchmark models, including standard ESN, Small-World ESN [31], Cycle ESN, SVR, and LSTM. They show that our model predicts a power curve that is very close to the real measurements with very little deviation, even under sharp fluctuations, providing an effective and very stable prediction. However, the current framework has some shortcomings. First, multi-module architecture and optimization increase computational load, which may restrict the scaling of large datasets or deployment in resource-limited environments. Second, the present framework only supports a fixed 15 min forecasting horizon. Expanding the model to support multiple horizons would help in grid dispatching, where different tasks require forecasts at different timescales.

6. Conclusions

In this study, we present the HMLT-ESN for ultra-short-term photovoltaic power forecasting. Using this model decreased the MAE by 41.57% and the NMSE by 51.73% compared to ESN. Ablation studies showed that the triangular lattice topology improved state conditioning, the circle map improved input discriminability, and frequency-specific activation captured multi-scale dynamics. The attention mechanism compensates for multi-frequency dynamics, which is key to the overall accuracy gain. The model always works well under sunny, cloudy, and rainy/snowy conditions and is adaptable to various weather conditions. The full HMLT-ESN achieves the lowest MAE and NMSE among all competitors, with a highly competitive R^2 value.

The HMLT-ESN framework provides a reliable photovoltaic power forecasting solution, and future work will focus on reducing model complexity and developing multi-horizon variants for different time requirements in real-world power system operation. Overall, we provide a practical technical reference for power grid dispatching and renewable energy integration, supporting stable and efficient next generation power systems.

Author Contributions: Conceptualization, S.L. and M.L.; methodology, Y.Y. and S.L.; software, Y.Y.; validation, Y.Y.; formal analysis, M.L.; investigation, Y.Y.; resources, M.L.; data curation, Y.Y.; writing—original draft preparation, Y.Y.; writing—review and editing, S.L.; visualization, Y.Y.; supervision, M.L.; project administration, S.L.; funding acquisition, S.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the National Natural Science Foundation of China (No. 62473061) and the Basic Scientific Research Project of the Educational Department of Liaoning Province–Scientific Research Platform Construction Project (No. LJ232510167002).

Data Availability Statement: The raw data supporting the findings of this study are openly available in the Science Data Bank repository at https://doi.org/10.11922/sciencedb.01094 (DOI: 10.11922/sciencedb.01094; CSTR: 31253.11.sciencedb.01094).

Acknowledgments: We thank the College of Control Science and Engineering, Bohai University, for providing the test site and the computers used for simulation, and all those who contributed to this work. We are also grateful to the National Natural Science Foundation of China and the Scientific Research Fund of Liaoning Provincial Education Department for their financial support, which made this research possible. During the preparation of this study, the authors used DeepSeek-V4.1-Flash for the purposes of improving the flowchart of Figure 5. The authors have reviewed and edited the output and take full responsibility for the content of this publication.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

HMLT-ESN	High–Mid–Low Triangular Echo State Network
QCBBO	Quasi-Oppositional Learning Cyclonic Foraging Beaver Behavior Optimizer
VMD	Variational Mode Decomposition
MAE	Mean Absolute Error
NMSE	Normalized Mean Square Error
AI-based	Artificial Intelligence-Based
SVR	Support Vector Regression
EEMD	Ensemble Empirical Mode Decomposition
BiGRU	Bidirectional Gated Recurrent Unit
BP	Backpropagation
ESN	Echo State Network
IMFs	Intrinsic Mode Functions
BBO	Beaver Behavior Optimizer
QOBL	Quasi-Opposition-Based Learning

PSO	Particle Swarm Optimization
GA	Genetic Algorithm
WOA	Whale Optimization Algorithm
EMD	Empirical Mode Decomposition
R^2	Coefficient of Determination

References

1. Luo, C.; Al-Messabi, N.; Kuang, Z.; Ma, C.; El-Amin, I.; Deng, H.; Li, Y. Photovoltaic system modeling and forecasting techniques: A survey. *Eng. Appl. Artif. Intell.* **2025**, *162*, 112516. [CrossRef]
2. Wu, Z.; Fang, G.; Ye, J.; Zhu, D.; Huang, X. A reinforcement learning-based ensemble forecasting framework for renewable energy forecasting. *Renew. Energy* **2025**, *244*, 122692. [CrossRef]
3. Chen, Y.; Liu, J.; Liu, B.; Liu, S.; Zhang, D. Research on Photovoltaic Long-Term Power Prediction Model Based on Superposition Generalization Method. *Processes* **2025**, *13*, 1263. [CrossRef]
4. Khouili, O.; Hanine, M.; Louzazni, M.; Flores, M.; Villena, E.; Ashraf, I. Evaluating the impact of deep learning approaches on solar and photovoltaic power forecasting: A systematic review. *Energy Strategy Rev.* **2025**, *59*, 101735. [CrossRef]
5. Asghar, R.; Fulginei, F.; Quercio, M.; Mahrouch, A. Artificial Neural Networks for Photovoltaic Power Forecasting: A Review of Five Promising Models. *IEEE Access* **2024**, *12*, 90461–90485. [CrossRef]
6. Liu, Q.; Li, Y.; Jiang, H.; Chen, Y.; Zhang, J. Short-term photovoltaic power forecasting based on multiple mode decomposition and parallel bidirectional long short term combined with convolutional neural networks. *Energy* **2024**, *286*, 129580. [CrossRef]
7. Zi, X.; Liu, F.; Liu, M.; Wang, Y. Transformer with Adaptive Sparse Self-Attention for Short-Term Photovoltaic Power Generation Forecasting. *Electronics* **2025**, *14*, 3981. [CrossRef]
8. Liu, L.; Liu, F.; Zheng, Y. A Novel Ultra-Short-Term PV Power Forecasting Method Based on DBN-Based Takagi-Sugeno Fuzzy Model. *Energies* **2021**, *14*, 6447. [CrossRef]
9. Wadood, A.; Albalawi, H.; Alatwi, A.; Anwar, H.; Ali, T. Design of a Novel Fractional Whale Optimization-Enhanced Support Vector Regression (FWOA-SVR) Model for Accurate Solar Energy Forecasting. *Fractal Fract.* **2025**, *9*, 35. [CrossRef]
10. Zhang, J.; Qi, H.; Ju, X.; Wang, H.; Ye, G.; Huang, B.; Qi, M.; Tang, Y. Short-Term Photovoltaic Power Forecasting Based on EEMD Residual Secondary IWOA-VMD Decomposition and ISSA-Optimized BiGRU. *Sustainability* **2026**, *18*, 2234. [CrossRef]
11. Sun, Y.; Wang, Z.; Wang, J.; Li, Q. Short-Term Solar Photovoltaic Power Prediction Utilizing the VMD-BKA-BP Neural Network. *Symmetry* **2025**, *17*, 784. [CrossRef]
12. Wang, Q.; Lin, H. Ultra-short-term PV power prediction using optimal ELM and improved variational mode decomposition. *Front. Energy Res.* **2023**, *11*, 1140443. [CrossRef]
13. Zhang, B.; Miller, D.J.; Wang, Y. Nonlinear system modeling with random matrices: Echo state networks revisited. *IEEE Trans. Neural Netw. Learn. Syst.* **2012**, *23*, 175–182. [CrossRef] [PubMed]
14. Jaeger, H. *The 'Echo State' Approach to Analysing and Training Recurrent Neural Networks—With an Erratum Note*; GMD Report 148; German National Research Center for Information Technology: Bonn, Germany, 2001.
15. Lun, S.; Hu, H.; Yao, X. The modified sufficient conditions for echo state property and parameter optimization of leaky integrator echo state network. *Appl. Soft Comput.* **2019**, *77*, 750–760. [CrossRef]
16. Dragomiretskiy, K.; Zosso, D. Variational Mode Decomposition. *IEEE Trans. Signal Process.* **2014**, *62*, 531–544. [CrossRef]
17. Ouyang, K.; Wei, D.; Sha, X.; Yu, J.; Zhao, Y.; Qiu, M.; Fu, S.; Heidari, A.A.; Chen, H. Beaver behavior optimizer: A novel metaheuristic algorithm for solar PV parameter identification and engineering problems. *J. Adv. Res.* **2026**, *84*, 525–555. [CrossRef] [PubMed]
18. Hassan, M.; Kamel, S.; Jurado, F.; Ebeed, M.; Elnaggar, M. Economic load dispatch solution of large-scale power systems using an enhanced beluga whale optimizer. *Alex. Eng. J.* **2023**, *72*, 573–591. [CrossRef]
19. Zhu, C.; Tu, Y.; Wei, Q.; Zang, Y.; Zhou, P.; Yang, L.; Wang, J.; Yu, Y.; Lv, R.; Du, J.; et al. Short-term power prediction of photovoltaic power station based on LSTM-XGBoost model. *Sol. Energy* **2025**, *300*, 113819. [CrossRef]
20. Zhai, C.; He, X.; Cao, Z.; Abdou-Tankari, M.; Wang, Y.; Zhang, M. Photovoltaic power forecasting based on VMD-SSA-Transformer: Multidimensional analysis of dataset length, weather mutation and forecast accuracy. *Energy* **2025**, *324*, 135971. [CrossRef]
21. Steiner, P.; Jalalvand, A.; Birkholz, P. Cluster-Based Input Weight Initialization for Echo State Networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, *34*, 7648–7659. [CrossRef] [PubMed]
22. Qiao, J.; Wang, L.; Yang, C.; Gu, K. Adaptive Levenberg-Marquardt Algorithm Based Echo State Network for Chaotic Time Series Prediction. *IEEE Access* **2018**, *6*, 10720–10732. [CrossRef]
23. Xie, M.; Wang, Q.; Yu, S. Time Series Prediction of ESN Based on Chebyshev Mapping and Strongly Connected Topology. *Neural Process. Lett.* **2024**, *56*, 30. [CrossRef]
24. Wang, B.; Lun, S.; Li, M.; Lu, X. Multi-reservoir echo state network with five-elements cycle. *Inf. Sci.* **2024**, *661*, 120166. [CrossRef]

25. Singh, P.; Madare, O.; Raman, B. Hodge-projected echo-state networks with topologically anchored memory for chaotic flows. *Chaos Solitons Fractals* **2026**, *202*, 117460. [CrossRef]
26. Xue, F.; Li, Q.; Li, X. The combination of circle topology and leaky integrator neurons remarkably improves the performance of echo state network on time series prediction. *PLoS ONE* **2017**, *12*, e0181816. [CrossRef] [PubMed]
27. Khan, A.; Almuhaideb, S. TRex: A Smooth Nonlinear Activation Bridging Tanh and ReLU for Stable Deep Learning. *Electronics* **2025**, *14*, 4661. [CrossRef]
28. Lun, S.; Zhang, Z.; Li, M.; Lu, X. Parameter Optimization in a Leaky Integrator Echo State Network with an Improved Gravitational Search Algorithm. *Mathematics* **2023**, *11*, 1514. [CrossRef]
29. Wang, B.; Lun, S.; Li, M.; Lu, X. Echo state network structure optimization algorithm based on correlation analysis. *Appl. Soft Comput.* **2024**, *152*, 111214. [CrossRef]
30. Mustaqeem, K.; El Saddik, A.; Alotaibi, F.; Pham, N. AAD-Net: Advanced end-to-end signal processing system for human emotion detection & recognition using attention-based deep echo state network. *Knowl.-Based Syst.* **2023**, *270*, 110525. [CrossRef]
31. Kawai, Y.; Park, J.; Asada, M. A small-world topology enhances the echo state property and signal propagation in reservoir computing. *Neural Netw.* **2019**, *112*, 15–23. [CrossRef] [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.