

Article

An Attention-Enhanced Hybrid Deep Learning Framework with Improved Harris Hawks Optimization for Short-Term Wastewater Flow Forecasting

Qingyang Zhang ¹, Jingjing Sun ^{1,2,*} and Shuang Li ²¹ Basic Department, Shanxi Agricultural University, Jinzhong 030801, China; 20230909026@stu.sxau.edu.cn² College of Agriculture, Shanxi Agricultural University, Jinzhong 030801, China; 20235004@stu.sxau.edu.cn

* Correspondence: sunjingjing@sxau.edu.cn

Abstract

Short-term wastewater flow forecasting is essential for pump-station scheduling, influent-load regulation, and overflow-risk warning in wastewater treatment systems. However, wastewater flow series often exhibit strong nonlinearity, large-amplitude fluctuations, and sensitivity to model hyperparameters, which makes accurate real-time prediction challenging. To address these issues, this study proposes an attention-enhanced hybrid deep learning framework optimized by an Improved Harris Hawks Optimisation algorithm. The prediction target is the wastewater flow value at the next target time step, and the model uses only historical wastewater flow observations as input. The framework integrates a Temporal Convolutional Network to extract local fluctuation and multi-scale temporal features, a Bidirectional Long Short-Term Memory network to capture contextual dependencies within the historical input window, and a Simple Attention Module to recalibrate high-dimensional features and emphasize key information. The improved optimisation algorithm further enhances hyperparameter search through hybrid initialization, early-stage differential evolution mutation, stagnation detection with diversity injection, and late-stage Lévy flight perturbation. Experimental results show that the unoptimized Temporal Convolutional Network (TCN)–Bidirectional Long Short-Term Memory (BiLSTM)–Simple Attention Module (SimAM) model achieved an R^2 of 0.9697, an RMSE of 231.1821, and an MAE of 177.5324 on the validation set. After Improved Harris Hawks Optimization (IHHO) optimization, these values improved to 0.9728, 218.8228, and 161.8777, respectively. Compared with the unoptimized model, Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) were reduced by approximately 5.35% and 8.82%, respectively, while R^2 increased by 0.0031. These results indicate that the proposed framework improves prediction accuracy, stability, and robustness under the tested conditions. It provides a feasible technical solution for short-term wastewater flow forecasting and supports intelligent operation management of wastewater treatment systems.

Academic Editors: Joanna Boguniewicz-Zabłocka and Ewelina Łukasiewicz

Received: 17 June 2026

Revised: 22 July 2026

Accepted: 24 July 2026

Published: 29 July 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 \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

Keywords: wastewater flow prediction; deep learning; SimAM attention module; temporal convolutional network; improved Harris hawks optimization; hyperparameter optimization

1. Introduction

Short-term wastewater flow forecasting is important for the stable operation of wastewater treatment systems, pump-station scheduling, sewer-network load allocation, and overflow-risk warning [1]. Accurate forecasting can also support refined process management and real-time operational decision-making [2]. However, wastewater flow series commonly exhibit strong nonlinearity, abrupt peak–valley variations, random disturbances, and multi-scale temporal dependencies. Consequently, existing forecasting methods may encounter difficulties in jointly extracting local fluctuations, modelling temporal dependencies, maintaining robustness, and selecting appropriate hyperparameters [3].

Existing wastewater flow forecasting methods can generally be divided into three categories: traditional statistical methods, machine-learning methods, and deep-learning methods. Statistical approaches, including Autoregressive Integrated Moving Average (ARIMA) [4], autoregressive models [5], and Holt–Winters methods [6], are commonly used as baseline models because of their simple structures, low parameter requirements, and strong interpretability; however, their dependence on predefined linear or stationary relationships limits their ability to represent strong nonlinearities, abrupt fluctuations, and random disturbances. Do et al. [7] applied Seasonal Autoregressive Integrated Moving Average (SARIMA) to influent-flow forecasting at a wastewater treatment plant and showed that it could effectively capture trend and seasonal patterns, although its adaptability remained limited under highly nonlinear conditions. Giberti et al. [8] reported that autoregressive models are easy to implement and interpret, but their prediction accuracy may deteriorate for long time series and complex operating conditions. Ye et al. [9] further demonstrated that ARIMA–Markov and related hybrid models perform well in short-term and small-sample forecasting, whereas their ability to represent complex nonlinear relationships and long-range temporal dependencies decreases as the forecasting horizon increases. To address these limitations, machine-learning methods with stronger nonlinear mapping capabilities have been increasingly applied, including Random Forest [10], Support Vector Regression [11], K-Nearest Neighbours (KNN) [12], Extreme Gradient Boosting (XGBoost) [13], Categorical Boosting (CatBoost) [14], and Light Gradient Boosting Machine (LightGBM) [15]. Zhou et al. [1] found that adaptive random forests, adaptive KNN, and adaptive Multilayer Perceptrons (MLPs) were more responsive to changing influent-flow patterns, while Al Saleem et al. [16] showed that CatBoost and XGBoost achieved competitive prediction performance under different weather conditions. These studies confirm the potential of machine-learning methods in nonlinear and multivariable forecasting tasks; nevertheless, their performance often remains dependent on manual feature engineering, data quality, and consistency between training and application scenarios, which may limit model interpretability, transferability, and generalisation.

Compared with traditional statistical and machine-learning approaches, deep-learning models can automatically learn temporal representations from raw sequences and reduce dependence on manual feature engineering. Representative models include Long Short-Term Memory (LSTM), BiLSTM, Convolutional Neural Network (CNN)-LSTM, and Temporal Convolutional Network (TCN)-LSTM. Huang et al. [17] developed an LSTM-based wastewater flow forecasting model and showed that deep learning can outperform conventional statistical methods under complex and dynamic conditions, although its performance is sensitive to training parameters. Yin et al. [18] similarly reported that deep-learning models achieve higher forecasting accuracy than autoregressive models over longer horizons, but their structural complexity and limited interpretability may hinder practical application. Ye et al. [9] further investigated hybrid frameworks such as ARIMA-LSTM-Transformer and confirmed their strong nonlinear representation capability, while also identifying high training costs and complex parameter tuning as major lim-

itations. Moreover, individual deep-learning structures usually emphasise only one aspect of temporal information: recurrent networks are effective in modelling sequential dependencies but may smooth sharp local fluctuations, whereas convolution-based networks are more suitable for extracting local and multi-scale patterns but may inadequately capture contextual dependencies. Therefore, further improvements are required in feature fusion, model complexity, and hyperparameter optimisation to enhance the accuracy and engineering applicability of wastewater flow forecasting models.

In recent years, integrated modelling has been increasingly adopted to overcome the limitations of individual forecasting models. Xie et al. [19] proposed a TCN-LSTM framework that combines local feature extraction with temporal dependency modelling and achieved better performance than individual TCN, LSTM, and Feedforward Neural Network (FFNN) models. Chen et al. [20] integrated signal decomposition, dynamic feature selection, Principal Component Analysis (PCA), and LSTM-CNN to reduce input redundancy and enhance feature representation, although the resulting workflow increased dependence on preprocessing quality. Cao et al. [21] developed a Graph Convolutional Network (GCN)-CNN-GRU-attention model (CNNGA) framework that combined graph-based optimisation, convolutional feature extraction, temporal modelling, and attention mechanisms to improve prediction performance under complex nonlinear conditions. Despite these advances, existing hybrid frameworks still exhibit several common limitations. Multi-module coupling increases structural complexity, computational burden, and sensitivity to data quality and parameter settings, while simple stacking of heterogeneous modules does not necessarily guarantee effective feature fusion. In addition, the growing number of coupled parameters may reduce optimisation efficiency and increase the risk of overfitting and unstable generalisation. Therefore, systematic hyperparameter optimisation is essential for improving the reliability and practical applicability of hybrid forecasting models.

Conventional hyperparameter optimisation methods mainly include grid search [22], random search [23], and Bayesian optimisation [24]. Shams et al. [25] showed that grid search can identify relatively stable parameter combinations within a predefined search space, but its computational cost increases rapidly with the number of candidate parameters. Ye et al. [26] applied Bayesian optimisation to wastewater-treatment prediction and reported that it can reduce ineffective evaluations, although its performance remains sensitive to prior assumptions and surrogate-model updates. Ramya et al. [27] found that random search can obtain competitive solutions with fewer evaluations in some wastewater-prediction tasks, but its search stability is not always sufficient. Overall, these methods can improve model performance, yet they remain constrained by high computational cost, limited search stability, or dependence on search-space and prior settings. These limitations become more evident in multi-module networks involving discrete and continuous coupled hyperparameters, motivating the use of intelligent optimisation algorithms with stronger global search and adaptive capabilities.

The integration of intelligent optimization algorithms with multi-module forecasting models has recently emerged as a promising approach for improving complex time-series prediction. Feng et al. [28] proposed a TCN-BiGRU/BiLSTM hybrid model based on data decomposition and an improved dung beetle optimization algorithm for predicting effluent water quality. Their results showed that, compared with multiple benchmark models, the proposed method reduced average RMSE and MAE by 35.22–52.41% and 39.38–55.53%, respectively, while increasing average R^2 by 2.91–7.55%, thereby demonstrating strong accuracy and robustness. However, the multi-stage coupling of decomposition, dual-branch networks, and optimization also increased the complexity of hyperparameters and the maintenance cost. Peng et al. [29] developed a CNN-Improved Sparrow

Search Algorithm (ISSA)-Bidirectional Gated Recurrent Unit (BiGRU) framework for parameter prediction and operational optimization in water treatment processes, demonstrating that the synergy of convolutional feature extraction, bidirectional temporal modelling, and intelligent optimization can improve prediction accuracy and reduce operating costs by approximately 17.1%. Nevertheless, such methods still rely heavily on frontend feature organization, optimizer settings, and network coupling strategies, which limit their practical deployment. Existing studies have also demonstrated the potential of Harris Hawks Optimisation (HHO) for temporal model parameter optimization. For example, Houssein et al. [30] developed an HHO-SVR model for air-quality forecasting and showed that HHO-based hyperparameter optimisation could achieve lower MAE and RMSE than Grey Wolf Optimizer (GWO) and Sparrow Search Algorithm (SSA) on multiple datasets. More recently, Wang et al. [31] proposed Multi-Strategy Improved (MSI)-HHO by combining an inverted S-shaped escape-energy strategy, Gaussian mutation, and refracted opposition-based learning to enhance population diversity and balance global exploration and local exploitation. Although these studies demonstrate the potential of HHO and its variants, existing improvements mainly focus on modifying individual search operators or adopting fixed combinations of strategies, with limited consideration of the different failure modes arising during coupled neural-network hyperparameter optimisation. In particular, HHO may remain sensitive to the initial population distribution, lose population diversity during iteration, and become trapped in local optima in high-dimensional and strongly coupled search spaces. To address these limitations, this study develops an IHHO framework that coordinates hybrid population initialization, DE mutation, stagnation-triggered diversity recovery with elite-solution preservation, and Lévy-flight perturbation. This design is intended to improve the coordination of initialization, global exploration, stagnation recovery, and local exploitation for hyperparameter optimisation in multi-module wastewater flow forecasting models.

In summary, to overcome the limitations of conventional wastewater flow forecasting models in multi-scale feature extraction, long-term dependency modelling, and non-linear dynamic characterization, this study proposes an IHHO-optimised TCN-BiLSTM-SimAM model for short-term wastewater flow forecasting. In terms of model architecture, a TCN first introduces causal dilated convolutions to enlarge the receptive field, thereby enhancing the ability to capture local fluctuations and multi-scale temporal patterns while preserving causality [32]. Subsequently, a BiLSTM is employed to model temporal dependencies in both forward and backward directions, thereby alleviating information loss and phase lag caused by unidirectional recurrent structures [33]. Furthermore, the SimAM is incorporated to adaptively recalibrate the high-dimensional features generated by the BiLSTM, thereby highlighting critical information at key moments while suppressing redundant noise [34]. To address the increased complexity of hyperparameter tuning in the integrated architecture, an IHHO algorithm is introduced. Specifically, Latin Hypercube Sampling (LHS) improves the diversity of the initial population [35], Differential Evolution (DE) mutation [36] and perturbation mechanisms strengthen global exploration [37], and Lévy flight perturbation enhances local fine search in the later stage [38]. Experimental results demonstrate that the proposed framework achieves higher prediction accuracy, greater robustness, and better convergence, providing reliable support for wastewater treatment plant operation scheduling and water volume management. The main contributions of this study are summarised as follows:

- (1) To address the difficulty of simultaneously capturing rapid local fluctuations and contextual temporal dependencies in short-term wastewater flow series, a TCN-BiLSTM-SimAM forecasting model is developed. TCN extracts local and multi-scale temporal patterns from the historical input window, while BiLSTM models contex-

tual dependencies within the same window. SimAM is further introduced to recalibrate intermediate features without adding extra trainable layers. This complementary design is intended to improve the representation of peak–valley transitions, rapid flow changes, and local disturbances that are particularly important for wastewater-treatment scheduling and equipment-load management.

- (2) To address the inefficient and unstable tuning of coupled hyperparameters in multi-module forecasting networks, an IHHO-based optimisation strategy is proposed. LHS and opposition-based learning are used to improve initial population coverage, DE mutation strengthens global exploration, stagnation-triggered population reconstruction restores diversity while preserving elite solutions, and Lévy-flight perturbation enhances local refinement. IHHO jointly optimises the main parameters of TCN, BiLSTM, SimAM, and the learning rate, reducing dependence on manual tuning. Its optimisation performance is evaluated using CEC2017 tests and wastewater-flow forecasting experiments.
- (3) To examine the applicability of the proposed framework under actual wastewater-treatment operating conditions, experiments are conducted using 5-min flow measurements collected from a wastewater treatment plant in Quanzhou, China. The proposed model is compared with individual, hybrid, attention-based, and optimisation-based benchmark models, and additional ablation and Gaussian-noise experiments are conducted to assess prediction accuracy and robustness. The results provide preliminary evidence that the proposed framework can support short-term flow monitoring and may offer useful decision information for operational scheduling and equipment-load regulation. Because the dataset was obtained from one plant over a limited period, broader engineering generalisation still requires validation using longer-term and multi-site datasets.

2. Hybrid Forecasting Model Based on TCN-BiLSTM-SimAM

This study proposes an integrated TCN-BiLSTM-SimAM forecasting model for univariate wastewater flow prediction. The prediction target is the wastewater flow value at the next target time step, and the model uses only historical wastewater flow observations as input. No additional water-quality, hydraulic, or operational variables are included in the current dataset. The overall architecture is shown in Figure 1. The input is a historical flow sequence constructed by a sliding window. First, the TCN module maps the original sequence into a high-dimensional feature space through causal dilated convolutions. This process helps extract local abrupt changes and multi-scale temporal patterns while avoiding information leakage from future time steps. Then, the extracted features are fed into the BiLSTM module to model contextual dependencies within the historical input window. Finally, the SimAM attention module recalibrates temporal-channel features by enhancing informative components. The calibrated features are passed through a fully connected layer to generate the final prediction. Overall, the model integrates multi-scale feature extraction, temporal dependency modelling, and key feature enhancement for short-term wastewater flow forecasting.

2.1. Temporal Convolutional Network

Within the integrated forecasting framework developed in this study, the TCN serves as the main feature extraction module. Wastewater flow series usually contain nonstationary fluctuations and multi-scale temporal patterns. These characteristics make feature extraction difficult for conventional sequential models. Compared with RNN-based structures, TCN can process sequences more efficiently through convolutional operations. By combining causal convolution and dilated convolution, it extracts both local and long-range temporal features from the historical flow sequence. These features provide stable

inputs for the subsequent BiLSTM module. The detailed structure of the TCN module is shown in Figure 2.

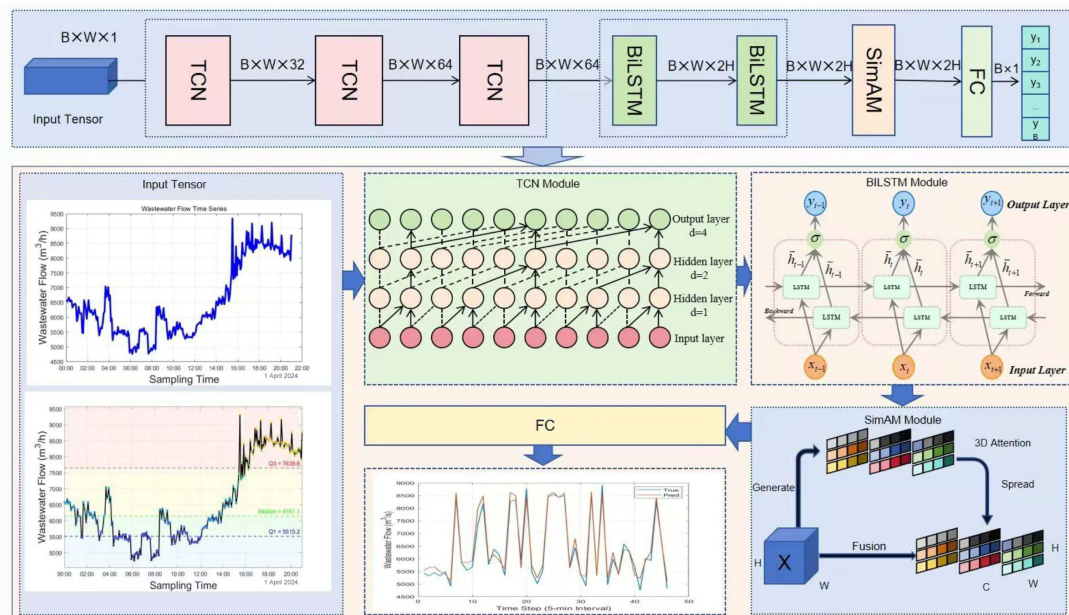


Figure 1. Framework of the TCN-BiLSTM-SimAM Model.

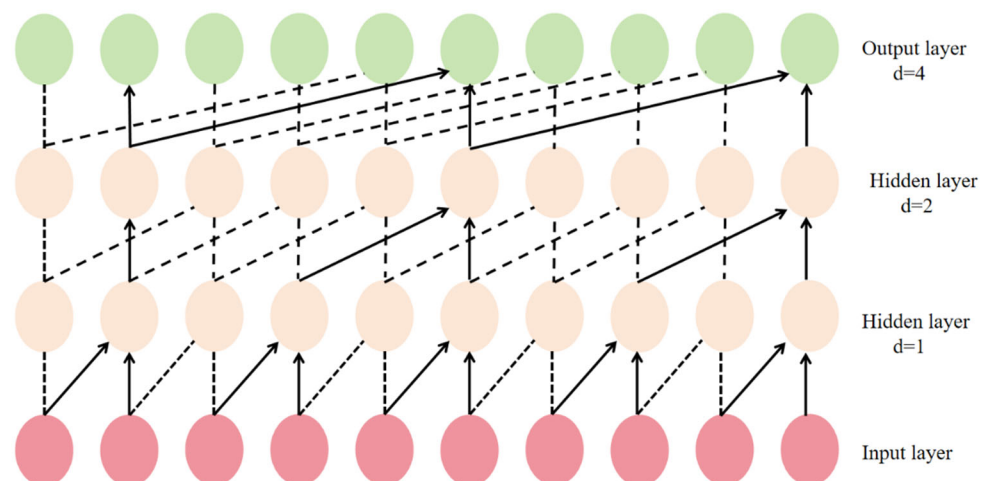


Figure 2. Structure of the Temporal Convolutional Network (TCN) Module.

The core architecture of the TCN is designed to capture multi-scale historical information by expanding the receptive field while strictly adhering to the causality principle in time-series forecasting. To prevent information leakage from future time steps, the model employs Causal Convolution. Through asymmetric padding of the input sequence, it ensures that the convolutional output at time step t depends only on the input at time step t and the corresponding historical inputs prior to it. To enlarge the effective historical window without increasing the number of parameters or computational complexity, the model further introduces the Dilated Convolution mechanism. By inserting intervals between adjacent sampling points in the convolution kernel, dilated convolution enlarges the receptive field without substantially increasing the number of parameters. Considering the 5-min sampling interval and the 24-step input window, the dilation factors were set to 1, 2, and 4 to capture short-term multi-scale variations within the 120-min historical

sequence, rather than a complete 24-h cycle. In addition, to address information degradation in deep networks and accelerate convergence, this study adopts a Residual Connection structure. Through identity mapping, the input is directly added to the activated output, facilitating cross-layer propagation of the original information and improving training stability. Through the combined effect of these mechanisms, the TCN can efficiently extract both local and long-term temporal features from time-series data, thereby providing robust high-dimensional feature representations for the subsequent BiLSTM and attention modules.

Let the normalized wastewater flow matrix processed by a sliding window be denoted as $X \in \mathbb{R}^{B \times W \times 1}$, where B is the batch size, W is the time-window length, and 1 indicates a univariate feature. For any time step t , the dilated convolution operation in the l -th layer is defined as:

$$x_t^{(l)} = \sum_{i=0}^{k-1} f_i x_{t-d \cdot i}^{(l-1)}, t - d \cdot i \geq 1 \quad (1)$$

Where $x^{(l-1)}$ denotes the output of the $(l-1)$ -th layer d is the dilation factor of the current layer, $f \in \mathbb{R}^k$ is the convolution kernel filter, and k is the kernel size. In the code implementation, each residual block contains two identical convolutional units. To simplify the mathematical formulation and improve the readability of the equations, a composite non-linear transformation function $M(\cdot)$ is defined to represent the normalization and activation processes after a single convolution:

$$M(z) = \text{Dropout}(\sigma(\text{BN}(z))) \quad (2)$$

where $\text{BN}(\cdot)$ denotes Batch Normalization, and $\sigma(\cdot)$ represents the ReLU activation function. Based on this, the nonlinear transformation mapping $F(x)$ within the residual block is constructed by two cascaded causal dilated convolution layers, denoted as Conv_1 and Conv_2 :

$$F(x) = M(\text{Conv}_2(M(\text{Conv}_1(x)))) \quad (3)$$

Finally, the output y of the TCN residual block is computed through the residual connection as:

$$y = \sigma(F(x) + T(x)) \quad (4)$$

where $T(x)$ denotes the identity mapping. When the number of input and output channels is inconsistent, a 1×1 convolution is used to project x so that the dimensions match linearly. After three stacked layers, the high-dimensional feature tensor output by the TCN module is denoted as $H^{TCN} \in \mathbb{R}^{B \times W \times 64}$. This tensor fully preserves the temporal dimension W while mapping the original data into a 64-dimensional feature space rich in local contextual information.

2.2. Bidirectional Long Short-Term Memory

Although the TCN exhibits strong local feature extraction and a certain degree of temporal modelling, its convolutional structure still has limitations in capturing long-range temporal dependencies for long sequences characterized by complex dynamic behaviours. For time-series data such as wastewater flow, which exhibit pronounced non-linear fluctuations, periodic variations, and long-term dependencies, Recurrent Neural Networks (RNNs) retain important advantages for temporal memory modelling. On this basis, this study adopts the Long Short-Term Memory (LSTM) network as the core component for temporal modelling and further extends it to a bidirectional structure, namely

the BiLSTM, to capture contextual dependency information within the sequence more comprehensively.

LSTM is an improved variant of the traditional RNN architecture. It uses an input gate, a forget gate, and an output gate to alleviate the vanishing and exploding gradient problems in long-sequence modelling. Therefore, LSTM is effective in capturing long-term dependencies. Compared with conventional RNNs, LSTM can selectively retain important information and discard redundant information, which improves its stability in time-series forecasting tasks. However, a unidirectional LSTM only models temporal information in one direction. To obtain richer contextual representations, this study adopts a BiLSTM architecture. The BiLSTM consists of a forward LSTM and a backward LSTM, which process the input sequence in opposite directions. It should be noted that the bidirectional operation is performed only within the historical sliding window. The prediction target is outside this input window, and no observations after the target time are used during training or testing. Therefore, the BiLSTM does not violate the causality requirement of time-series forecasting. Through this design, the model can capture contextual dependencies within the available historical sequence and provide more informative temporal features for the subsequent SimAM module. The detailed structure of the BiLSTM module is illustrated in Figure 3.

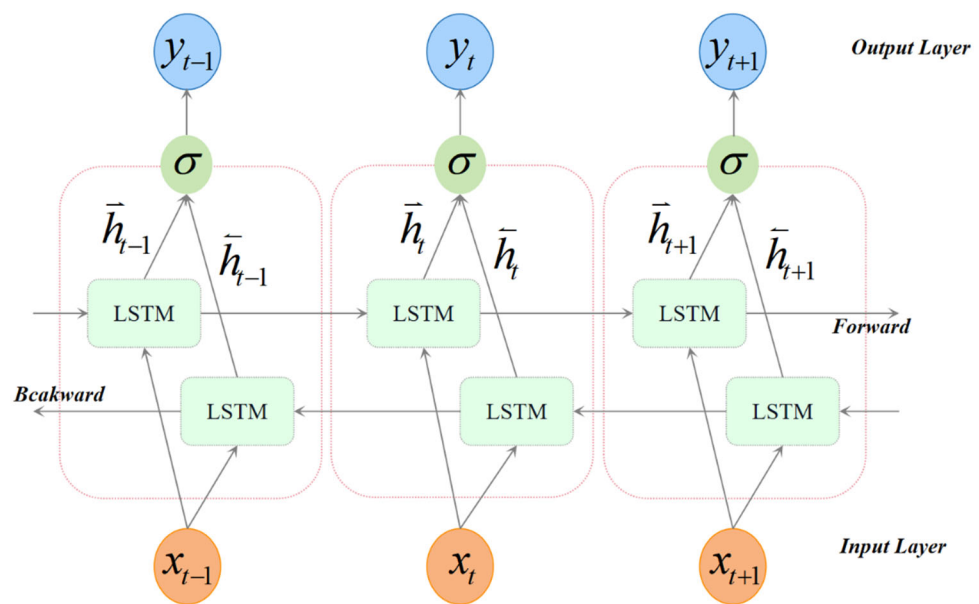


Figure 3. Structure of the Bidirectional Long Short-Term Memory (BiLSTM) Module.

Let the input to the BiLSTM module be the feature sequence generated by the TCN, denoted by, $H^{ECA} = [x_1, x_2, \dots, x_w]$, where w denotes the length of the time window, $x_t \in \mathbb{R}^c$ denotes the feature vector at time step t , and c represents the feature dimension. At each time step t , the LSTM unit updates its historical memory, incorporates the current input, and produces the hidden-state output through the forget gate, input gate, and output gate, respectively. The computation process is given as follows:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (5)$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (6)$$

$$\tilde{c}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \quad (7)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \quad (8)$$

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

$$h_t = o_t \odot \tanh(c_t) \quad (10)$$

where $\odot$ represents the Hadamard product, and W_f , W_i , W_c , W_o , together with b_f , b_i , b_c and b_o , denote the corresponding weight matrices and bias terms for each gate, respectively. In addition, c_t denotes the cell state, and h_t denotes the hidden state. In the bidirectional structure, the sequence is processed separately by the forward LSTM and the backward LSTM, yielding the forward hidden state $\vec{h}_t$ and the backward hidden state $\bar{h}_t$, respectively. Therefore, the bidirectional output at time step t can be expressed as:

$$h_t^{BiLSTM} = \text{Concat}(\vec{h}_t, \bar{h}_t) \quad (11)$$

where $\text{Concat}(\cdot)$ denotes the vector concatenation operation. After bidirectional recurrent computation over W time steps, the temporal feature sequence output by the BiLSTM can be written as:

$$H^{BiLSTM} = [h_1^{BiLSTM}, h_2^{BiLSTM}, \dots, h_W^{BiLSTM}] \quad (12)$$

where, $H^{BiLSTM} \in \mathbb{R}^{W \times 2H}$, H denotes the hidden-state dimension of the unidirectional LSTM, and the factor 2 indicates the concatenation of bidirectional features. While preserving the original temporal dimension, this output sequence integrates both forward and backward contextual information, thereby providing a more comprehensive representation of the long-term and short-term dynamic dependencies in the wastewater flow sequence. Subsequently, the output features of the BiLSTM are fed into the SimAM attention module, which further improves the predictive performance of the model by adaptively enhancing key temporal features and suppressing redundant information.

2.3. SimAM Attention Module

Although the TCN-BiLSTM module can extract high-dimensional spatiotemporal features and model both long- and short-term dependencies, the BiLSTM's hidden features do not contribute equally across different channels and temporal positions in wastewater flow forecasting tasks. In complex industrial monitoring environments, some channels may contain redundant information. If these features are directly fed into subsequent layers, the model's ability to focus on critical fluctuation patterns may be weakened. To address this issue, the SimAM is introduced between the BiLSTM and the final regression layer to assign attention weights across the channels and temporal dimensions adaptively. In this way, features that are more discriminative for the forecasting task can be emphasized, while irrelevant and redundant information is suppressed. The structure of the SimAM module is illustrated in Figure 4.

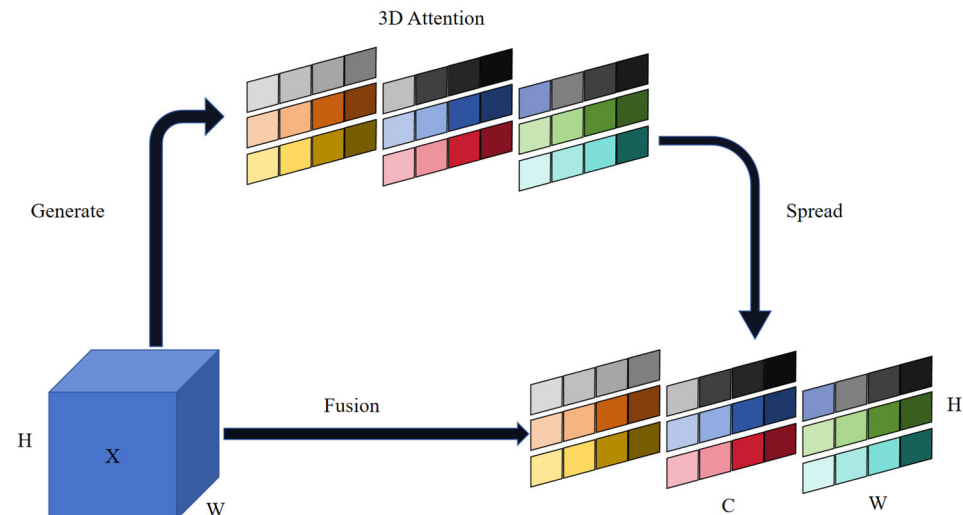


Figure 4. Structure of the Simple Attention Module (SimAM).

The attention mechanism is inspired by the selective attention of human vision, enabling a model to emphasize informative features during high-dimensional information processing. SE recalibrates channel responses using global pooling and fully connected layers, whereas CBAM sequentially applies channel and spatial attention through additional learnable operations. In contrast, SimAM derives three-dimensional attention weights from an energy function without introducing extra trainable fully connected or convolutional layers or reducing the feature dimension. In the proposed TCN-BiLSTM-SimAM model, SimAM recalibrates the feature sequences generated by BiLSTM, strengthens features that are more relevant to wastewater-flow forecasting, and attenuates less informative responses. Its parameter-free attention design limits the additional model complexity and is therefore suitable for computationally constrained real-time monitoring applications.

Let the feature tensor output by the BiLSTM module be denoted as $H^{BiLSTM} \in \mathbb{R}^{B \times W \times 2H}$, where B is the batch size, W is the time-window length, and $2H$ is the number of channels obtained by concatenating the bidirectional hidden states. The SimAM computation first performs local energy estimation across the channel and temporal dimensions of the input features to measure the importance of each feature position. For the c -th channel at the t -th time step, the energy function $e_{t,c}$ is defined as:

$$e_{t,c} = (H_{t,c}^{BiLSTM} - \mu_c)^2 + \lambda \sum_{i \in N(c)} (H_{t,c}^{BiLSTM} - H_{t,i}^{BiLSTM})^2 \quad (13)$$

where $H_{t,c}^{BiLSTM}$ denotes the feature value of the c -th channel at time step t , $N(c)$ denotes the local temporal neighbourhood within the historical input window, and μ_c is the mean feature value of the c -th channel in $N(c)$. The parameter λ is a regularization hyperparameter that controls the sensitivity of the SimAM attention weight. A smaller λ makes the attention response more sensitive to local feature changes, whereas a larger produces smoother attention weights. In this study, λ is optimized by IHHO together with other model hyperparameters.

Subsequently, the energy function is mapped to the attention weight $\alpha_{t,c}$:

$$\alpha_{t,c} = \sigma(-e_{t,c}) \quad (14)$$

where $\sigma(\cdot)$ denotes the Sigmoid activation function, which maps the attention weights to the interval $(0, 1)$.

Finally, the attention weights are used to perform element-wise reweighting of the original BiLSTM features, yielding the calibrated feature tensor:

$$H_{t,c}^{SimAM} = \alpha_{t,c} \cdot H_{t,c}^{BiLSTM} \quad (15)$$

The output tensor $H^{SimAM} \in \mathbb{R}^{B \times W \times 2H}$ preserves the same dimensions as the input, while its feature distribution is adaptively optimized through weighted refinement. It significantly enhances the model's ability to capture key variation patterns in wastewater flow and provides highly representative input features for the subsequent fully connected regression layer.

In summary, the proposed TCN-BiLSTM-SimAM model combines three complementary functions. The TCN extracts local and multi-scale temporal features. The BiLSTM captures contextual dependencies within the historical input window. The SimAM module enhances informative features. However, this cascaded structure also introduces more hyperparameters, such as window length, convolution kernel size, hidden units, and learning rate. These hyperparameters may interact with each other, making the training process a high-dimensional and nonconvex optimization problem. Manual tuning and grid search are inefficient in such a large search space. Therefore, an improved HHO algorithm is introduced to optimize key hyperparameters adaptively.

Despite its significant advantages in feature representation, the multi-module cascaded architecture of the proposed model also introduces considerable parameter sensitivity. Key hyperparameters, such as the input sequence window length, convolution kernel size, number of hidden units in the BiLSTM layer, and learning rate, exhibit complex interactions and couplings. As a result, the training process becomes a high-dimensional, nonconvex, and computationally intensive optimization problem. Traditional manual tuning strategies or grid search methods are often unable to identify the global optimum in such a large hyperparameter space, which limits the full predictive potential of the model.

Therefore, to fully exploit the advantages of the TCN-BiLSTM-SimAM integrated model, it is necessary to introduce an efficient swarm intelligence optimization algorithm that possesses both strong global search capability and fast convergence. Such an algorithm can enable adaptive, robust, and accurate optimization of key hyperparameters, thereby ensuring superior performance in complex industrial wastewater flow-forecasting scenarios.

3. Improved Harris Hawks Optimisation

To address the complex optimization problem arising during hyperparameter tuning of the TCN-BiLSTM-SimAM hybrid model, this study proposes an IHHO algorithm. The proposed algorithm improves conventional HHO from four aspects. (1) First, a hybrid initialization strategy combines Latin Hypercube Sampling with opposition-based learning. This strategy improves the uniformity and coverage of the initial population and enhances the quality of initial solutions. (2) Second, an early-stage Differential Evolution (DE) mutation mechanism is introduced to strengthen global exploration and accelerate the discovery of promising regions. (3) Third, a stagnation detection and diversity injection mechanism is used to monitor population stagnation during the search process. By injecting diversity when stagnation occurs, this mechanism suppresses premature convergence and maintains search activity and stability. (4) Fourth, a late-stage Lévy flight perturbation strategy is adopted to reinforce local fine-grained search in promising regions. It also improves the approximation accuracy to the global optimum. Overall, IHHO demonstrates clear advantages in hyperparameter optimization accuracy, convergence

speed, and robustness, and provides efficient optimization support for the TCN-BiLSTM-SimAM hybrid model.

3.1. Conventional HHO Algorithm

The HHO algorithm is a metaheuristic optimization method that simulates swarm intelligence. It is inspired by the hunting behaviour of Harris hawks in the wild. By modelling three strategies, encircling prey, spiral attack, and random search, HHO explores the solution space to find the global optimum.

Specifically, the algorithm treats the current global optimal solution as the prey, denoted by X^* , while the remaining individuals X update their positions according to the prey's location. The position update mechanism mainly includes the following three strategies:

(1) Encircling Prey: the individuals update their positions around the current optimal solution. The mathematical model is given by:

$$D = |C \cdot X^*(t) - X(t)| \quad (16)$$

$$X(t+1) = X^*(t) - A \cdot D \quad (17)$$

where $A = 2a \cdot r - a$, $C = 2r$, and $r \in [0, 1]$ are a random vectors. The convergence factor a decreases linearly with the number of iterations.

(2) Spiral Attack: This strategy simulates the hawks approaching the prey along a spiral path, which can be expressed as:

$$X(t+1) = D' \cdot e^{bl} \cdot \cos(2\pi l) + X^*(t) \quad (18)$$

where $D' = |X^*(t) - X(t)|$, b is the spiral constant, and $l \in [-1, 1]$.

(3) Search for Prey: when $|A| \geq 1$, the individuals update their positions according to a randomly selected individual X_{rand} , to enhance the global exploration capability:

$$D = |C \cdot X_{rand} - X(t)| \quad (19)$$

$$X(t+1) = X_{rand} - A \cdot D \quad (20)$$

Although HHO has shown advantages in many engineering optimization problems, it still has limitations in this study. These advantages mainly include its simple mechanism, small number of parameters, and relatively strong global search capability. However, the hyperparameter optimization problem of the TCN-BiLSTM-SimAM model is high-dimensional, nonconvex, and strongly coupled. Under this condition, conventional HHO faces several limitations. These limitations are mainly reflected in four aspects. (1) First, the quality of population initialization is insufficient. Conventional HHO usually generates the initial population randomly. This may lead to an uneven distribution of individuals in the search space and unstable initial solution quality. As a result, the coverage and effectiveness of the early search stage are reduced, making it difficult to identify promising regions rapidly. (2) Second, the global exploration capability is limited. Although conventional HHO has some exploratory ability in the early search stage, it lacks a stronger external mutation mechanism. Therefore, the search range may not be sufficiently expanded in complex high-dimensional spaces. (3) Third, conventional HHO lacks an effective stagnation-handling mechanism. When the population is trapped in a local optimum or enters global stagnation, it is difficult to restore population diversity in time. This may cause premature convergence and limit further improvement in optimization

accuracy. (4) Fourth, the local exploitation capability in the later stage is inadequate. Although conventional HHO gradually shifts to local exploitation as the search progresses, it lacks a targeted fine-grained perturbation mechanism. This restricts its deep search ability in promising regions and makes it difficult to approximate the global optimum with high precision.

3.2. Improvement Strategies

To overcome the limitations of the conventional HHO algorithm in terms of population initialization quality, global exploration capability, ability to escape from stagnation, and local fine-grained exploitation, this study proposes improvement strategies from four perspectives. The underlying principles, specific mechanisms, and mathematical formulations of these strategies are described as follows.

3.2.1. Hybrid Population Initialisation Strategy

To address the limitations of conventional HHO associated with random initialization, including uneven initial population distribution and strong randomness in the search process, this study introduces a hybrid population initialization strategy. The proposed strategy combines Latin Hypercube Sampling (LHS) with Opposition-Based Learning (OBL) and is designed to improve solution-space coverage and initial population diversity, thereby providing a broader candidate-solution basis for subsequent iterations. However, the extent of the resulting diversity improvement was not quantitatively evaluated in the present study.

First, to improve the uniformity of the initial population distribution in the search space, this study employs Latin Hypercube Sampling (LHS) to generate the initial population. Let the population size be N , the problem dimension be D , and the value range of the variable in the j -th dimension be $[lb_j, ub_j]$. LHS divides the search interval of each dimension into N equiprobable subintervals and randomly samples one point from each subinterval. The initial component of the i -th individual in the j -th dimension can be expressed as:

$$X_{i,j} = lb_j + \frac{\Pi_j(i) + \varepsilon}{N} (ub_j - lb_j) \quad (21)$$

where Π_j is a random permutation of $\{0, 1, \dots, N-1\}$, and $\varepsilon \sim U(0, 1)$ is a random perturbation term. Statistically, this mechanism ensures that the sample points in each dimension are uniformly distributed over the entire search space.

Next, to further improve the quality of the population and enhance the exploration ability in potentially promising regions, an opposition-based learning mechanism is introduced based on LHS initialization. For a generated initial individual, X_i its opposite individual X'_i is defined as:

$$X'_{i,j} = lb_j + ub_j - X_{i,j} \quad (22)$$

By evaluating the fitness of both the original and the opposite population and retaining the top N individuals with better performance as the initial population, the quality of the initial solutions can be effectively improved, thereby enhancing the algorithm's global exploration capability in the early stage of the search.

3.2.2. Early-Stage DE Mutation Mechanism

To address the limitations of conventional HHO in the early search stage, particularly its insufficient global exploration capability and the tendency of the population to become

confined to a limited search range in complex high-dimensional spaces, this study introduces a Differential Evolution (DE) mutation mechanism in the early stage of the algorithm to enhance the population's global exploration performance.

Specifically, during the early stage of iteration, for the current individual X_i^t , three distinct individuals, X_{r1}^t , X_{r2}^t , and X_{r3}^t , are randomly selected to construct the differential mutation vector:

$$V_i^t = X_{r1}^t + F \cdot (X_{r2}^t - X_{r3}^t) \quad (23)$$

where F is the differential scaling factor used to control the mutation step size. Subsequently, a crossover operation is performed between the mutation vector and the current individual to generate the candidate individual U_i^t .

$$U_{i,j}^t = \begin{cases} V_{i,j}^t, & \text{if } \text{rand}_{i,j} \leq CR \text{ or } j = j_{rand} \\ X_{i,j}^t, & \text{otherwise} \end{cases} \quad (24)$$

where CR denotes the crossover probability, and $\text{rand}_{i,j}$ is a random number in the interval $[0, 1]$, and j_{rand} is a randomly selected dimension index used to ensure that the trial individual has at least one dimension inherited from the mutant vector. If the fitness of the candidate individual U_i is better than that of the current individual, it replaces the current individual in the next generation. This strategy can significantly expand the search radius for individuals in the early stage of the search, improve the population's ability to escape local regions, and thereby accelerate the discovery of promising regions.

3.2.3. Stagnation Detection and Diversity Injection Mechanism

To address the tendency of conventional HHO to fall into local optima during the middle stage of iteration due to reduced population diversity, as well as the subsequent stagnation that may occur in the later search process, this study proposes a stagnation-detection and diversity-injection mechanism. The core idea of this strategy is to monitor whether the algorithm achieves meaningful fitness improvement over several consecutive iterations. When no effective improvement is observed, the current population is likely to have become excessively concentrated in a certain region. In this case, the search direction tends to become increasingly uniform. Without timely intervention, the population may further lose its exploration capability and converge prematurely. Therefore, stagnation detection is combined with directed perturbation to restore population diversity while preserving existing high-quality information. This enables the algorithm to escape the attraction basin of local optima and continue effective search.

Population diversity reflects the spatial dispersion of candidate solutions. A larger diversity value indicates that individuals are more widely distributed, which benefits global exploration, whereas a smaller value indicates population concentration and stronger local exploitation. Therefore, diversity is not monotonically better when it is larger or smaller. For an effective diversity-recovery mechanism, the diversity value should temporarily increase after stagnation is detected, indicating that the reconstructed individuals expand the search range. It should then gradually decrease as the population reconverges toward promising regions. This dynamic pattern provides evidence that the mechanism restores exploration without preventing subsequent convergence [39].

Specifically, at the end of each iteration, the algorithm records the current global best fitness $f^*(t)$. If, over T_{stall} consecutive iterations, the improvement in the global best fitness remains smaller than the preset threshold δ , the algorithm is considered to have entered a stagnation state. The corresponding criterion can be expressed as:

$$|f^{ns}(t) - f^{ns}(t - T_{stall})| < \delta \quad (25)$$

where T_{stall} denotes the length of the stagnation detection window, and δ is the minimum effective improvement threshold. This criterion helps prevent misjudging the algorithm's state due to minor fluctuations in fitness, thereby making stagnation detection more robust.

Once stagnation is detected, directly applying large-scale random reinitialization to the entire population can quickly restore diversity, but it may severely disrupt the valuable search experience accumulated in the early stage and reduce the stability of algorithm convergence. To address this issue, this study adopts a diversity recovery strategy based on local elite preservation and directional injection: only a subset of poorly performing individuals is perturbed and reconstructed, while elite individuals and their neighbourhood structures are retained. Let X_{worst} denote the poorly performing individuals in the current population. Then, the positions of the reconstructed individuals are defined as:

$$X_{new} = X_{best} + \alpha \cdot randn(1, D) \odot (UB - LB) \quad (26)$$

where X_{best} denotes the current global best individual, α is the perturbation intensity coefficient, $randn(1, D)$ represents a random vector following the standard normal distribution, and $\odot$ denotes element-wise multiplication.

Moreover, UB and LB are the upper and lower bound vectors of the search space, respectively. This update strategy does not generate new solutions in a completely random manner; instead, it uses the current best individual as the centre and injects new individuals with random directions and controllable magnitudes into its neighbourhood. In this way, new search directions are introduced while preserving beneficial genetic information.

Furthermore, to improve the effectiveness of diversity recovery, this study defines injection intensity as a dynamic variable that reflects the degree of stagnation. When the stagnation duration is short, only a small perturbation is applied to fine-tune the search direction. As the stagnation duration increases further, α is appropriately enlarged to expand the re-exploration range. Its dynamic adjustment can be expressed as:

$$\alpha(t) = \alpha_0 \left(1 + \frac{C_{stall}}{T_{stall}}\right) \quad (27)$$

where α_0 is the basic perturbation intensity, and C_{stall} denotes the current accumulated number of stagnation iterations. This design enables the algorithm to maintain mild correction during slight stagnation, while rapidly expanding the search radius during severe stagnation, thereby improving its ability to escape from stagnation.

From the perspective of search dynamics, this mechanism serves two important functions. On the one hand, the trigger mechanism based on the stagnation criterion ensures that diversity injection is activated only when necessary, thereby avoiding excessive interference with the normal convergence process. On the other hand, by introducing local perturbations relative to the current best individual, the algorithm not only ensures that newly generated individuals retain relatively high feasibility and potential fitness, but also effectively prevents the population from continuously clustering around a single solution. Through the coordinated effect of stagnation detection and diversity injection, IHHO can significantly enhance population activity and improve its ability to escape local optima during the middle search stage, thereby preserving greater potential for subsequent fine-grained exploitation.

3.2.4. Late-Stage Lévy Flight Perturbation Strategy

To address the limitations of conventional HHO in the later search stage, particularly its insufficient local exploitation capability and inadequate fine-grained search in the neighbourhood of the optimal solution, this study introduces a Lévy flight perturbation strategy to enhance the fine search ability of individuals around promising regions. The basic idea of this strategy is as follows. During the later stage of iteration, the population gradually converges toward several potentially promising regions. At this stage, the main task of the algorithm shifts from large-scale global exploration to deep local exploitation near the optimal solution. If a single fixed-scale update strategy is still used, the search step may become unsuitable. A step that is too large may cause the search to overshoot the optimum, whereas a step that is too small may lead to repeated oscillations within a local region. Lévy flight has a heavy-tailed distribution with many small steps and occasional large jumps. This property helps balance local exploitation and global exploration. Therefore, it is suitable as a random perturbation strategy in the later stage of the search.

Specifically, during the later stage of the iterative process, a Lévy flight perturbation is imposed on the current individual position X_i^t . The update rule is defined as:

$$X_i^{t+1} = X_i^t + \beta \cdot \text{Levy}(\lambda) \odot (X_i^t - X_{best}^t) \quad (28)$$

where β is the step-size control factor, $\text{Levy}(\lambda)$ denotes a random variable following a Lévy distribution, λ is the distribution parameter, and X_{best}^t represents the current global best individual. This equation indicates that the perturbation direction is related to the positional difference between the individual and the current best solution, while the Lévy random step size controls the perturbation magnitude. In this way, the search is guided toward promising regions while preserving the individual's non-uniform, non-Gaussian random exploration within the local neighbourhood.

In practical implementation, the random step size of the Lévy flight is commonly generated by Mantegna's method and can be expressed as:

$$\beta(t) = \beta_0 \left(1 - \frac{t}{T_{max}}\right)^\gamma \quad (29)$$

where β_0 is the initial step-size coefficient, T_{max} is the maximum number of iterations, and γ is the decay exponent. This strategy enables the algorithm to retain a certain degree of jumping capability in the later stage to avoid falling into local suboptimal solutions. As the algorithm approaches termination, the step size gradually decreases, thereby allowing a finer search in the optimal region.

In addition, given the inherent randomness of Lévy flights, applying the same magnitude of perturbation to all individuals in the later search stage may excessively disturb those already close to the optimal solution. To address this issue, this study applies Lévy perturbation only to non-elite individuals or to those whose fitness improves slowly, while adopting a conservative update strategy for the current elite individuals in order to balance population stability and search flexibility. In this manner, elite individuals are responsible for maintaining the convergence direction, whereas non-elite individuals conduct supplementary search within the neighbourhood of the optimum. Together, these two groups promote high-precision optimization in the later stage of the algorithm.

Overall, the late-stage Lévy flight perturbation strategy does not merely introduce additional randomness. Instead, by combining a heavy-tailed step-size distribution with a dynamic decay mechanism, it establishes a search pattern during the local exploitation stage in which small-step refinement serves as the primary mode, with occasional large-step jumps as a supplementary mechanism. This strategy significantly enhances IHHO's fine-grained search capability near promising regions, improves the accuracy with which

the global optimum is approached, and reduces the risk of the algorithm becoming trapped in minor local depressions during the later search stage. Consequently, IHHO can further improve the stability and accuracy of the optimization results while maintaining a favourable convergence rate. The pseudocode of IHHO is presented in Algorithm 1.

Algorithm 1 An Improved Harris Hawks Optimization Algorithm (IHHO)

Require: Population size N , maximum iterations T_max , problem dimension D , search space bounds $[LB, UB]$, objective function $f(\cdot)$, DE parameters F and CR , stagnation parameters T_stall and δ , perturbation parameters α_0 , β_0 , γ , q , and λ

Ensure: Global best solution X_{best} and its fitness f^*

```

1: //Strategy 1: Hybrid initialization strategy
2: Initialize population  $X$  using Latin Hypercube Sampling (LHS).
3: Generate the opposite population  $X'$  according to  $X'_{ij} = lb_j + ub_j - X_{ij}$ .
4: Evaluate all individuals and retain the best  $N$  solutions as the initial population.
5: Determine the initial global best solution  $X_{best}$  and fitness  $f^*$ .
6: Initialize stagnation counter  $C_{stall} = 0$ .
7: Set  $t = 1$ .
8: while  $t \leq T_{max}$  do
9:   Update the diversity-injection intensity  $\alpha(t)$  according to  $\alpha(t) = \alpha_0(1 + C_{stall}/T_{stall})$ .
10:  Update the Lévy step-size coefficient  $\beta(t)$  according to  $\beta(t) = \beta_0(1 - t/T_{max})^\gamma$ .
11:  //Strategy 2: Early-stage DE mutation mechanism
12:  for  $i = 1$  to  $N$  do
13:    if  $t/T_{max} \leq \theta_{DE}$  then
14:      //Generate a DE trial individual
15:      Randomly select three distinct individuals  $X_{t-1r1}$ ,  $X_{t-1r2}$ , and  $X_{t-1r3}$ .
16:      Generate the mutation vector  $V_{i^*t} = X_{t-1r1} + F \cdot (X_{t-1r2} - X_{t-1r3})$ .
17:      Generate the trial individual  $U_{i^*t}$  by crossover using  $CR$  and  $j_{rand}$ .
18:      Set  $X_{new} = U_{i^*t}$ .
19:    else
20:      Generate  $X_{new}$  using the original HHO position-update rule.
21:    end if
22:  //Strategy 4: Late-stage Lévy flight perturbation strategy
23:  if  $t/T_{max} > \theta_L$  and  $i \notin E$  then
24:     $X_{new} = X_{new} + \beta(t) \cdot Levy(\lambda) \odot (X_{new} - X_{best})$ .
25:  end if
26:   $X_{new} = X_{new} + \beta(t) \cdot Levy(\lambda) \odot (X_{new} - X_{best})$ .
27:  Project  $X_{new}$  into  $[LB, UB]$ .
28:  Evaluate  $f_{new} = f(X_{new})$ .
29:  if  $f_{new} < f(X_{i^*t})$  then
30:     $X_{i^*t+1} = X_{new}$ .
31:    if  $f_{new} < f^*$  then
32:       $X_{best} = X_{new}$ ,  $f^* = f_{new}$ .
33:      Reset  $C_{stall} = 0$ .
34:    end if
35:  end if
36: end for
37: //Strategy 3: Global stagnation detection
38: if the global best fitness is not improved then
39:    $C_{stall} = C_{stall} + 1$ .
40: end if
41: if  $|f^*(t) - f^*(t - T_{stall})| < \delta$  then
42:  //Strategy 3: Diversity injection by restricted OBL re-injection

```

```

43:   Select poorly performing individuals  $X_{\text{worst}}$  and reconstruct them by  $X_{\text{new}} = X_{\text{best}} + \alpha(t) \cdot \text{randn}(1,D) \odot (UB - LB)$ .
44:   Re-evaluate these individuals and update  $X_{\text{best}}, f^*$ .
45:   Reset  $C_{\text{stall}} = 0$ .
46: end if
47:   Set  $t = t + 1$ .
48: end while
49: return  $X_{\text{best}}, f^*$ 

```

3.3. Algorithm Flow

The IHHO algorithm proposed in this study establishes a closed-loop optimization framework for hyperparameter tuning of the TCN-BiLSTM-SimAM hybrid model by integrating hybrid population initialization, early-stage DE mutation enhancement, stagnation detection and diversity injection, and late-stage Lévy flight-based fine-tuning. The overall framework is designed to achieve an effective balance between global exploration and local exploitation in a high-dimensional, nonconvex search space with strong parameter coupling, as illustrated in Figure 5.

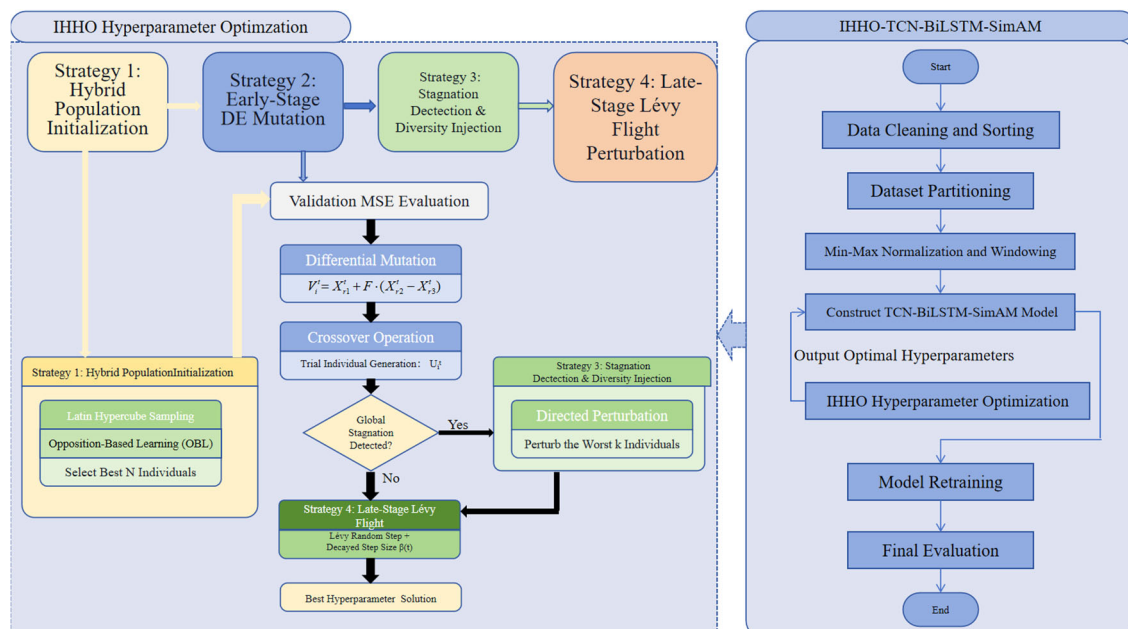


Figure 5. Overall Framework of the IHHO-Optimised TCN-BiLSTM-SimAM Model.

First, the key hyperparameters to be optimized and their corresponding search ranges are determined. In this study, the optimization variables include the convolutional channel number C and kernel size k the TCN module, the number of hidden units H in the BiLSTM layer, the hyperparameter λ of the SimAM module, and the learning rate η used during model training. For these key parameters, IHHO first adopts a hybrid population initialization strategy that combines the stratified sampling property of Latin Hypercube Sampling (LHS) with the opposition-based learning mechanism. This strategy ensures good uniformity and coverage of the initial solutions within the search space, reduces the blindness of the early search stage, and provides a high-quality population basis for subsequent iterations. After entering the iterative optimization stage, the algorithm uses the validation Root Mean Square Error (Validation RMSE) as the fitness criterion for evaluating each individual, thereby avoiding overfitting to the training set at the expense of generalization performance. At the beginning of each generation, IHHO dynamically adjusts its search strategy according to the current iteration stage. In the early search stage,

a Differential Evolution (DE) mutation mechanism is introduced to enhance individual updates. By increasing the differential perturbation amplitude among candidate solutions, the algorithm improves the population's global exploration capability in the high-dimensional parameter space, thereby accelerating the discovery of potentially promising regions. Meanwhile, the algorithm continuously tracks the current global best individual and the overall fitness distribution of the population to maintain effective guidance toward promising search directions. As iterations proceed, IHHO further monitors the population state using the stagnation detection and diversity injection mechanisms. When no significant improvement in the best fitness value is observed over several consecutive generations, this indicates that the algorithm may have become trapped in a local optimum region or entered a state of global stagnation. In this case, IHHO activates the diversity injection strategy by applying directed perturbations to, or regenerating, individuals with poor fitness, thereby restoring the population's distributional breadth and preventing premature convergence caused by excessive population aggregation. In this way, the algorithm preserves the current high-quality search information while reactivating exploration toward other potentially promising regions.

In the later search stage, as the population gradually approaches the optimal region, IHHO further introduces a Lévy flight perturbation strategy to refine individual positions. By exploiting the heavy-tailed property of the Lévy distribution, which produces a random walk pattern dominated by small-step searches with occasional large jumps, the algorithm can achieve higher-resolution local exploitation in the neighbourhood of the optimum while avoiding entrapment in minor local depressions. Through this late-stage fine-tuning mechanism, IHHO significantly improves the accuracy and stability of hyperparameter optimization. After the coordinated iteration and deep optimization of these multi-stage strategies, once the preset maximum number of iterations is reached or the convergence criterion is satisfied, IHHO outputs the globally optimal hyperparameter combination that minimizes the validation error and uses it to construct the final TCN-BiLSTM-SimAM forecasting model. As a result, the model can more fully exploit the local feature extraction capability of TCN, the bidirectional temporal modelling capability of BiLSTM, and the key feature enhancement capability of SimAM, while also achieving higher prediction accuracy and stronger generalization performance in complex wastewater flow forecasting tasks.

4. Experiments and Analysis

4.1. Experimental Environment

(1) Dataset Description

The dataset used in this study was collected from measured flow records at a wastewater treatment plant in Quanzhou, China. The sampling interval was 5 min, and a total of 254 records were obtained, covering the period from 00:00 to 21:00 on 1 April 2024. These data adequately reflect the operating conditions and daily fluctuation characteristics of the wastewater treatment plant, thereby providing reliable real-world samples for short-term flow forecasting. In model development, a sliding-window strategy was adopted for input construction, using historical flow observations from the previous 24 time steps to predict the flow at the next time step. This setting not only captures intraday periodic variations but also enables the model to learn the dynamic characteristics of short-term fluctuations and abrupt changes. To ensure the scientific rigour of model training and its generalization capability, the dataset was split chronologically: 80% of the samples were used for training, and the remaining 20% for validation to evaluate predictive performance on unseen data.

(2) Experimental Configuration

All experiments in this study were conducted in a CPU-based computing environment without GPU acceleration. Model construction, training, and prediction were implemented in MATLAB R2023b (The MathWorks, Inc., Natick, MA, USA). Given that the data scale and model complexity considered in this study were within a manageable range, the CPU environment was sufficient to meet the computational requirements while ensuring training stability and reproducibility of results. The main hardware and software configurations of the experimental platform are summarised in Table 1.

Table 1. Experimental Environment Configuration.

Name	Setting
CPU	13th Gen Intel® Core™ i7-13620H
Frequency	2.50 GHz
RAM	16.0 GB
Storage	512 GB
GPU	NVIDIA GeForce RTX 4050 Laptop GPU
Operating system	Windows 10
Software	MATLAB R2023b (The MathWorks, Inc., Natick, MA, USA)

(3) Parameter Settings

Table 2 presents the core parameter settings for the TCN and BiLSTM models, the training process, and the Improved Harris Hawks Optimisation (IHHO) algorithm, including the early-stage DE threshold $\theta_{DE} = 0.5T_{max}$, the stagnation window $T_{stall} = 25$, the fitness-improvement threshold $\delta = 1 \times 10^{-12}$, and the elite selection ratio $q = 20$. These settings clarify the triggering conditions of the IHHO improvement strategies and provide a basis for experimental reproducibility and result analysis.

Table 2. Parameter Setting.

Module	Parameter Name	Value/Range
TCN	Number of TCN Layers	3
TCN	Dilation Factor	[1, 2, 4]
TCN	Kernel Size	2~7 (Optimized by IHHO)
TCN	Number of Convolution Channels	16~64 (Optimized by IHHO)
TCN	Padding	causal
TCN	Activation Function	ReLU
BiLSTM	Number of Hidden Units	8~64 (Optimized by IHHO)
BiLSTM	Output Mode	sequence
SimAM	attention parameter λ	0.5~2.0 (Optimized by IHHO)
SimAM	Bias	0.5
SimAM	Residual Strength	0.7
Training	Optimizer	Adam
Training	Maximum Number of Iterations	400
Training	Learning Rate	1×10^{-4} ~ 1×10^{-2} (Optimized by IHHO)
IHHO	Population Size	30
IHHO	Maximum Number of Iterations	40
IHHO	Number of Chaotic Map Iterations	2
IHHO	Initial DE Mutation Probability	0.6
IHHO	DE Crossover Probability	0.9

IHHO	Convergence Factor	$2 \cdot (1 - t/\text{MaxIt})$
IHHO	Jump Adjustment Factor	$1 + \cos(\pi \cdot t/\text{MaxIt})$
IHHO	Lévy Perturbation Factor	$(1 - t/\text{MaxIt})^2$
IHHO	Stagnation Window T_{stall}	25
IHHO	Early-Stage DE Threshold θ_{DE}	$0.5T_{\text{max}}$
IHHO	Fitness Improvement Threshold δ	1×10^{-12}
IHHO	Elite Selection Ratio q	0.2

(4) Evaluation Metrics

To comprehensively and objectively evaluate the performance of each forecasting model in short-term wastewater flow prediction, this study adopts three commonly used evaluation metrics: Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and the coefficient of determination (R^2). Among them, MAE and RMSE reflect the absolute magnitude of prediction errors, while R^2 measures the model's ability to explain the variation trend of the true sequence. Let the ground-truth sequence in the test set be denoted by $\{y_i\}_{i=1}^n$, the predicted sequence by, $\{\hat{y}_i\}_{i=1}^n$ and the number of samples by n . Mean Absolute Error (MAE) is used to measure the average magnitude of the deviation between the predicted and true values, and is defined as:

$$\text{MAE} = \frac{1}{n} \sum_i^n |y_i - \hat{y}_i| \quad (30)$$

Root Mean Square Error (RMSE) is an indicator obtained by averaging the squared prediction errors and then taking the square root. It is more sensitive to larger prediction errors and can more prominently reflect model performance under scenarios involving peak loads or abrupt fluctuations. It is defined as:

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (31)$$

The coefficient of determination (R^2) is used to measure the extent to which the model explains the variance of the true values, thereby reflecting the goodness of fit of the model. It is defined as:

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y}_i)^2} \quad (32)$$

4.2. Model Validation

To evaluate the convergence stability and generalization capability of the TCN-BiLSTM-SimAM model for short-term wastewater flow forecasting, the model was trained for 200 iterations on a one-day wastewater flow dataset collected in Quanzhou, with a 5-min sampling interval. Meanwhile, the variation in the training loss, measured by RMSE, was recorded throughout the training process. Figure 6 illustrates the trend in the training RMSE over iterations.

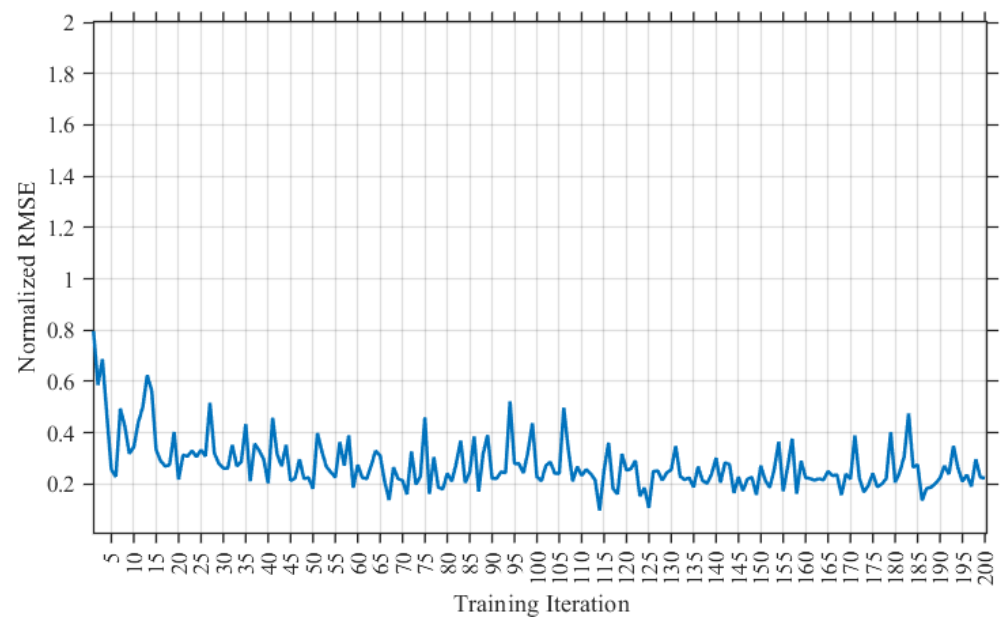


Figure 6. Normalized training RMSE over the first 200 training iterations.

As shown in Figure 6, the RMSE curve on the training set reflects the convergence behaviour of the model during training. Overall, the RMSE exhibits a clear downward trend as the number of training iterations increases, indicating that the TCN-BiLSTM-SimAM model continuously optimizes its parameters and improves its fit to the training samples. In the early stage of training, approximately within the first 50 iterations, the curve fluctuates markedly, suggesting that the model is actively exploring the parameter space and rapidly adjusting network weights to adapt to the characteristics of the data. As training proceeds, the RMSE curve gradually stabilises and eventually remains at a relatively low level, indicating that the model is approaching convergence with small, stable fitting errors. Although minor fluctuations are still observed, the overall trend remains clear, demonstrating that the model can effectively capture the dominant fluctuation patterns and key features of the training data, thereby providing a robust parameter basis for subsequent validation-set prediction. These results indicate that the proposed TCN-BiLSTM-SimAM architecture achieves high fitting accuracy and satisfactory convergence efficiency on the training set, thereby offering reliable support for short-term wastewater flow forecasting.

As shown in Figure 7, the mean SimAM attention response decreases rapidly from approximately 0.68 to 0.64 during the early stage and then declines more gradually. After approximately 100 iterations, the curve exhibits only small fluctuations and gradually stabilises at around 0.625. This trend indicates that the attention response changes markedly in the initial stage and subsequently approaches a relatively stable level. However, because the plotted values were obtained by interpolating the attention responses of the final model rather than by recording the SimAM weights directly during training, the figure should be interpreted as a smoothed visualisation of the attention-response trend rather than direct evidence of iteration-wise weight optimisation.

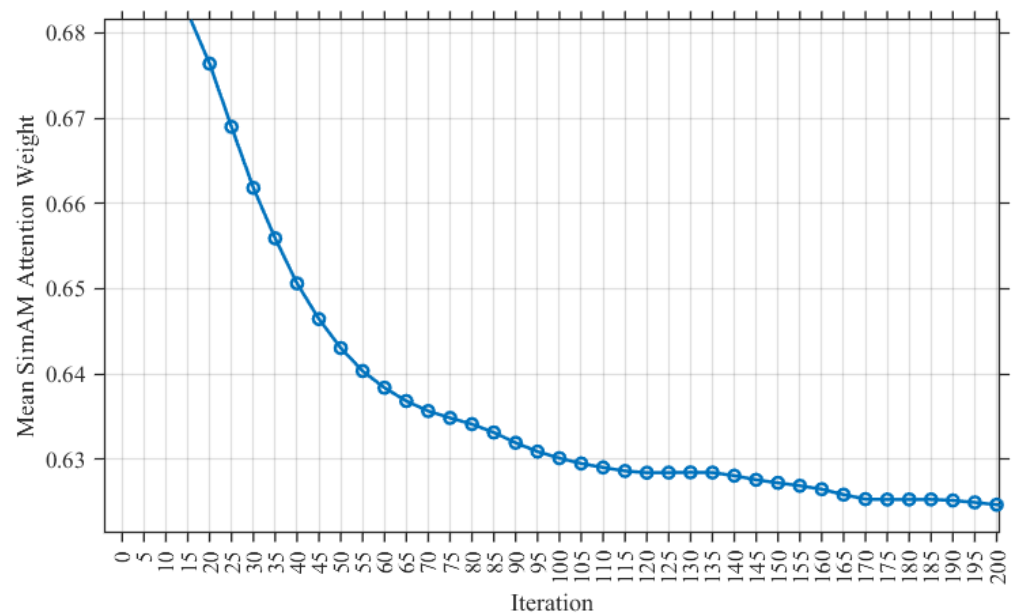


Figure 7. Interpolated Mean SimAM Attention Response over 200 Iterations.

To visually evaluate the fitting and tracking capability of the TCN-BiLSTM-SimAM model, Figure 8 compares the predicted and observed wastewater flow values over 46 consecutive validation samples. In the **left panel**, the blue solid line represents the measured wastewater flow, while the red solid line represents the model predictions. The **right panel** presents the pointwise absolute relative errors, together with the overall L_2 relative error and the mean pointwise relative error, thereby providing both visual and quantitative assessments of the model's prediction performance.

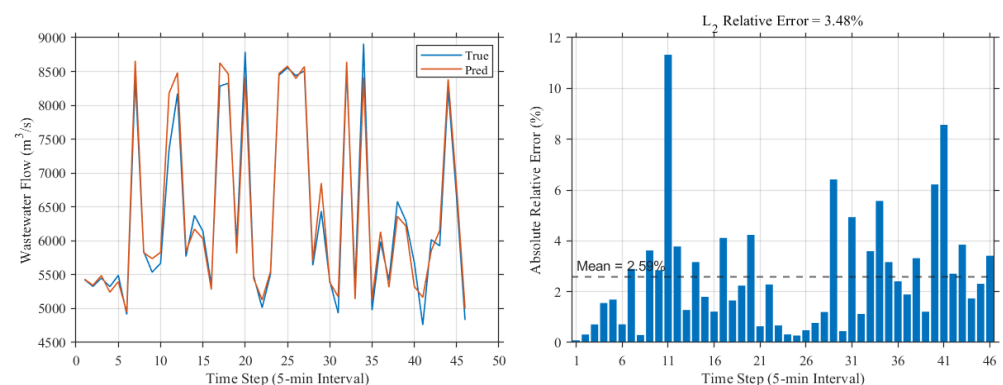


Figure 8. Prediction Results and Relative Error Analysis of the TCN-BiLSTM-SimAM Model.

As shown in the left panel of Figure 8, the predicted wastewater flow generally follows the measured series and reproduces the main peak–valley variation pattern over the validation period. The model performs particularly well in relatively stable intervals and captures most short-term increases and decreases. Nevertheless, noticeable deviations remain near several abrupt-change points, especially around time steps 11, 28, and 40–42, indicating that rapid transitions and sharp local fluctuations are more difficult to predict than stable segments. The right panel of Figure 8 provides a quantitative assessment of these deviations. The overall L_2 relative error is 3.48%, while the mean pointwise absolute relative error is 2.59%. Most pointwise errors remain below 5%, although a few abrupt-change intervals exhibit larger errors, with the maximum reaching approximately 11%. These results demonstrate that the TCN-BiLSTM-SimAM model achieves good overall

agreement with the measured wastewater flow under the tested conditions, while prediction accuracy can still be improved around sudden peaks, troughs, and rapid transition intervals. Such performance may support short-term wastewater-flow monitoring and provide useful information for operational scheduling and equipment-load management.

Figure 2 was calculated for both the training and validation sets, as shown in Table 3. The training set was used to assess the model's fit on seen samples, whereas the validation set was used to evaluate its predictive accuracy and stability on unseen samples. In this study, the model was run 10 times, and the best result on the validation set, together with the corresponding mean and standard deviation, was reported to reflect the stability and robustness of model performance and to provide a reliable basis for subsequent analysis.

Table 3. Performance and Robustness on the Training and Validation Sets.

DATASET	MAE	RMSE	R ²
Training Set-Best	188.5473	249.8586	0.9601
Training Set-Mean	196.0141	265.6495	0.9548
Training Set-Standard Deviation	6.4220	8.6347	0.0029
Validation Set-Best	177.5324	231.1821	0.9697
Validation Set-Mean	195.3396	244.7874	0.9660
Validation Set-Standard Deviation	12.0753	10.2953	0.0029

As shown in Table 3, the model achieved an MAE of 188.5473, an RMSE of 249.8586, and an R^2 of 0.9601 across 10 independent runs on the training set, indicating that it fits the training data well and delivers high predictive accuracy and explanatory power. On the validation set, after 10 independent runs, the best performance metrics were an MAE of 177.5324, an RMSE of 231.1821, and an R^2 of 0.9697. In addition, the mean values over the 10 runs were 196.0141 for MAE, 265.6495 for RMSE, and 0.9548 for R^2 on the training set, while the corresponding standard deviations were 6.4220, 8.6347, and 0.0029, respectively. For the validation set, the mean values over the 10 runs were 195.3396 for MAE, 244.7874 for RMSE, and 0.9660 for R^2 , with corresponding standard deviations of 12.0753, 10.2953, and 0.0029. These results indicate that the model not only achieves excellent predictive performance on unseen samples but also demonstrates good stability and consistency, with relatively small fluctuations in prediction error. It suggests that the model is highly adaptable and robust across different data distributions. Even under conditions of extreme data fluctuations, although prediction accuracy may decline slightly, the overall performance remains high, reflecting the model's reliability and robustness in handling complex coupled time-series forecasting tasks. These findings further demonstrate that the proposed model not only possesses strong fitting capability on the training data but also provides stable, reliable support for wastewater flow forecasting in practical applications, thereby offering a solid data basis for subsequent decision-making and scheduling.

4.3. Ablation Experiments

To further evaluate the contribution of each key module in the proposed TCN-BiLSTM-SimAM model to wastewater flow forecasting performance and to demonstrate the effectiveness of the integrated architecture, module ablation and recombination were conducted from two perspectives: the feature extraction module (none/CNN/TCN) and the attention mechanism (none/SimAM). Accordingly, five models were constructed for comparison, namely LSTM, BiLSTM, CNN-BiLSTM, TCN-BiLSTM, BiLSTM-SimAM and TCN-BiLSTM-SimAM (the proposed model). The experimental results are summarised in Figure 8 and Table 4.

As shown in Figure 9, the prediction curves of all six models generally follow the measured wastewater flow trend over the 46 validation samples, but their performance differs across different types of time-series segments. In relatively stable intervals, such as the low-flow segments around time steps 1–6 and the high-flow plateau around time steps 24–27, most models show good tracking performance because the flow variation is gradual. In contrast, larger deviations occur in peak-flow and abrupt-change segments, such as the sharp peak near time step 7, the peak region around time steps 12–13, and the rapid peak-drop process around time steps 32–35. The LSTM and BiLSTM models tend to produce smoother curves, which makes them more suitable for stable intervals but less effective in capturing sharp peaks and rapid rising or falling processes. They show underestimated peak amplitudes and slight response delays in several abrupt transition segments.

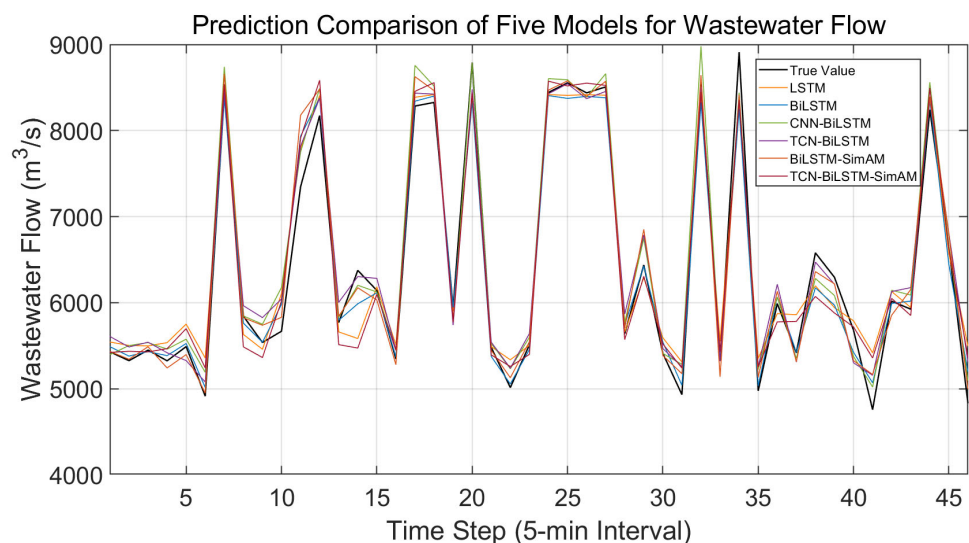


Figure 9. Prediction Curves of Different Models in the Ablation Study.

Compared with BiLSTM, the BiLSTM-SimAM model shows better fitting near some local extrema, indicating that the SimAM attention mechanism helps strengthen informative temporal features. However, because BiLSTM-SimAM lacks convolutional temporal feature extraction, its improvement is still limited in segments with rapid flow changes. The CNN-BiLSTM model improves local fluctuation representation to some extent, but it still shows insufficient sensitivity in sharp peak-flow regions. After introducing the TCN structure, the TCN-BiLSTM model performs better in peak-valley transitions and rising or falling slopes, especially in medium- and high-flow intervals. Further incorporating SimAM, the TCN-BiLSTM-SimAM model provides the closest fit near several peaks, troughs, and turning points, including the abrupt-change regions around time steps 17–20 and 32–35.

These segment-specific errors may also be affected by external factors not included in the present univariate dataset. Do et al. [7] showed that wastewater inflow exhibits seasonal and periodic variations, while Al Saleem et al. [16] reported that weather conditions can alter wastewater-treatment variables. Rainfall may produce rapid wet-weather peaks, and abnormal discharge events may increase prediction errors near abrupt transitions [1]. In addition, pump operating frequency can change hydrograph attenuation in pumping mains [40]; therefore, pump schedules and wet-well storage may affect peak magnitude and timing. Because only single-day flow data from one plant were available, these effects could not be quantitatively separated. Future work will incorporate meteorological, hydraulic, and operational variables using longer-term, multi-season, and multi-plant datasets.

Overall, the results indicate that stable-flow intervals are the easiest to predict for all models, while sharp peak-flow segments and rapid decline intervals are more challenging. The proposed TCN-BiLSTM-SimAM model performs better in peak-flow regions, abrupt transition processes, and medium-to-high flow segments, whereas simpler recurrent models are mainly effective in stable intervals. This suggests that the combined use of TCN and SimAM improves the model's ability to represent local extrema and rapid temporal changes.

Table 4. Performance Comparison of Different Ablation Models.

Model	MAE	RMSE	R^2
LSTM	221.3816	294.4494	0.9508
BiLSTM	203.9696	247.6103	0.9652
CNN-BiLSTM	202.5703	236.0688	0.9616
TCN-BiLSTM	189.1790	236.9650	0.9681
BiLSTM-SimAM	190.4441	237.8141	0.9623
TCN-BiLSTM-SimAM	177.5324	231.1821	0.9697

According to Table 4, all six models achieved R^2 values above 0.95 under the tested conditions. LSTM showed the weakest performance, whereas BiLSTM improved prediction accuracy by using bidirectional temporal information. CNN and TCN further enhanced local and multi-scale temporal feature extraction. The TCN-BiLSTM-SimAM model achieved the best results, with an MAE of 177.5324, RMSE of 231.1821, and R^2 of 0.9697. Compared with TCN-BiLSTM, its MAE and RMSE decreased by 6.16% and 2.44%, respectively.

Table 5 further compares the two models under different Gaussian noise levels. As the noise level increased from 10% to 30%, the performance of both models declined. However, TCN-BiLSTM-SimAM consistently produced lower MAE and RMSE and higher R^2 . At the 30% noise level, it reduced MAE and RMSE by 5.18% and 6.22%, respectively, and increased R^2 from 0.9227 to 0.9315. These results show that SimAM improves prediction accuracy and maintains better performance under the tested noise conditions.

Table 5. Prediction performance of TCN-BiLSTM and TCN-BiLSTM-SimAM under different noise levels.

Noise Level	Model	MAE	RMSE	R^2
10%	TCN-BiLSTM	190.1722	245.5454	0.9564
	TCN-BiLSTM-SimAM	181.5685	244.8411	0.9613
20%	TCN-BiLSTM	202.5277	278.7388	0.9459
	TCN-BiLSTM-SimAM	200.1139	262.744	0.9508
30%	TCN-BiLSTM	263.7482	327.7924	0.9227
	TCN-BiLSTM-SimAM	250.0746	307.3904	0.9315

Table 6 shows that the model is more sensitive to λ than to $N(c)$. The lowest RMSE occurs at $\lambda = 0.7$, while $\lambda = 0.8$ gives the lowest MAE and highest R^2 . In contrast, the performance differences for $N(c) = 24-7$ are relatively small. Therefore, $N(c)$ is fixed according to the input-window structure, whereas λ is included in the subsequent IHHO-based hyperparameter optimisation.

Table 6. Sensitivity Analysis of the SimAM Attention Parameter λ and Local-Region Diversity Index $N(c)$ for Wastewater-Flow Prediction.

Model	Parameter	Value	MAE	RMSE	R^2
TCN-BiLSTM-SimAM	λ	0.5	189.4485	236.5546	0.9625
		0.6	197.2141	246.4936	0.9654
		0.7	188.5531	235.4586	0.9633
		0.8	187.3125	238.2821	0.9663
	$N(c)$	24	195.3396	238.7874	0.9625
		25	194.9963	237.4431	0.9623
		26	195.6412	238.6671	0.9631
		27	196.3647	238.7741	0.9629

4.4. Comparative Experiments

To further evaluate the effectiveness and superiority of the proposed TCN-BiLSTM-SimAM model in short-term wastewater flow forecasting, a comprehensive comparison was conducted against six representative methods, namely CNN-GRU-Multi-Head Attention [41], TCN-Self-Attention-BiLSTM [42], GRU-XGBoost [43], CNN-Transformer-BiLSTM [44], TCN-LSTM-Multi-Head Self-Attention [45], and CNN-BiGRU-Self-Attention [46]. All models were evaluated under identical conditions, including the same dataset, training-validation split strategy, and training parameter settings. The experimental results are summarised in Figure 9 and Table 5.

As shown in Figure 10, all seven prediction models generally reproduced the main fluctuation trend of the wastewater flow series over the 46 validation samples, although their performance varied across different types of time-series segments. In relatively stable intervals, such as the low-flow segments around time steps 1–5 and the high-flow plateau around time steps 24–27, most models achieved acceptable tracking performance. Larger differences were observed in peak-flow regions, rapid-decline intervals, and transition processes, including the sharp peaks around time steps 7, 12, 18–20, 33, and 44, as well as the subsequent falling stages. CNN-GRU-MHA captured the overall trend reasonably well but underestimated several peak values and showed slight response delays during rapid increases. TCN-SA-BiLSTM provided relatively stable predictions in medium- and high-flow intervals, although its estimates near some local extrema were conservative. GRU-XGBoost produced a comparatively smooth curve, which resulted in attenuated peak amplitudes and insufficient representation of several transition processes. CNN-Transformer-BiLSTM showed improved tracking of global variations and high-flow stages, but deviations remained near some sharp peaks and valleys. TCN-LSTM-MHSA captured the rising and falling slopes effectively, although it underestimated several sharp peaks. TCN-BiGRU-SA maintained relatively stable predictions in gradual-flow intervals but showed limited responsiveness in some abrupt-change segments. In comparison, TCN-BiLSTM-SimAM generally provided the closest agreement with the measured curve in peak-flow regions, rapid rising and falling stages, and peak–valley transitions, while also maintaining good performance in stable intervals. Overall, stable-flow intervals were easier for all models to predict, whereas sharp peaks and rapid declines were more challenging. Under the tested conditions, TCN-BiLSTM-SimAM achieved the most balanced performance across these different time-series segments. However, this comparison is based mainly on visual observations, and segment-specific MAE and RMSE values should be further calculated to quantitatively evaluate prediction performance in peak, decline, stable, and transition intervals.

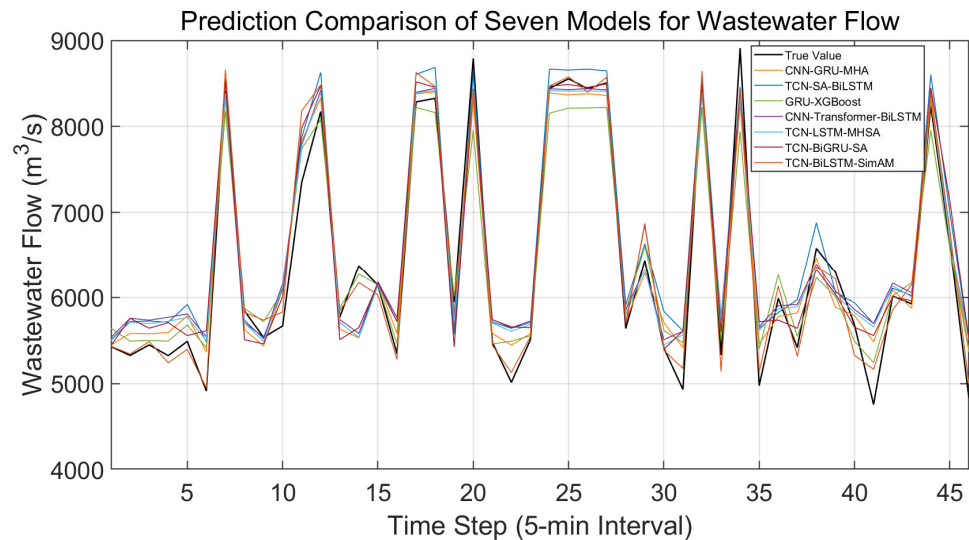


Figure 10. Prediction Curves of Different Models in the Comparative Study.

According to Table 7, clear differences were observed among the seven models. TCN-BiLSTM-SimAM achieved the best overall performance, with the lowest MAE of 177.5324, the lowest RMSE of 231.1821, and the highest R^2 of 0.9697. CNN-Transformer-BiLSTM obtained the second-lowest MAE of 180.8540, indicating good control of average prediction error; however, its RMSE of 263.3720 and R^2 of 0.9210 were notably inferior to those of the proposed model, suggesting weaker control of large deviations and lower overall fitting consistency. Compared with CNN-Transformer-BiLSTM, TCN-BiLSTM-SimAM reduced MAE and RMSE by approximately 1.84% and 12.22%, respectively, while increasing R^2 by 0.0487. GRU-XGBoost ranked second in terms of R^2 , reaching 0.9410, but its MAE and RMSE remained relatively high at 252.3648 and 322.5698. CNN-GRU-Multi-Head Attention and TCN-LSTM-Multi-Head Self-Attention showed moderate performance, whereas TCN-Self-Attention-BiLSTM and CNN-BiGRU-Self-Attention produced larger errors, with the latter obtaining the highest MAE and RMSE and the lowest R^2 . These results indicate that introducing Transformer, self-attention, or multi-head attention alone does not necessarily guarantee improved forecasting performance, because the effectiveness of an attention mechanism depends on its coordination with the temporal feature-extraction structure. The superior performance of TCN-BiLSTM-SimAM can therefore be attributed to the complementary roles of TCN in extracting multi-scale temporal patterns, BiLSTM in modelling contextual dependencies within the historical input window, and SimAM in enhancing informative feature representations. Overall, this combination provides more consistent error control and fitting performance for short-term wastewater flow forecasting under the tested conditions.

Table 7. Performance Comparison of Different Comparative Models.

Model	MAE	RMSE	R^2
CNN-GRU-Multi-Head Attention	269.7032	341.1420	0.9318
TCN-Self-Attention-BiLSTM	331.7151	395.4914	0.9113
GRU-XGBoost	252.3648	322.5698	0.9410
CNN-Transformer-BiLSTM	180.8540	263.3720	0.9210
TCN-LSTM-Multi-Head Self-Attention	283.5558	369.0580	0.9227
CNN-BiGRU-Self-Attention	366.7122	463.0933	0.8749
TCN-BiLSTM-SimAM	177.5324	231.1821	0.9697

4.5. Algorithm Validation

To evaluate the global search capability and stability of the proposed multi-strategy Improved Harris Hawks Optimisation (IHHO) algorithm for continuous optimisation problems, experiments were conducted on the CEC2017 benchmark function suite. Eight representative swarm intelligence algorithms, including SSA, AGA, GWO, WOA, DBO, ISPSO, AVOA, and HHO, were selected as comparison methods.

(1) CEC2017 Benchmark Functions

The CEC2017 benchmark suite contains various types of test functions with distinct characteristics, including unimodal, multimodal, hybrid, and composition functions. It provides a comprehensive platform for evaluating the exploration and exploitation abilities, as well as the resistance to local optima, of optimisation algorithms. In this study, the 10-dimensional CEC2017 benchmark functions were adopted for algorithm comparison, with the search bounds set to $[-100, 100]$, to assess the robustness and scalability of the algorithms in complex search spaces. For fairness, all algorithms were evaluated under the same computational budget and experimental settings, with a population size of 20 and a maximum number of 1000 iterations. Each algorithm was independently executed 30 times to reduce the influence of randomness. The mean and standard deviation of the final best solutions were used as indicators of optimisation accuracy and stability. The key parameter settings of the compared algorithms are summarised in Table 8.

Table 8. Core Parameter Settings of Different Intelligent Optimisation Algorithms.

Algorithm	Parameter Name	Value
SSA	Safety Threshold	0.8
	Producer Ratio	0.2
	Vigilant Ratio	0.1
	Velocity Initialization Coefficient	0.3
AGA	Initial Gravitational Constant	60
	Gravitational Decay Coefficient	30
	Final Elite Ratio	5
	Distance Norm	2
	Distance Power Exponent	1
GWO	Convergence Factor	$2 \rightarrow 0$
	Encirclement Step-Size Coefficient	$[-a, a]$
	Distance Weight Coefficient	$[0, 2]$
	Position Fusion Strategy	$(X_1 + X_2 + X_3)/3$
WOA	Convergence Factor	$2 \rightarrow 0$
	Spiral Auxiliary Factor	$-1 \rightarrow -2$
	Encircling Update Coefficient	$[-a, a]$
	Distance Weight Coefficient	$[0, 2]$
	Spiral Shape Constant	1
	Update Strategy Switching Probability	$[0, 1]$
DBO	Producer Ratio	0.2
	Producer Branch Probability Threshold	0.9
	Differential Perturbation Factor	0.3
	Guided Perturbation Factor	0.1
	Angular Perturbation Range	$[1^\circ, 180^\circ]$
ISPSO	Exploration Control Parameter	0.45
	Perturbation Step-Size Parameter	0.03
	Exploitation Control Parameter	1.6
	Exploration Threshold	0.25

	Temperature Threshold	0.6
AVOA	Exploration-Stage Branch Probability	0.6
	Exploitation Stage I Branch Probability	0.4
	Exploitation Stage II Branch Probability	0.6
	Best-Guidance Weight	0.8
	Second-Best Guidance Weight	0.2
	Nonlinear Control Exponent	2.5
HHO	Random Reset Probability	0.03
	Exploitation Perturbation Intensity Co-efficient	3
	Threshold Parameter	0.8
IHHO	Number of Clusters	3
	Differential Factor	0.5
	Crossover Probability	0.9
	Worst-Individual Replacement Ratio	0.25
	Worst-Individual Perturbation Period	15
	Stagnation Window	25
	Initial DE Mutation Probability	0.6
	Dynamic Opposition Trigger Period	10
	Lévy Perturbation Trigger Probability	0.3

The validation results of the algorithms are summarised in Table 9, which reports eight representative CEC2017 benchmark functions, namely F1, F3, F5, F10, F14, F20, F21, and F28. These functions were selected to present the overall performance trends and relative competitiveness of the compared algorithms in the 10-dimensional benchmark tests, while avoiding an excessively large table in the main text. The complete statistical results for all 30 CEC2017 functions are provided in Table S1.

Table 9. Validation Results of the Optimisation Algorithms.

Test Function	Algorithm	Mean	Standard Deviation	Friedman Value	Friedman Rank
F1	SSA	1.330×10^{10}	2.609×10^9	8.77	9
	AGA	9.711×10^9	2.949×10^9	8.10	8
	GWO	9.885×10^7	3.911×10^8	4.67	7
	WOA	4.426×10^7	9.399×10^7	5.90	6
	DBO	3.414×10^9	3.663×10^9	7.07	3
	ISPSO	1.153×10^5	6.048×10^4	4.50	4
	AVOA	3.044×10^3	2.963×10^3	2.43	5
	HHO	2.670×10^3	3.114×10^3	2.10	2
	IHHO	9.666×10^2	1.187×10^3	1.47	1
F3	SSA	1.118×10^4	1.305×10^3	7.87	8
	AGA	1.657×10^4	4.500×10^3	8.83	9
	GWO	3.526×10^3	2.520×10^3	5.17	7
	WOA	4.878×10^3	4.663×10^3	5.63	3
	DBO	3.987×10^3	2.182×10^3	5.63	4
	ISPSO	5.480×10^3	3.537×10^3	5.87	5
	AVOA	3.343×10^2	3.979×10^3	2.77	6
	HHO	3.000×10^2	6.078×10^{-3}	1.17	1
	IHHO	3.010×10^2	2.947×10^0	2.07	2
F5	SSA	6.117×10^2	1.428×10^1	8.43	9
	AGA	6.150×10^2	2.297×10^1	8.50	3
	GWO	5.192×10^2	7.342×10^0	2.23	6
	WOA	5.528×10^2	2.189×10^1	5.83	8

	DBO	5.530×10^2	1.232×10^1	6.13	7
	ISPSO	5.317×10^2	1.323×10^1	3.77	4
	AVOA	5.404×10^2	1.496×10^1	4.60	5
	HHO	5.356×10^2	1.374×10^1	4.20	1
	IHHO	5.111×10^2	5.018×10^0	1.30	2
F10	SSA	3.095×10^3	1.622×10^2	8.63	9
	AGA	2.938×10^3	3.064×10^2	8.27	3
	GWO	1.699×10^3	3.414×10^2	2.80	6
	WOA	2.183×10^3	3.112×10^2	5.27	7
	DBO	2.279×10^3	2.310×10^2	6.00	8
	ISPSO	1.752×10^3	1.868×10^2	3.10	4
	AVOA	1.982×10^3	3.446×10^2	4.03	5
	HHO	2.001×10^3	2.938×10^2	4.57	2
	IHHO	1.618×10^3	2.638×10^2	2.33	1
F14	SSA	3.937×10^4	8.698×10^4	8.37	9
	AGA	4.827×10^5	1.813×10^6	7.87	8
	GWO	3.838×10^3	1.964×10^3	5.93	6
	WOA	2.817×10^3	1.628×10^3	5.43	5
	DBO	2.388×10^3	2.567×10^3	4.77	7
	ISPSO	1.964×10^3	1.106×10^3	4.17	4
	AVOA	3.081×10^3	2.958×10^3	5.23	3
	HHO	1.470×10^3	2.633×10^1	2.00	2
	IHHO	1.446×10^3	2.580×10^1	1.23	1
F20	SSA	2.268×10^3	2.710×10^1	8.00	9
	AGA	2.329×10^3	9.173×10^1	8.43	8
	GWO	2.108×10^3	4.985×10^1	4.17	3
	WOA	2.208×10^3	8.687×10^1	6.30	6
	DBO	2.127×10^3	6.134×10^1	4.43	4
	ISPSO	2.124×10^3	9.199×10^1	4.33	7
	AVOA	2.135×10^3	5.981×10^1	4.70	4
	HHO	2.101×10^3	5.285×10^1	3.63	1
	IHHO	2.018×10^3	1.172×10^1	1.00	2
F21	SSA	2.392×10^3	4.610×10^1	7.67	5
	AGA	2.421×10^3	4.167×10^1	8.50	8
	GWO	2.317×10^3	7.111×10^1	4.57	7
	WOA	2.339×10^3	4.285×10^1	6.30	9
	DBO	2.226×10^3	2.035×10^1	2.13	3
	ISPSO	2.329×10^3	1.650×10^1	5.63	6
	AVOA	2.268×10^3	6.645×10^1	3.60	4
	HHO	2.253×10^3	6.395×10^1	2.80	1
	IHHO	2.309×10^3	2.105×10^1	3.80	2
F28	SSA	3.907×10^3	7.188×10^1	8.77	9
	AGA	3.830×10^3	7.550×10^{60}	8.13	8
	GWO	3.417×10^3	1.370×10^2	4.67	7
	WOA	3.470×10^3	1.711×10^2	5.37	6
	DBO	3.376×10^3	1.535×10^2	4.37	5
	ISPSO	3.346×10^3	1.244×10^2	3.97	3
	AVOA	3.337×10^3	1.235×10^2	3.37	4
	HHO	3.329×10^3	1.486×10^2	3.27	2
	IHHO	3.320×10^3	1.188×10^2	3.10	1

As shown in Table 9, the nine algorithms exhibit clear performance differences on the eight representative 10-dimensional CEC2017 benchmark functions. SSA, AGA, and WOA generally produced relatively large mean objective values, indicating weaker final optimisation accuracy under the tested settings. GWO, DBO, ISPSO, and AVOA achieved improved results on several functions, although their performance varied across different

problem types. In comparison, IHHO showed the most competitive overall performance. Among the eight functions reported in Table 8, IHHO ranked first on six functions, second on F3, and fourth on F21. In particular, IHHO achieved the lowest mean objective values on F1, F5, F10, F14, F20, and F28, while HHO and DBO performed better on F3 and F21, respectively. These results indicate that IHHO performs consistently well across functions with different optimisation characteristics, although it is not uniformly superior on every test problem. The standard deviations further show that IHHO generally maintains relatively stable results across repeated runs, particularly on F4, F6, F9, F11, F14, F17, F19, F20, F27, and F29, as reported in the complete results in Table S1. Overall, the results support the effectiveness of the proposed IHHO framework in balancing exploration and exploitation under the tested conditions. However, because separate ablation experiments were not conducted for all four improvement strategies, the individual contribution of each strategy cannot be determined from these results alone.

(2) Application Validation

Wastewater flow sequences are influenced by multiple factors, including rainfall, drainage-system conditions, water-consumption patterns, and random disturbances, resulting in pronounced nonlinearity, non-stationarity, and multi-scale fluctuations. To address these characteristics, HHO and IHHO were employed to optimise the hyperparameters of the TCN-BiLSTM-SimAM model, forming the HHO-TCN-BiLSTM-SimAM and IHHO-TCN-BiLSTM-SimAM frameworks, respectively. The optimised hyperparameter vector includes the number of TCN convolutional channels, convolution kernel size, number of BiLSTM hidden units, the SimAM hyperparameter, and the learning rate. Under identical experimental conditions, the proposed IHHO-TCN-BiLSTM-SimAM model was compared with the unoptimised TCN-BiLSTM-SimAM and HHO-TCN-BiLSTM-SimAM models to evaluate its hyperparameter optimisation capability and forecasting performance. Figure 10 and Table 10 present the prediction and optimisation results, while Figure 11 shows the average convergence curves of nine algorithms on four representative CEC2017 benchmark functions. The results further illustrate the convergence accuracy and search efficiency of IHHO under different benchmark-function characteristics.

As shown in Figure 11, the convergence curves of the best RMSE reveal a clear difference between HHO and IHHO when optimizing the hyperparameters of the TCN-BiLSTM-SimAM model. First, in terms of convergence speed, IHHO rapidly approaches a near-optimal solution during the early iterations: its best RMSE decreases quickly from approximately 0.04 to about 0.028 within the first several iterations and then enters a stable stage. In contrast, HHO starts with a much higher initial best RMSE of around 0.09 and needs more iterations to converge to a similar level. Second, from the perspective of convergence behaviour, IHHO exhibits better early-stage search efficiency, while both algorithms gradually become stable in the later iterations. In the final stage, the two methods converge to nearly the same RMSE level, indicating that the main advantage of IHHO lies in accelerating convergence rather than substantially changing the final optimum. These results demonstrate the effectiveness of the proposed improvement strategies in enhancing the convergence efficiency of hyperparameter optimization for the TCN-BiLSTM-SimAM model.

As shown in Figure 12, the proposed IHHO initialization starts with a lower best RMSE than random initialization and decreases more smoothly during the early iterations. It approaches approximately 0.029 within the first 6–8 iterations, whereas random initialization begins at about 0.090 and requires around 16 iterations to reach a comparable level. After approximately 17 iterations, both curves stabilise near 0.028, indicating similar final convergence accuracy. Therefore, under the tested conditions, the proposed initialization mainly improves the initial solution quality and early convergence efficiency, while its advantage in the final objective value is relatively limited.

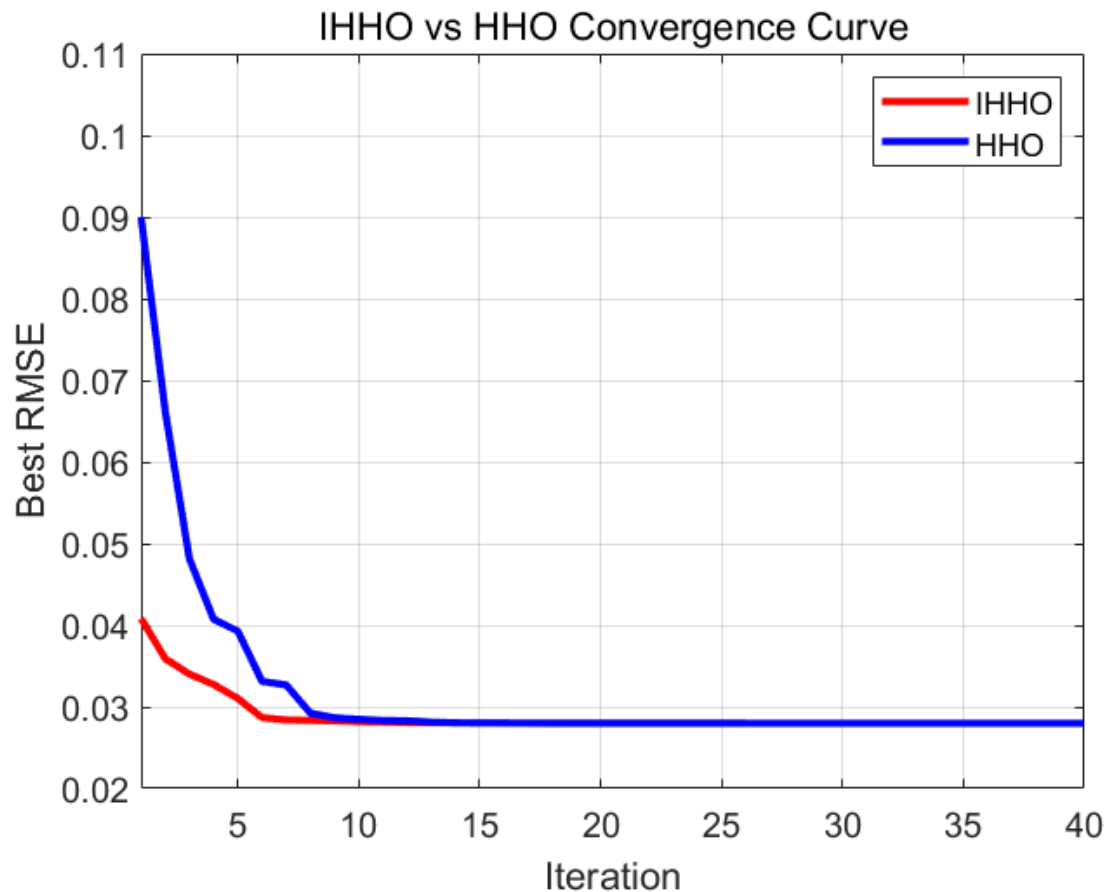


Figure 11. Training Convergence Curves of IHHO and HHO.

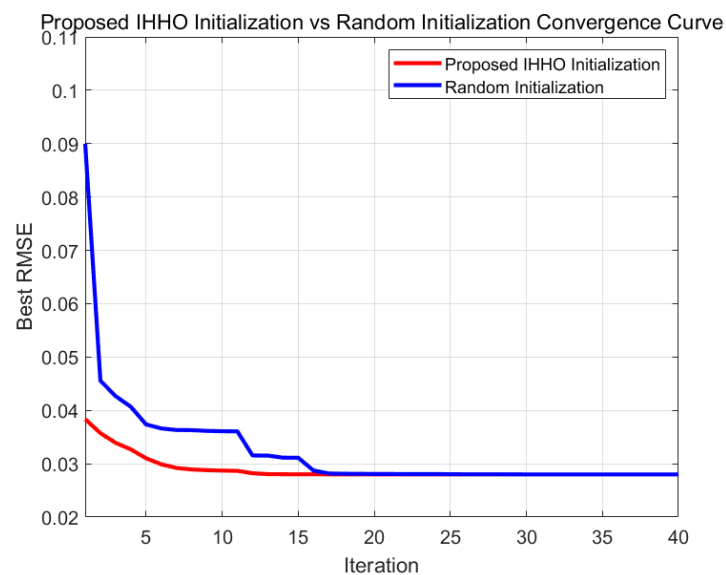


Figure 12. Convergence Curves under Random and Proposed IHHO Initialization Strategies.

Figure 13 presents the average convergence curves of nine algorithms over 30 independent runs on four representative CEC2017 benchmark functions. IHHO achieved the fastest convergence and the lowest final fitness values on all four functions. On F1, IHHO rapidly reduced the fitness value by several orders of magnitude and reached a stable solution earlier than the competing algorithms. Similar behaviour was observed on F10,

F20, and F28, where IHHO exhibited a sharp decline during the early iterations and continued to improve after several comparison algorithms had entered premature convergence plateaus. These results indicate that the proposed improvement strategies enhance both early-stage exploration and late-stage exploitation, enabling IHHO to achieve better convergence accuracy and search efficiency across different benchmark-function characteristics.

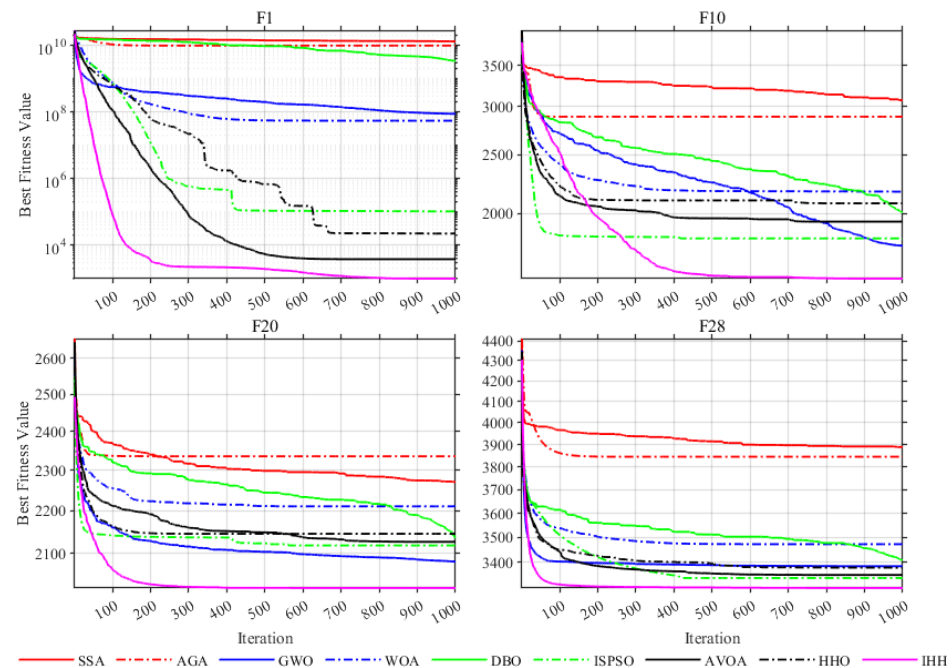


Figure 13. Convergence curves of different algorithms on representative CEC2017 benchmark functions.

Table 10. Performance Comparison of Different Models for Sewage Flow Prediction.

Model	MAE	RMSE	R^2	Training Time
TCN-BiLSTM-SimAM	177.5324	231.1821	0.9697	24 min
HHO-TCN-BiLSTM-SimAM	165.4875	222.4089	0.9719	63 min
IHHO-TCN-BiLSTM-SimAM	161.8777	218.8228	0.9728	71 min

Table 10 compares the validation performance and training time of the baseline TCN-BiLSTM-SimAM model and its HHO- and IHHO-optimised variants. Compared with the baseline model, HHO reduced MAE from 177.5324 to 165.4875 and RMSE from 231.1821 to 222.4089, corresponding to reductions of 6.78% and 3.79%, respectively, while R^2 increased from 0.9697 to 0.9719. IHHO achieved the best overall prediction performance, with an MAE of 161.8777, an RMSE of 218.8228, and an R^2 of 0.9728. Relative to the baseline model, these values represent reductions of 8.82% in MAE and 5.35% in RMSE, together with an increase of 0.0031 in R^2 . Compared with conventional HHO, IHHO further reduced MAE and RMSE by 2.18% and 1.61%, respectively, and increased R^2 by 0.0009. These results indicate that hyperparameter optimisation improves the predictive performance of the hybrid model and that the proposed IHHO provides additional, although relatively moderate, gains over conventional HHO under the tested conditions. However, this improvement is accompanied by increased computational cost: the training time increased from 24 min for the baseline model to 63 min for HHO and 71 min for IHHO.

Therefore, IHHO provides the highest prediction accuracy among the three models, but its accuracy–computation trade-off should be considered in practical deployment.

(3) Comparative Analysis of Algorithms

To further evaluate the overall performance of the proposed model, predictions driven by different intelligent optimization algorithms were compared, including IPSO-CNN-GRU-Multi-Head-Attention [47], IGWO-TCN-Self-

Attention-BiLSTM [48], IBA-GRU-XGBoost [40], ISSA-CNN-Transformer-BiLSTM [49], ISMA-TCN-LSTM-Multi-Head-Self-Attention [50], IWOA-CNN-BiGRU-Self-Attention [51], and IHHO-TCN-BiLSTM-SimAM. The experimental results are presented in Figure 14 and Table 10.

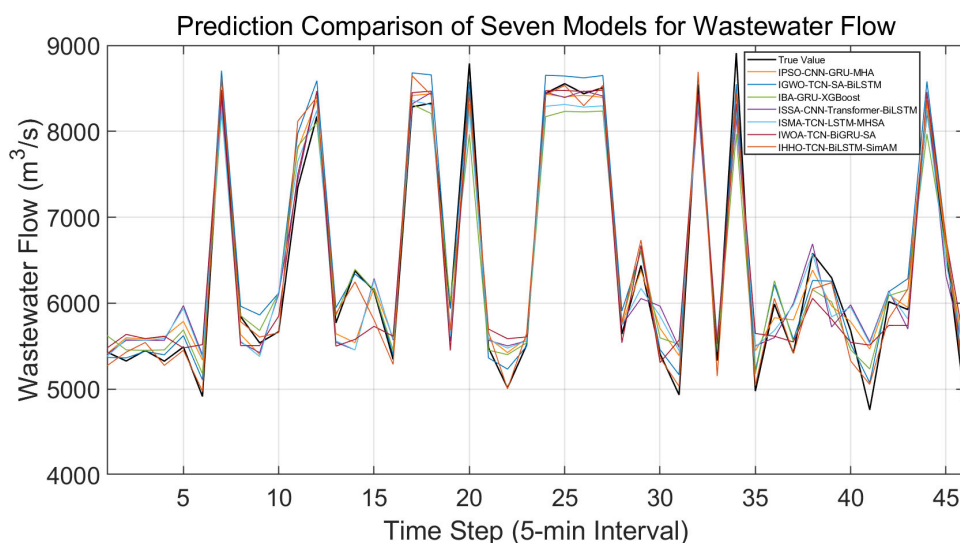


Figure 14. Prediction Curves of Different Models in the Algorithm Comparison.

As shown in Figure 14, all seven models generally reproduced the main variation trend of the wastewater flow series over the 46 validation samples, but their performance differed across specific time-series segments. In relatively stable intervals, such as time steps 1–5 and 24–27, most models maintained acceptable tracking performance. Larger differences appeared in peak-flow regions, rapid-decline intervals, and transition processes, particularly around time steps 7, 11–13, 18–20, 32–34, and 43–45. Several benchmark models underestimated peak amplitudes, produced excessive smoothing during rapid changes, or showed slight deviations near turning points. In comparison, IHHO-TCN-BiLSTM-SimAM generally followed the measured curve more closely in sharp peaks, rapid rising and falling, and peak-valley transition segments, while also maintaining stable predictions in gradual-flow intervals. This improvement can be attributed to the complementary roles of TCN in extracting multi-scale temporal features, BiLSTM in modelling contextual dependencies within the historical input window, SimAM in enhancing informative feature representations, and IHHO in obtaining a more suitable hyperparameter combination. Overall, stable-flow intervals were relatively easy for all models to predict, whereas sharp peaks and rapid declines were more challenging. Under the tested conditions, IHHO-TCN-BiLSTM-SimAM achieved the most balanced visual performance across these different time-series segments.

As shown in Table 11, the integrated models exhibited clear differences in MAE, RMSE, and R^2 , indicating that the combination of optimization algorithms and prediction architectures substantially affects wastewater flow forecasting performance. The proposed IHHO-TCN-BiLSTM-SimAM model achieved the lowest RMSE of 218.8228 and the highest R^2 of 0.9728, while its MAE of 161.8777 was the second lowest among all models.

Although ISSA-CNN-Transformer-BiLSTM obtained a slightly lower MAE of 160.8792, its RMSE and R^2 were 235.7785 and 0.9367, respectively, indicating weaker control of large prediction deviations and lower overall fitting consistency. Compared with IGWO-TCN-Self-Attention-BiLSTM, the proposed model reduced MAE and RMSE by approximately 15.72% and 6.96%, respectively, while increasing R^2 by 0.0042. The remaining models produced higher errors and lower R^2 values, particularly ISMA-TCN-LSTM-Multi-Head Self-Attention and IWOA-CNN-BiGRU-Self-Attention. These results suggest that prediction performance depends not only on the optimization algorithm or attention mechanism used independently, but also on their coordination with the temporal feature-extraction structure. The advantages of IHHO-TCN-BiLSTM-SimAM can be attributed to IHHO-based hyperparameter optimization, TCN-based multi-scale temporal feature extraction, BiLSTM-based contextual dependency modelling within the historical input window, and SimAM-based enhancement of informative features. Overall, the proposed model provides the most balanced error control and fitting performance under the tested conditions.

Table 11. Performance Comparison of Different Integrated Models.

Model	MAE	RMSE	R^2
IPSO-CNN-GRU-Multi-Head Attention	241.8398	310.6189	0.9453
IGWO-TCN-Self-Attention-BiLSTM	192.0761	235.1950	0.9686
IBA-GRU-XGBoost	228.8029	304.3472	0.9475
ISSA-CNN-Transformer-BiLSTM	160.8792	235.7785	0.9367
ISMA-TCN-LSTM-Multi-Head Self-Attention	285.6863	353.5377	0.9291
IWOA-CNN-BiGRU- Self-Attention	279.9261	353.2094	0.9292
IHHO-TCN-BiLSTM-SimAM	161.8777	218.8228	0.9728

5. Conclusions

In this study, a wastewater flow prediction framework based on IHHO-TCN-BiLSTM-SimAM was developed, integrating an intelligent optimisation algorithm with a deep prediction model to address the complex characteristics of wastewater flow sequences, including nonlinearity, multi-scale fluctuations, and peak-valley variations. In terms of model design, the TCN module extracts multi-scale local patterns and expands the receptive field through dilated convolutions and residual connections; the BiLSTM module enhances bidirectional temporal dependency modelling; and the SimAM module adaptively strengthens the response of key features, thereby improving the synergistic characterization of sequence trends, peak-valley fluctuations, and local disturbances. For hyperparameter optimisation, the IHHO algorithm was used to automatically tune the number of TCN convolutional channels and kernel sizes, the number of BiLSTM hidden units, the SimAM hyperparameter λ , and the model training learning rate. IHHO enhances global search capability, suppresses premature convergence, and supports fine-grained local search via stagnation detection, diversity injection, early DE mutation, and dynamic opposition-based perturbation. Experimental results demonstrate that the IHHO-TCN-BiLSTM-SimAM model achieves high accuracy and stability in capturing overall trends, local peaks and valleys, and sudden disturbances in wastewater flow sequences, providing a reliable and robust solution for wastewater flow prediction. The core insight of this study is that accurate short-term wastewater flow forecasting requires the coordinated integration of temporal feature extraction, key feature enhancement, and intelligent hyperparameter optimization, rather than relying on a single model component alone.

When applied to actual wastewater flow data, the TCN-BiLSTM-SimAM model accurately captured the overall sequence trend, local peaks and valleys, and sudden disturbances. The predicted curves over 46 consecutive sample points closely matched the

observed values, with stable peak-valley positions, slope variations, and turning-point responses, resulting in high consistency in the detailed fit. However, the unoptimized model still exhibited certain smoothing effects or local overshoots in some sharp peaks or sudden drop intervals, leading to elevated local errors. After applying IHHO for hyperparameter optimisation, the resulting IHHO-TCN-BiLSTM-SimAM model achieved further improvements in MAE, RMSE, and R^2 (MAE = 161.8777, RMSE = 218.8228, $R^2 = 0.9728$), while exhibiting faster convergence and smoother training, effectively mitigating the influence of high-error samples and enhancing predictive accuracy and stability under complex flow fluctuations. Overall, IHHO optimisation not only improved the hyperparameter configuration but also significantly enhanced the accuracy and robustness of TCN-BiLSTM-SimAM in wastewater flow prediction tasks.

Despite its high prediction accuracy and stability, the IHHO-TCN-BiLSTM-SimAM model still has several limitations. The cascaded structure increases model complexity, and the number of trainable parameters and single-step prediction latency should be further quantified before large-scale engineering deployment. Although the model is suitable for the tested short-term wastewater flow forecasting task, its computational burden may become more significant in multi-station real-time monitoring or edge-device applications. In addition, the current model is based on univariate wastewater flow data, which limits its ability to explain the effects of rainfall, pump-station operation, water quality variation, hydraulic conditions, and abnormal discharge events on flow changes. Moreover, IHHO still relies on iterative population search and may require additional computational time in larger hyperparameter spaces. Future work will focus on lightweight model design, multivariate joint prediction, and engineering deployment. Techniques such as pruning, quantization, knowledge distillation, and simplified attention mechanisms will be explored to reduce computational cost. When available, rainfall, temperature, pH, pump operation status, influent water quality, flow velocity, and other hydraulic or operational variables will be incorporated to improve prediction accuracy and interpretability. Longer-term, multi-season, and multi-plant datasets will also be used to test model generalization. From a practical water-management perspective, the predicted short-term flow can support pump-station scheduling, influent-load balancing, overflow-risk warning, and dynamic adjustment of treatment capacity.

Supplementary Materials: The following supporting information can be downloaded at: <https://www.mdpi.com/article/10.3390/w18151846/s1>. Table S1. Validation results of the optimisation algorithms for all 30 CEC2017 functions.

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

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in this study are included in the article and Supplementary Materials. Further inquiries can be directed to the corresponding author.

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

References

1. Zhou, P.; Li, Z.; Zhang, Y.; Snowling, S.; Barclay, J. Online machine learning for stream wastewater influent flow rate prediction under unprecedented emergencies. *Front. Environ. Sci. Eng.* **2023**, *17*, 152.
2. Zhang, Q.; Li, Z.; Snowling, S.; Siam, A.; El-Dakhakhni, W. Predictive models for wastewater flow forecasting based on time series analysis and artificial neural network. *Water Sci. Technol.* **2019**, *80*, 243–253.
3. Zhou, P.; Li, Z.; Snowling, S.; Baetz, B.W.; Na, D.; Boyd, G. A random forest model for inflow prediction at wastewater treatment plants. *Stoch. Environ. Res. Risk Assess.* **2019**, *33*, 1781–1792.
4. Rezaei, R.; Shabri, A. An innovative hybrid W-EEMD-ARIMA model for drought forecasting using the standardized precipitation index. *Nat. Hazards* **2024**, *120*, 13513–13542.
5. Liu, D.; Wang, T.; Cao, J. Combining autoregressive and non-autoregressive models for ship license plate recognition. *IEEE Trans. Intell. Transp. Syst.* **2025**, *26*, 12719–12724.
6. Wang, N.; Zhuang, W.; Ran, Z.; Wan, P.; Fu, J. Prediction of acute onset of chronic cor pulmonale: Comparative analysis of Holt-Winters exponential smoothing and ARIMA model. *BMC Med. Res. Methodol.* **2024**, *24*, 204.
7. Do, P.; Chow, C.W.K.; Rameezdeen, R.; Gorjian, N. Wastewater inflow time series forecasting at low temporal resolution using SARIMA model: A case study in South Australia. *Environ. Sci. Pollut. Res.* **2022**, *29*, 70984–70999.
8. Giberti, M.; Dereli, R.K.; Bahramian, M.; Flynn, D.; Casey, E. Forecasting wastewater flows and pollutant loads: A comparison of data-driven models within the urban water system framework. *J. Environ. Chem. Eng.* **2024**, *12*, 113478.
9. Ye, J.; Meng, X.; Wang, H.; Zhou, Q.; An, S.; An, T.; Ghorbani Bam, P.; Rosso, D. ARIMA-based forecasting of wastewater flow across short to long time horizons. *Mathematics* **2025**, *13*, 2098.
10. Qi, Y.; Xue, K.; Wang, W.; Cui, X.; Liang, R. Prediction model of borehole spontaneous combustion based on machine learning and its application. *Fire* **2023**, *6*, 357.
11. Martínez-Merino, L.I.; Puerto, J.; Rodríguez-Chía, A.M. Ordered weighted average support vector regression. *Expert Syst. Appl.* **2025**, *274*, 126882.
12. Cao, P.; Wang, K.; Zhang, T.; Yang, L. Extrapolative prediction in multiphase flow pipelines: A multi-fidelity surrogate approach with stacking ensemble. *Eng. Appl. Comput. Fluid Mech.* **2026**, *20*, 2646081.
13. Ashraf, M.W.; Khan, A.; Tu, Y.; Wang, T.; Wang, C. Prediction of the sustainable concrete strength using machine learning-based boosting and expression techniques. *Eur. J. Environ. Civ. Eng.* **2026**, *30*, 2637605.
14. Tanjim, R.; Ahmed, T.; Mostafa, M.; Hannan, N.; Shufian, A.; Pathik, B.B. A CatBoost-based framework for data-driven fault diagnosis in power transformers using dissolved gas analysis. *Results Eng.* **2026**, *30*, 109972.
15. Wang, C.; Zhang, X.; Liu, L.; Zhou, X.; Wu, J. Urban flood risk projection under climate change: An approach based on explainable and optimized ensemble learning. *Geomat. Nat. Hazards Risk* **2026**, *17*, 2638526.
16. Al Saleem, M.; Harrou, F.; Sun, Y. Explainable machine learning methods for predicting water treatment plant features under varying weather conditions. *Results Eng.* **2024**, *21*, 101930.
17. Huang, J.; Yang, S.; Li, J.; Oh, J.; Kang, H. Prediction model of sparse autoencoder-based bidirectional LSTM for wastewater flow rate. *J. Supercomput.* **2023**, *79*, 4412–4435.
18. Yin, H.; Chen, Y.; Zhou, J.; Xie, Y.; Wei, Q.; Xu, Z. A probabilistic deep learning approach to enhance the prediction of wastewater treatment plant effluent quality under shocking load events. *Water Res. X* **2025**, *26*, 100291.
19. Xie, Y.; Chen, Y.; Wei, Q.; Yin, H. A hybrid deep learning approach to improve real-time effluent quality prediction in wastewater treatment plant. *Water Res.* **2024**, *250*, 121092.
20. Chen, Y.; Zhang, H.; You, Y.; Zhang, J.; Tang, L. A hybrid deep learning model based on signal decomposition and dynamic feature selection for forecasting the influent parameters of wastewater treatment plants. *Environ. Res.* **2025**, *266*, 120615.
21. Cao, J.; Xue, A.; Yang, Y.; Lu, R.; Hu, X.; Zhang, L.; Cao, W.; Cao, G.; Geng, X.; Wang, L. A hybrid deep learning framework for predicting industrial wastewater influent quality based on graph optimisation. *J. Water Process Eng.* **2024**, *65*, 105831.
22. Liu, Y.; Cao, A.; Yang, X.; Zhang, Y.; Liu, Y.; Yu, L.; Ju, M. Seismic source location using deep learning and a metaheuristic-optimized grid search algorithm under a sparse network in underground coal mines. *Geomat. Nat. Hazards Risk* **2026**, *17*, 2634969.
23. Nguyen, K.L.; Do, T.T.; Nguyen, G.H.; Ahmad, A. Low-code application and practical implications of common machine learning models for predicting punching shear strength of concrete reinforced slabs. *Adv. Civ. Eng.* **2023**, *2023*, 8853122.
24. Coelho, R.P.C.; Alves, A.F.C.; Pires, T.M.N.; Pires, F.M.A. A composite Bayesian optimisation framework for material and structural design. *Comput. Methods Appl. Mech. Eng.* **2025**, *434*, 117516.
25. Shams, M.Y.; Elshewey, A.M.; El-kenawy, E.-S.M.; Ibrahim, A.; Talaat, F.M.; Tarek, Z. Water quality prediction using machine learning models based on grid search method. *Multimed. Tools Appl.* **2024**, *83*, 35307–35334.

26. Ye, G.; Wan, J.; Deng, Z.; Wang, Y.; Chen, J.; Zhu, B.; Ji, S. Prediction of effluent total nitrogen and energy consumption in wastewater treatment plants: Bayesian optimization machine learning methods. *Bioresour. Technol.* **2024**, *395*, 130361.
27. Ramya, S.; Srinath, S.; Tuppada, P. Enhanced wastewater parameter prediction using comprehensive regression framework: Leveraging feature engineering and hyperparameter optimization with machine learning. *Procedia Comput. Sci.* **2025**, *260*, 822–830.
28. Feng, X.; Tang, L.; An, S.; Yang, K.; Wang, L.; Tang, D.; Wang, Z.; Liu, C. Water quality prediction in wastewater treatment based on data decomposition and dung beetle optimized TCN-BiGRU/BiLSTM. *CIESC J.* **2025**, *76*, 5249–5261.
29. Mei, P.; Li, M.; Zhang, Q.; Peng, J.; Bao, Y.; Zhang, S. Optimization model of process parameters for waterworks based on CNN-ISSA-BiGRU. *J. Hydrol.* **2024**, *633*, 130933.
30. Houssein, E.H.; Mohamed, M.; Younis, E.M.G.; Mohamed, W.M. A hybrid Harris Hawks Optimization with Support Vector Regression for air quality forecasting. *Sci. Rep.* **2025**, *15*, 2275.
31. Wang, H.; Tang, J.; Pan, Q. MSI-HHO: Multi-strategy improved HHO algorithm for global optimization. *Mathematics* **2024**, *12*, 415.
32. Zhang, S.; Zhu, K.; Wang, C. A novel monthly runoff prediction model based on KVMD and KTCN-LSTM-SA. *Water* **2025**, *17*, 460.
33. Lyu, F.; Zhang, X.; Ma, H.; Huang, X. Feature-flow guided modular BiLSTM for multi-parameter degradation representation in aero-engines. *Aerosp. Sci. Technol.* **2026**, *176*, 111832.
34. Wu, H.; Wan, X.; Qian, R.; Ruan, C.; Zhao, J.; Wang, C. Integrating SimAM attention and S-DRU feature reconstruction for Sentinel-2 imagery-based soybean planting area extraction. *Agriculture* **2026**, *16*, 693.
35. Ye, Z.; Song, D.; Xie, H.; Zhang, J.; Zhou, W.; Lei, M.; Zheng, X.; Sun, J.; Zhou, J.; Li, M. LOEV-APO-MLP: Latin hypercube opposition-based elite variation artificial protozoa optimizer for multilayer perceptron training. *Comput. Mater. Contin.* **2025**, *85*, 5509–5530.
36. Ye, J.; Zhang, X.; Hu, C.; Liang, X.; Tang, J.; Wang, Y. Optimising ramp closure strategy of urban expressway based on MBSSA-DE algorithm. *Transp. B Transp. Dyn.* **2026**, *14*, 2624646.
37. Guo, S.-S.; Wang, J.-S.; Ma, X.-X. Improved bat algorithm based on multipopulation strategy of island model for solving global function optimization problem. *Comput. Intell. Neurosci.* **2019**, *2019*, 6068743.
38. Qi, J.-R.; Peng, J.-X.; Ruan, W.-J.; Lei, Z.-J.; Zhou, B.-L. An improved tabu search approach for efficient job-shop scheduling. *PeerJ Comput. Sci.* **2026**, *12*, e3598.
39. Subburaj, B.; Maheswari, J.U.; Ibrahim, S.P.S.; Kavitha, M.S. Population diversity control based differential evolution algorithm using fuzzy system for noisy multi-objective optimization problems. *Sci. Rep.* **2024**, *14*, 17863.
40. David, L.M.; Martins, A.; Rodrigues, M.; Oliveira, A.; Fortunato, A.B.; Silva, O.; Ataíde, A.; Mendes, A. Sanitary sewer overflow discharges: Estimation based on flow rate measurement in pumping mains. *Water* **2024**, *16*, 1916.
41. Chen, J.; Lu, Z.; Li, Z.; Li, W.; Zhou, A.; Pi, W.; Wei, Y.; Yao, S. Dynamic NO_x emission prediction in coal-fired power plants based on joint multi-head attention CNN-GRU hybrid model. *Can. J. Chem. Eng.* **2026**, *104*, 1298–1313.
42. Yang, W.B.; Xia, K.W.; Fan, S.R. Oil logging reservoir recognition based on TCN and SA-BiLSTM deep learning method. *Eng. Appl. Artif. Intell.* **2023**, *121*, 105950.
43. Khayat, A.; Kissaoui, M.; Bahatti, L.; Raihani, A.; Atifi, Y.; Errakkas, K. A novel hybrid GRU-XGBoost model for day-ahead photovoltaic generation forecasting in microgrids. *Sci. Afr.* **2025**, *29*, e02884.
44. Ye, J.; Dai, L.; Wang, H. Enhancing sewage flow prediction using an integrated improved SSA-CNN-Transformer-BiLSTM model. *AIMS Math.* **2024**, *9*, 26916–26950.
45. Peng, K.; Zhou, W.B.; Jiang, L.Z.; Xiong, L.J.; Yan, W.J. Multimodal fusion hybrid neural network approach for multi-class damage classification in high-speed rail track-bridge systems with multi-parameter. *Eng. Struct.* **2025**, *328*, 119710.
46. Yang, Z.; Lv, K.; Yuan, L.; Song, D.; Li, B.; Qian, T. MCCFB: Multi-scale CNN-BiGRU with cross-scale self-attention fusion for fault diagnosis of chemical processes. *Meas. Sci. Technol.* **2026**, *37*, 035105.
47. Cheng, M.; Wang, F. An integrated DDPG-IPSO algorithm for path planning of plant protection UAVs in complex environments. *Inf. Sci.* **2026**, *741*, 123290.
48. Li, X.; Yu, Z.; Liu, M.; Qian, J. Research on prediction of dynamic behaviors of droplet collision and diffusion based on piezoelectric jetting. *J. Braz. Soc. Mech. Sci. Eng.* **2026**, *48*, 294.
49. Fan, C.; Li, G.; Xiao, L.; Yi, L.; Nie, S. Short-term power load forecasting in city based on ISSA-BiTCN-LSTM. *Cogn. Comput.* **2025**, *17*, 39.

50. Yu, T.; Pan, J.; Qian, Q.; Song, M.; Yin, J.; Feng, Y.; Fu, Y.; Li, Y. Improved slime mould algorithm by perfecting bionic-based mechanisms. *Int. J. Bio-Inspir. Comput.* **2023**, *22*, 1–15.
51. Bozorgi, S.M.; Yazdani, S. IWOA: An improved whale optimization algorithm for optimization problems. *J. Comput. Des. Eng.* **2019**, *6*, 243–259.

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.