

## Article

# Modeling and Performance Analysis of a Liquid Desiccant Cooling and Dehumidification System Using ITSO-TCN-BiGRU-SA

Xianhua Ou , Xinkai Wang and Zheyu Wang

College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China;  
18868447091@163.com (X.W.)

\* Correspondence: xhou@zjut.edu.cn

## Abstract

Liquid desiccant dehumidification, an energy-efficient technology for air humidity control, has gained significant attention in recent years. In this study, the air temperature and humidity prediction models of liquid desiccant cooling and dehumidification (LDCD) system are built based on the proposed ITSO-TCN-BiGRU-SA. In the proposed model, the TCN is employed to obtain local features within sequence and improve the learning ability of temporal dependencies; BiGRU strengthens the model through global and bidirectional contextual relationships; self-attention mechanism assigns different weights to each time step. The ITSO algorithm, which combines the nonlinear adaptive weights and Levy flight strategy, is proposed to find the optimal hyperparameters of network. Accordingly, the model prediction accuracy is improved. Through comprehensive comparative analysis with other models under a series of experiment results, the superior performance of the developed models was systematically validated. Furthermore, based on the model predictions and experimental results, a comprehensive analysis was performed to systematically investigate the impact of system inlet parameters on cooling and dehumidification capacity and efficiency, which can provide valuable guidance for system control and operation.

**Keywords:** liquid desiccant; TCN; BiGRU; cooling and dehumidification; performance investigation

## 1. Introduction

Air cooling and dehumidification are essential for maintaining indoor thermal comfort and ensuring high air quality. However, these processes performed by air conditioning systems account for a significant portion of building energy consumption. In Singapore, building air conditioning systems consume approximately 23% of the nation's total energy [1]. In the field of air conditioning research, energy efficiency and system capacity are two key areas of focus. Since the development of the first liquid desiccant dehumidification system (LDDS) in 1955 [2], its high energy efficiency and exceptional dehumidification capabilities have spurred extensive research. Over recent decades, the LDDS system has been studied in-depth in system design [3–5], system modeling [6–8], performance analysis [9–11] and optimization [12–14].

An accurate model serves as the foundation for performance analysis and optimization of the LDDS system. To this end, various models have been established, which can be classified into three main categories, i.e., the grey-box model, physical-based model and



Academic Editor: Chao Liu

Received: 16 June 2026

Revised: 13 July 2026

Accepted: 15 July 2026

Published: 17 July 2026

**Copyright:** © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

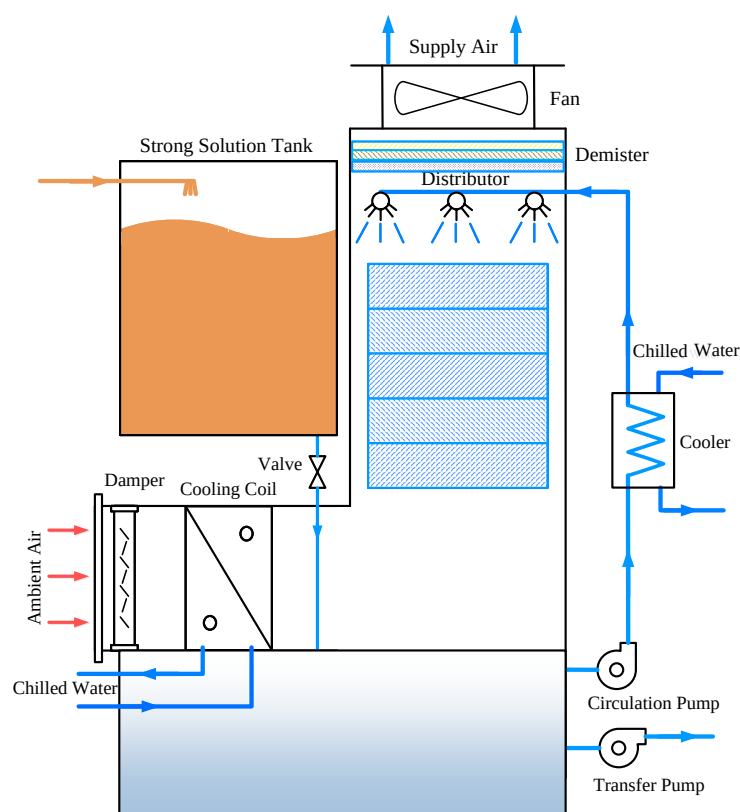
This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

data-driven model. Among these, physical-based models are derived from fundamental thermodynamic principles and physical theories, integrating detailed information such as the structural configurations of system components and thermophysical properties of working fluids. Liu et al. [15] developed a state-space model for regenerator, incorporating thermodynamic properties of both the desiccant solution and packing materials. Chen et al. [16] introduced a finite-difference model for packed-type liquid desiccant equipment, effectively capturing the coupled heat and mass transfer processes, with model predictions demonstrating strong agreement with experimental data. In [17], a dehumidification performance model for an internally cooled dehumidifier was established, incorporating an analysis of the influence of the thickness and shrinkage of liquid falling film. Besides this, a physics-based modeling approach has been adopted by studies [18,19]. The grey-box model is derived from physical or thermodynamic principles, while the model parameter is determined through the parameter identification method with experimental measurements. Shen et al. [20] introduced a novel desiccant regeneration system and established a heat and mass transfer model to describe the liquid desiccant regeneration process. The numerical predictions demonstrated good agreement with experiments, maintaining a maximum relative error margin of 10.03%. Additionally, the system regeneration performance under different working conditions was analyzed and compared. Wang et al. [21] proposed a hybrid model for a batch-wise LDAC system. Meanwhile, the system's dehumidification performance was also investigated. In [22], a dynamic model for a regeneration system operating under vacuum conditions was established on the basis of mass and enthalpy balance principles. The model predictions aligned well with experimental measurements. Furthermore, the transient regeneration performance of the system was investigated under step changes in the inlet temperatures of chilled and hot water. Similarly, Wu et al. [23,24] and Li et al. [25] adopted comparable modeling approaches in their respective studies. Physical models are built upon the principles of thermodynamics and mass transfer, offering good interpretability. However, their construction often relies on internal system parameters that are difficult to accurately obtain in complex boundary conditions, and they incur high computational costs, making them less suitable for real-time applications. Gray-box models simplify the structure by identifying some parameters through experimental data, but their predefined model frameworks may face limitations in generalization when dealing with the highly nonlinear and strongly coupled dynamic characteristics of the system. In contrast, data-driven models, especially those based on deep learning, demonstrate unique practical advantages: they do not require predefined complex mechanistic equations and can autonomously learn the intricate nonlinear mapping and dynamic dependencies between multidimensional input variables and target outputs directly from historical time-series operational data. This approach avoids the cumbersome process of mechanistic derivation and parameter acquisition, resulting in higher development efficiency. Moreover, a well-trained data-driven model is capable of performing rapid and accurate forward predictions, which provides a feasible technical pathway for its integration into real-time optimization frameworks.

As an effective modeling approach, data-driven modeling has been extensively studied and applied in various system modeling applications. These models are constructed using experimental datasets obtained under a range of working conditions, enabling accurate prediction of system output variables from corresponding input variables. In recent years, neural network-based modeling methods have attracted considerable attention in the field of building air conditioning systems. Gandhidasan and Mohandes [26] proposed an artificial neural network (ANN) model of a dehumidifier, and the system's performance was investigated. Mohammad et al. [27] built a dehumidifier model by using the ANN approach; the dehumidification effectiveness and water condensation rate were then pre-

dicted. The relative errors between model predictions and experimental results were within 10%. In [28], a long short-term memory (LSTM) network was employed to predict indoor air temperature in a multi-loop central air conditioning system. The results revealed that the LSTM model outperformed the recurrent neural network (RNN) model in terms of predictive performance. To further enhance prediction accuracy, hybrid models integrating multiple deep learning modules have gained increasing attention. Elmaz et al. [29] developed a CNN-LSTM hybrid model for indoor air temperature prediction. In this architecture, the convolutional neural network (CNN) serves as a local feature extractor, with extracted features subsequently fed into the LSTM pipeline for multi-step temperature forecasting. Verification results indicated that the proposed model maintained stable performance across prediction periods of up to 2 h ( $R^2 > 0.9$ ). The above model also has some limitations. The hybrid model contains multiple adjustable hyperparameters, which will have a direct impact on the prediction results. How to find the most suitable hyperparameters for the model is a key issue. Therefore, integrating optimization algorithms into hybrid models is also the current mainstream trend. Wu et al. [30] combined the LSTM with the Genetic Algorithm (GA) to optimize the ventilation control system of residential buildings. They used the GA to automatically search for the optimal hidden layer structure and spatial dropout rate of the LSTM. This optimization method significantly improved the model's prediction accuracy. Guo et al. [31] considered the feasibility of optimizing the MLP model with four metaheuristic algorithms, including salp swarm optimization, whale optimization, spotted hyena optimization and wind-driven optimization. The results demonstrated significant improvements in prediction accuracy, with the heating load prediction accuracy increasing from 83.85% to 92.48% and the cooling load prediction accuracy rising from 60.61% to 89.29%. Furthermore, a data-driven modeling approach was also adopted in studies [32–34].

In LDDS, the regeneration of the desiccant is a major contributor to overall energy consumption [35]. To address this issue, a two-stage liquid desiccant cooling and dehumidification (LDCD) system, as illustrated in Figure 1, is constructed. The system aims to reduce the dilution rate of the desiccant solution in the dehumidifier, thereby decreasing the frequency and energy consumption associated with solution regeneration. Although the LDCD system demonstrates higher energy efficiency compared to conventional LDDS, its energy consumption can be further minimized through optimized operational strategies. To enable precise energy optimization of the LDCD system, an accurate system model must first be developed. The LDCD system involves complex two-stage heat and mass transfer processes, which require a robust modeling approach. In this study, we propose a novel hybrid prediction model integrating the temporal convolutional network (TCN), bidirectional gated recurrent unit (BiGRU), self-attention mechanism (SA), and improved tuna swarm optimization (ITSO) to accurately predict the supply air temperature and humidity. The TCN layer captures temporal dependencies and local patterns within the input sequence via convolutional operations. The BiGRU layer, connected to the TCN layer, operates at a higher level to model bidirectional relationships and long-range dependencies. The SA mechanism dynamically identifies critical features in the sequence and assigns adaptive weights to each time step. Meanwhile, the ITSO algorithm optimizes the model by identifying the most suitable hyperparameters. Finally, the proposed ITSO-TCN-BiGRU-SA model is rigorously evaluated and compared with other models to demonstrate its effectiveness. Additionally, based on the model predictions, the influence of system inlet parameters on cooling and dehumidification performance is analyzed. The findings of this study provide valuable insights for optimizing the performance of the LDCD system and guiding its operational control.



**Figure 1.** Schematic diagram of the LDCD system.

## 2. Methodology

As illustrated in Figure 1, the LDCD system comprises two primary components, namely a dehumidifier and a cooling coil, where the mass and heat transfer process occurs. In this system, outdoor air is drawn by fan into the cooling coil and dehumidifier. At the same time, chilled water is circulated by a water pump into the cooling coil and cooler, while the desiccant solution is pumped by a solution pump through the cooler and dehumidifier. In the cooling coil, air flows through the gaps between fins and undergoes heat exchange with the chilled water. As a result, the air is cooled and the humidity gradually reaches saturation. As the air temperature continues to drop, the moisture in the air is condensed and precipitated, achieving the first stage of cooling and dehumidification. The chilled water is heated and then sent back to the chiller system. The desiccant solution also undergoes a heat exchange reaction with chilled water in the cooler; its temperature is reduced to a specified value. After passing through cooling coil, the pre-cooled air enters the dehumidifier and contacts the flowing cooled desiccant solution on the filter surface, initiating simultaneous heat and mass transfer process. During this interaction, both heat and moisture are transferred from air to desiccant solution. The air undergoes further cooling and dehumidification, ultimately reaching the specified temperature and humidity level. Meanwhile, the desiccant solution becomes diluted as it absorbs moisture. When its concentration drops below a predefined threshold, the desiccant solution will be transferred to the regeneration system for concentration, restoring its dehumidification capacity.

The cooling capacity, moisture removal rate, thermal efficiency and moisture effectiveness are commonly employed as performance indicators for evaluating the LDDS system in various studies [36–38] and are defined as follows:

$$\begin{aligned} Q_{cd} &= \dot{m}_a (H_{a,i} - H_{a,o}) \\ N_{cd} &= \dot{m}_a (\omega_{a,i} - \omega_{a,o}) \\ \eta_{h,cd} &= \frac{T_{a,i} - T_{a,o}}{T_{a,i} - T_{s,i}} \\ \eta_{m,cd} &= \frac{\omega_{a,i} - \omega_{a,o}}{\omega_{a,i} - \omega_{cd,equ}} \end{aligned} \quad (1)$$

where  $\dot{m}_a$  represents air mass flow rate;  $T_{a,i}$ ,  $T_{a,o}$ ,  $H_{a,i}$ ,  $H_{a,o}$ ,  $\omega_{a,i}$  and  $\omega_{a,o}$  denote air inlet and outlet temperature, enthalpy and humidity, respectively.  $T_{s,i}$  is desiccant solution inlet temperature, and  $\omega_{cd,equ}$  is desiccant solution equilibrium humidity ratio.

### 3. ITSO-TCN-BiGRU-SA Prediction Model

#### 3.1. Temporal Convolutional Networks

The temporal convolutional network (TCN) is an innovative variant of CNN architecture that extends traditional one-dimensional CNN by incorporating three key components: dilated convolution, residual connections and causal convolution. In contrast to recurrent neural networks, TCN demonstrates superior capability in dealing with the issue of gradient vanishing and explosion.

##### 3.1.1. Causal Convolution

Causal convolution, a temporal modeling approach that enforces strict time constraints, can be mathematically expressed as follows:

$$P(x_t) = \prod_{i=1}^T P(x_t | x_1, x_2, \dots, x_{t-1}) \quad (2)$$

where  $T$  is total time, and  $P(\cdot)$  is predicted probability.

The output sequence generated by causal convolution maintains identical length to its input sequence, while ensuring temporal causality by exclusively depending on current and preceding inputs, thereby effectively eliminating future information leakage. The schematic representation of this mechanism is illustrated in Figure 2.

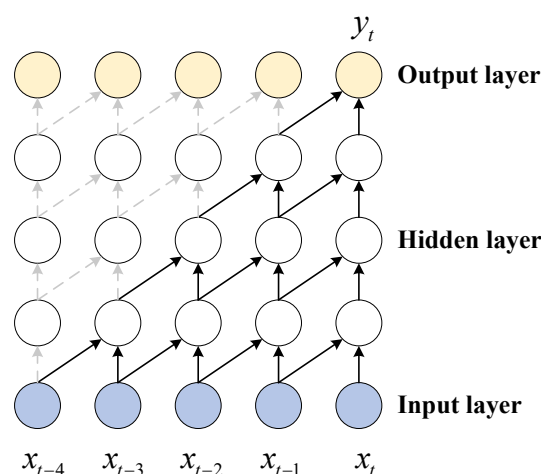


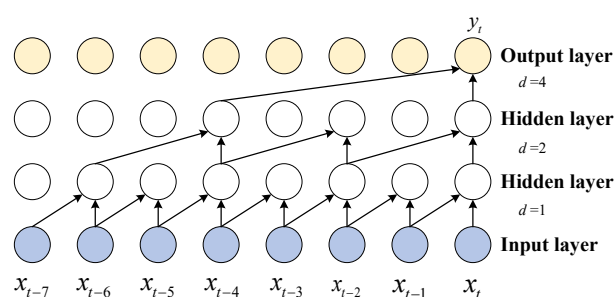
Figure 2. Schematic diagram of causal convolution.

### 3.1.2. Dilated Convolution

In contrast to standard convolution operations, dilated convolution significantly enhances the network's receptive field efficiency by exponentially increasing the dilation rate across successive layers, thereby enabling TCN to achieve an extensive perceptual domain with minimal network depth. As illustrated in Figure 3, the expansion coefficient of exponential growth in the dilated convolution is  $d_i = 2^i$  ( $i$  is network layer number).

$$F(s) = (x *_d f)(s) = \sum_{i=0}^{k-1} f(i) \cdot x_{s-d \cdot i} \quad (3)$$

where  $k$  is convolution kernel size, and  $d$  is expansion coefficient.



**Figure 3.** Schematic diagram of dilated convolution.

### 3.1.3. Residual Module

A residual module is incorporated into the TCN model to mitigate information instability and loss issues caused by excessive network depth. The fundamental concept involves the integration of one or more layers with “skip connection” operation. The residual module comprises two dilated causal convolutional layers coupled with a nonlinear mapping and incorporates both weight normalization and Dropout techniques for network regularization. The architecture is illustrated in Figure 4. To address potential dimensionality mismatches between the input and output, an additional one-dimensional convolution is employed, ensuring that the tensors involved in the summation operation maintain consistent dimensions. The output of residual module is expressed as

$$o = \text{Activation}(x + F(x)) \quad (4)$$

$$F(s) = (x *_d f)(s) = \sum_{i=0}^{k-1} f(i) \cdot x_{s-d \cdot i} \quad (5)$$

### 3.2. Bidirectional Gated Recurrent Unit

The Gated Recurrent Unit (GRU) is a variant of RNN, derived as a simplification of the LSTM architecture. It was designed to address challenges such as vanishing gradients in long-term memory and backpropagation. Compared to LSTM, GRU offers a more compact architecture with significantly fewer parameters, while achieving comparable predictive accuracy. The mathematical formulation of GRU is as follows:

$$\begin{aligned} z_t &= \sigma(W^Z x_t + U^Z h_{t-1}) \\ r_t &= \sigma(W^r x_t + U^r h_{t-1}) \\ \tilde{h}_t &= \tanh(r \odot U h_{t-1} + W x_t) \\ h_t &= (1 - z_t) \odot \tilde{h}_t + z_t \odot h_{t-1} \end{aligned} \quad (6)$$

where  $h_t$  and  $r_t$  represent the state of hidden unit and reset gate, respectively;  $\sigma(\cdot)$  stands for the sigmoid function and acts as an activation function.  $W^Z$  and  $U^Z$  are the weights of update gate,  $W^r$  and  $U^r$  are the weights of reset gate and of update gate.

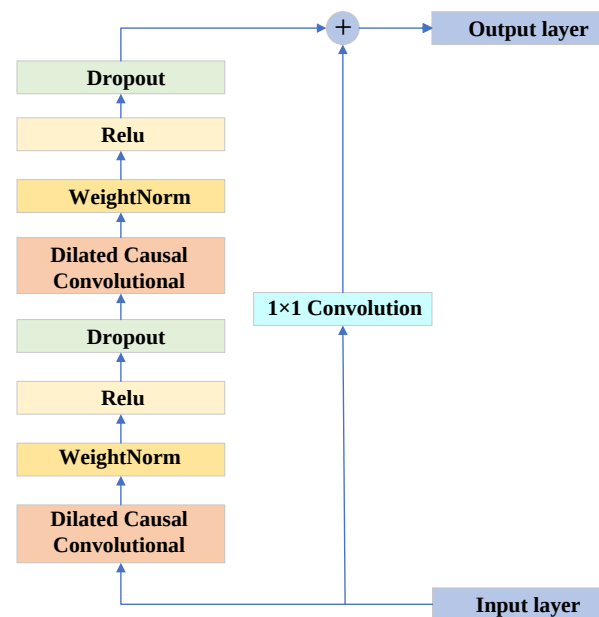


Figure 4. Schematic diagram of residual module.

Bidirectional Gated Recurrent Unit (BiGRU) is an advanced variant of GRU architecture. In this network, the output layer integrates information from both backward and forward propagation. Each training sequence is processed by two distinct hidden layers, one operating in a backward direction and the other in a forward direction, and their outputs subsequently combined at the same output layer. This bidirectional mechanism enables the output layer to capture comprehensive contextual information, incorporating both past and future states of each point in the input sequence. Compared to the unidirectional GRU model, BiGRU exhibits faster convergence, greater resistance to overfitting, and improved accuracy, as it leverages the dual weights of backward and forward states. The architecture of BiGRU is illustrated in Figure 5.

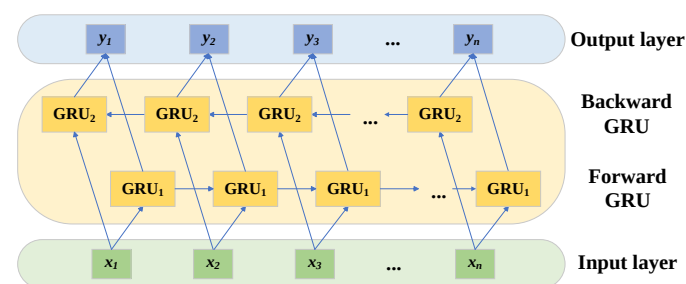


Figure 5. Schematic diagram of BiGRU.

### 3.3. Self-Attention Mechanism Module

The self-attention mechanism (SA) [39] is an intuitive method of explanation that mimics the mechanisms of human vision. It is extensively utilized in a variety of deep learning applications, such as image analysis, natural language processing and load prediction. It has been found that the relevant time series information can be effectively preserved by

adding attention mechanisms and weight allocation principles into the network model. The formula of SA is given as follows:

$$Attention(Q, K, V) = Softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (7)$$

where  $V$ ,  $K$  and  $Q$  are the value matrix, key matrix and query matrix, respectively. The effect of each query element on all key elements in the SA is computed by taking the dot product of  $K$  and  $Q$ . Then, the result is divided by  $\sqrt{d_k}$ . Finally, the attention weight of each position is obtained through the *Softmax* function, and the weighted sum is output.

### 3.4. Improved TSO Algorithm

#### 3.4.1. TSO Algorithm

The Tuna Swarm Optimization (TSO) algorithm proposed by Xie et al. [40] is a swarm intelligence optimization algorithm, inspired by the foraging behaviors of tuna schools, specifically parabolic foraging and spiral foraging. During the population initialization phase, the TSO algorithm generates an initial population randomly within the search space. Assuming that there are  $NP$  tunas in the tuna stocks, the mathematical formulation for initializing each tuna individual in algorithm is expressed as

$$X_i^{\text{int}} = rand \cdot (ub - lb) + lb, i = 1, 2, \dots, NP \quad (8)$$

where  $X_i^{\text{int}}$  is the initial tuna, with each tuna corresponding to a candidate solution in the TSO algorithm. *rand* represents a uniformly distributed random vector with values ranging from 0 to 1. *lb* and *ub* denote the lower and upper boundaries of the search space.

Two distinct foraging strategies are employed by tuna swarms: parabolic foraging and spiral foraging. The selection probability for each strategy is set at 50%. The spiral foraging strategy's mathematical formulation is expressed as

$$\begin{aligned} \alpha_1 &= a + (1 - a) \cdot \frac{t}{t_{\max}} \\ \alpha_2 &= (1 - a) - (1 - a) \cdot \frac{t}{t_{\max}} \\ \beta &= e^{bl} \cdot \cos(2\pi b) \\ l &= e^{3 \cos((t_{\max} + 1/t) - 1)\pi} \end{aligned} \quad (9)$$

$$X_i^{t+1} = \begin{cases} \alpha_1 \cdot (X_{rand}^t + \beta \cdot |X_{rand}^t - X_i^t|) \\ + \alpha_2 \cdot X_i^t, i = 1, \\ \alpha_1 \cdot (X_{rand}^t + \beta \cdot |X_{rand}^t - X_i^t|) \\ + \alpha_2 \cdot X_{i-1}^t, i = 2, 3, \dots, NP, \\ \alpha_1 \cdot (X_{best}^t + \beta \cdot |X_{best}^t - X_i^t|) \\ + \alpha_2 \cdot X_i^t, i = 1, \\ \alpha_1 \cdot (X_{best}^t + \beta \cdot |X_{best}^t - X_i^t|) \\ + \alpha_2 \cdot X_{i-1}^t, i = 2, 3, \dots, NP, \end{cases} \quad \begin{matrix} , if \text{ rand} < \frac{t}{t_{\max}} \\ \\ \\ , if \text{ rand} \geq \frac{t}{t_{\max}} \end{matrix} \quad (10)$$

where,  $a$  is a constant that governs the degree to which tuna follow the optimal individual and previous individual during the initial stage;  $b$  is a uniformly distributed random number within range  $[0, 1]$ .  $X_{best}^t$  denotes the current optimal individual, while  $\alpha_1$  denotes a trend weight coefficient that regulates the tendency of tuna individuals to either swim toward the optimal individual or randomly select adjacent individuals. Additionally,  $\alpha_2$  is a trend weight coefficient that controls the movement of tuna individuals toward the front.

$\beta$  serves as a distance parameter, determining the distance between tuna individuals and either the best individual or a randomly chosen reference individual.

The mathematical formulation of parabolic foraging strategy is expressed as

$$X_i^{t+1} = \begin{cases} X_{best}^t + rand \cdot (X_{best}^t - X_i^t) \\ + TF \cdot p^2 \cdot (X_{best}^t - X_i^t), & \text{if } rand < 0.5 \\ TF \cdot p^2 \cdot X_i^t, & \text{if } rand \geq 0.5 \end{cases} \quad (11)$$

$$p = \left(1 - \frac{t}{t_{max}}\right)^{(t/t_{max})} \quad (12)$$

where  $TF$  is a random variable that takes a value of either 1 or  $-1$ .

### 3.4.2. ITSO Algorithm

In the initialization stage of tuna algorithm, the initial position of tunas is generated randomly, which often leads to the initial tunas clustering in similar or identical positions. This results in minimal diversity among individuals and insufficient coverage of the search space, significantly impairing the algorithm's global search capability. To address this issue, this paper introduces an improved circle chaotic mapping to replace the random initialization of the population in the original algorithm. The proposed approach is defined by the following formula:

$$N_{i+1} = \text{mod}(3.8N_i + 0.4 - (0.7/3.8\pi) \sin(3.8\pi N_i), 1) \quad (13)$$

where  $N_i$  denotes chaotic particles.

The improved circle chaotic mapping operator is integrated into the TSO algorithm to ensure a more uniform distribution of initial tuna individuals within the search space. This enhancement significantly increases the population diversity of TSO, thereby facilitating the generation of more evenly distributed candidate solutions.

Weight parameters play a critical role in swarm intelligence optimization algorithms. Selecting appropriate weights can effectively balance the algorithm's exploration and convergence capabilities. In the TSO algorithm, the weight parameters  $\alpha_1$ ,  $\alpha_2$  and  $p$  determine the degree to which a tuna individual follows the optimal individual. When the weight parameters are larger, the tunas exhibit a stronger tendency to follow the optimal individual, enabling the entire population to explore the search space more thoroughly. Conversely, when the weight parameters are smaller, the tunas follow the optimal individual to a lesser degree, causing them to swim within a confined area, which helps establish a localized search region around each individual. However, the optimization process of TSO is highly complex, and the linearly varying weight parameters  $\alpha_1$ ,  $\alpha_2$  and  $p$  fail to accurately reflect the actual dynamics of the algorithm's optimization process. To address the disadvantages of linear weight variation, nonlinear adaptive weight strategies have been introduced to enhance swarm intelligence optimization algorithms [41]. Experimental results demonstrate that the nonlinear adaptive weight strategy outperforms the linear weight strategy in terms of optimization performance. Consequently, the weight coefficients  $\alpha_1$ ,  $\alpha_2$  and  $p$  are redefined as  $\alpha_{1i}$ ,  $\alpha_{2i}$  and  $p_i$ , respectively, and are expressed as follows:

$$\begin{aligned} \alpha_{1i}(t) &= \alpha_{1ini} - (\alpha_{1ini} - \alpha_{1fin}) \cdot \sin\left(\frac{t}{\mu \cdot t_{max}} \cdot \pi\right) \\ \alpha_{2i}(t) &= \alpha_{2ini} - (\alpha_{2ini} - \alpha_{2fin}) \cdot \sin\left(\frac{t}{\mu \cdot t_{max}} \cdot \pi\right) \\ p_i(t) &= p_{ini} - (p_{ini} - p_{fin}) \cdot \sin\left(\frac{t}{\mu \cdot t_{max}} \cdot \pi\right) \end{aligned} \quad (14)$$

where  $\mu = 2$ ,  $\alpha_{ini}$  and  $\alpha_{fin}$  represent the initial and final value of  $\alpha$ , respectively. The improved weight parameters  $\alpha_{1i}$ ,  $\alpha_{2i}$  and  $p_i$  exhibit rapid changes in the early stages, enabling tuna individuals to closely follow the optimal individuals and thereby enhancing the global exploration capability of TSO. In the later stages, the weight parameters change at a slower rate, allowing tuna individuals to more actively explore their surrounding areas, which improves the local search ability of TSO.

For swarm intelligence optimization algorithms, avoiding local optima during the optimization process is a significant challenge. The Levy flight strategy is a powerful approach to enhance the global exploration capability of intelligent optimization algorithms. This strategy incorporates a movement mechanism that combines both short and long steps. By employing long steps at low frequencies, the algorithm can conduct extensive searches across the entire search space, while using short steps at high frequencies enables intensive local exploration within the immediate vicinity. To leverage these advantages, the Levy operator is integrated into the TSO algorithm to refine its population update strategy.

Levy's flight mechanism is

$$levy(s) = 0.01 \times \frac{u}{|v|^{1/\zeta}}, u \sim N(0, \sigma^2), v \sim N(0, 1) \quad (15)$$

where  $\zeta = 1.5$ , and the variance  $\sigma$  is shown in Equation (16).

$$\sigma = \left( \frac{\Gamma(1 + \zeta) \sin\left(\frac{\pi\zeta}{2}\right)}{\zeta \cdot \Gamma\left(\frac{1+\zeta}{2}\right) \cdot 2^{\frac{1-\zeta}{2}}}\right)^{1/\zeta} \quad (16)$$

$$\Gamma(x) = \int_0^{+\infty} e^{-t} t^{x-1} dt \quad (17)$$

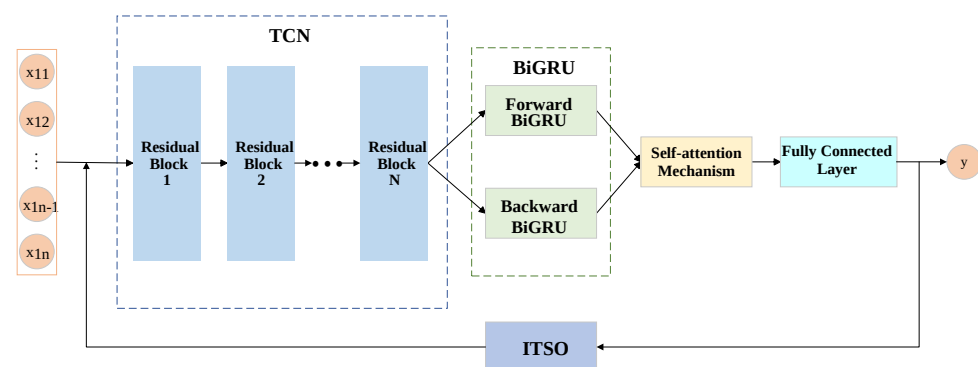
The arithmetic expression of the enhanced spiral foraging strategy, incorporating nonlinear adaptive weighting mechanism and Levy operator, is presented as follows:

$$X_i^{t+1} = \begin{cases} \alpha_{1i}(X_{rand}^t + Levy(s)|X_{rand}^t - X_i^t| + \alpha_{2i}X_i^t), i = 1, \\ \alpha_{1i}(X_{rand}^t + Levy(s)|X_{rand}^t - X_i^t| + \alpha_{2i}X_{i-1}^t), i = 2, \dots, NP & \text{if } rand < \frac{t}{t_{max}} \\ \alpha_{1i}(X_{best}^t + \beta|X_{best}^t - X_i^t| + \alpha_{2i}X_i^t), i = 1, \\ \alpha_{1i}(X_{best}^t + \beta|X_{best}^t - X_i^t| + \alpha_{2i}X_{i-1}^t), i = 2, \dots, NP & \text{if } rand \geq \frac{t}{t_{max}} \end{cases} \quad (18)$$

### 3.5. ITSO-TCN-BiGRU-SA Model

The structure diagram of the proposed ITSO-TCN-BiGRU-SA prediction model is illustrated in Figure 6. The proposed ITSO-TCN-BiGRU-SA model architecture is a targeted design solution to address the core modeling challenges of the LDCD system. The LDCD system involves complex coupled heat and mass transfer of multiple fluids, such as air, solution, and cold water, in a two-stage refrigeration and dehumidification process. The temperature and humidity state of the outlet air is a result of the combined effects of multiple historical inlet parameters and current dynamics. This requires the predictive model to have the ability to capture local dynamics, understand long-term bidirectional dependencies, focus on key events, and handle high-dimensional parameter optimization. Therefore, the role of each component of the model in the prediction task is as follows: (1) TCN, through its causal dilated convolution structure, can efficiently and hierarchically extract local dependency patterns and short-term trends from the input parameter sequence (such as air temperature and humidity, solution flow rate and temperature, etc., over a period of time), which directly corresponds to the physical characteristic of the system state being affected by recent operating condition fluctuations. (2) BiGRU learns the long-term contextual dependencies of the sequence from both forward and backward dimensions, based on

the local features extracted by TCN. This is crucial for simulating the LDCD system, as the current thermal and humidity state of the system is not only determined by past operating conditions but also contains an inertial trend towards a certain thermodynamic equilibrium. BiGRU can effectively model this global dynamic process. (3) SA is integrated to dynamically evaluate the importance of features at different historical moments. In LDCD operation, for example, the impact of step changes in solution concentration on current predictions is much greater than that during stable operation periods. The SA mechanism can adaptively assign higher weights to key time steps, allowing the model to “focus” on the transient processes that are most decisive for predictions, thereby improving the accuracy of characterizing the system’s nonlinear response. Finally, (4) the introduction of the ITSO algorithm aims to solve the inherent problem of the automatic optimization of high-dimensional hyperparameters in the aforementioned TCN-BiGRU-SA hybrid architecture. Manually tuning such a complex model is prone to falling into local optima and is inefficient. Through improvements in chaotic mapping initialization, nonlinear adaptive weights, and Levy flight strategies, the ITSO algorithm achieves intelligent and efficient global search in the model’s hyperparameter space, ensuring that this complex architecture can be automatically adjusted to its optimal performance state. It is a key guarantee for transforming the potential of theoretical models into practical high-precision predictions. In summary, each part of this model is tailored to the physical characteristics of LDCD system operation, collectively forming an adaptive collaborative prediction framework.



**Figure 6.** Schematic diagram of ITSO-TCN-BiGRU-SA.

The TCN-BiGRU model is well-suited for processing long-sequence data, while the SA enhances the model’s ability to focus on critical information across varying time scales and mitigates issues such as gradient vanishing in long sequences [42]. By dynamically assigning self-attention weights to different time steps, the mechanism improves feature identification and extracts more meaningful representations from temporal data. Consequently, the TCN-BiGRU-SA model, which integrates the SA into the TCN-BiGRU framework, can selectively extract key features from various time steps, capture long-term dependencies and essential information within the sequence, and ultimately improve the model’s predictive performance.

If the setting parameters in the neural network are given manually, it is difficult to find the most suitable parameter values for the model. Therefore, the ITSO optimization algorithm is introduced into the TCN-BiGRU-SA model, and the learning rate, TCN expansion coefficient, filter number, BiGRU hidden layer neuron number, self-attention mechanism key value and regularization parameter, which have great influence on the performance of model, are optimized to find the most suitable parameter values. The mean absolute

percentage error (MAPE) between the predicted and expected values is adopted as the fitness function, as defined by Equation (19).

$$Y_{MAPE} = \frac{1}{n} \sum_{t=1}^n \frac{|y_t - \hat{y}_t|}{y_t} \times 100\% \quad (19)$$

The specific steps are as follows:

Step 1: Divide the dataset into training and test sets, and normalize the data.

Step 2: Optimize the hyperparameters. Define the lower and upper bounds of hyperparameters, set the initial tuna population size, and specify the maximum number of iterations. Generate the initial population position using Equation (13).

Step 3: Compute the fitness function of the ITSO algorithm and determine optimal tuna individual position based on the ITSO algorithm's search principles.

Step 4: Set a random number  $rand$ ; if  $rand < z$ , then use Equation (10) to initialize the position, otherwise use the improved foraging method to update the position. When  $rand < 0.5$ , the parabolic foraging strategy of Equation (11) is used to update the position, and the improved spiral foraging strategy of Equation (18) is used to update the position.

Step 5: Substitute the individual parameters of the updated position into the TCN-BiGRU-SA model, and save the predicted temperature and humidity results. Return to step 3 if the maximum number of iterations is not reached. Otherwise, output the optimal parameters and results and complete the algorithm iteration procedure.

## 4. Results and Discussion

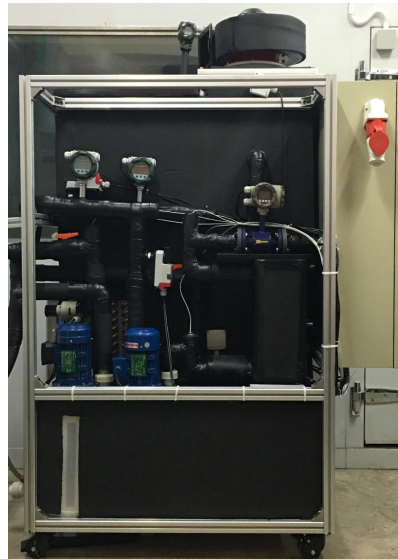
### 4.1. Experimental Data and Network Parameters

The experimental platform of the LDCD system is illustrated in Figure 7. The liquid desiccant used in the platform is lithium chloride solution. Key parameters, including the flow, temperature and humidity of air, flow and temperature of chilled water, and the flow, temperature and concentration of desiccant solution are measured and collected. Given that the outlet air temperature and humidity are influenced by multiple factors and considering the characteristics of the heat and mass transfer processes, the temperature and humidity predictions are conducted separately. The input features for air temperature prediction model include the inlet air temperature, humidity, and mass flow rate, as well as the mass flow rate and temperature of the chilled water and desiccant solution. For the air humidity prediction model, additional input features such as the desiccant solution concentration are included alongside the aforementioned parameters.

A total of 3268 sets of sample data were collected, with an interval of 10 s between each set of sample data. To avoid evaluation bias resulting from data leakage, we partitioned the dataset chronologically into training (80%) and testing (20%) sets, rather than using random partitioning. Although this approach may lead to slight differences in operating condition distributions between the training and testing sets, it better simulates real-world engineering scenarios where historical data is used to predict future operational states, thereby ensuring the practical utility and reliability of the evaluation results. Moreover, to eliminate dimensional differences and enhance computational efficiency, the sample data were normalized. The normalization formula is as follows:

$$D_n = \frac{D - D_{\min}}{D_{\max} - D_{\min}} \quad (20)$$

where  $D_{\min}$  and  $D_{\max}$  are the minimum and maximum values in sample data, respectively.



**Figure 7.** Pilot plant of LDCD system.

The sliding window method is adopted to construct the input data sequence, with a window length of 8 time steps and a step size of 1, thereby achieving overlapping sampling between adjacent windows. The hardware operating environment for this experiment consists of an Intel(R) Core(TM) Ultra 5 125H processor (with a main frequency of 3.60 GHz) and 32 GB of RAM, and the operating system is Windows 11. The training and simulation of the proposed model are both completed on the Matlab R2023b software platform. Before network training, the basic network parameters of ITSO-TCN-BiGRU-SA are set. In this paper, the Adam optimizer is selected, ReLU function is selected as the activation function, a single-head self-attention mechanism is adopted, convolution kernel size is set to 5, Dropout coefficient is 0.1, and maximum training times is 100. The learning rate, expansion coefficient, number of filters, number of neurons in the BiGRU hidden layer, key value of the self-attention mechanism, and the regularization parameter are optimized by the ITSO algorithm. The initial tuna population size is set to 10, and the maximum iteration is set to 50. The optimal parameters identified by the ITSO algorithm for the temperature model and the humidity models are presented in Table 1 and Table 2, respectively.

**Table 1.** Optimal parameters for air temperature prediction model.

| Model Parameters                        | Range            | Optimal Value |
|-----------------------------------------|------------------|---------------|
| Learning rate                           | [0.0001, 0.01]   | 0.003         |
| Number of filters                       | [1, 128]         | 3             |
| Expansion coefficient                   | [1, 2, 4, 8, 16] | [1, 2, 4]     |
| Number of neurons in BiGRU hidden layer | [1, 200]         | 79            |
| Key value of SA                         | [2, 100]         | 4             |
| Regularization parameter                | [0.00001, 0.001] | 0.00003       |

**Table 2.** Optimal parameters for air humidity prediction model.

| Model Parameters                        | Range            | Optimal Value |
|-----------------------------------------|------------------|---------------|
| Learning rate                           | [0.0001, 0.01]   | 0.0036        |
| Number of filters                       | [1, 128]         | 5             |
| Expansion coefficient                   | [1, 2, 4, 8, 16] | [1, 2, 4]     |
| Number of neurons in BiGRU hidden layer | [1, 200]         | 84            |
| Key value of SA                         | [2, 100]         | 38            |
| Regularization parameter                | [0.00001, 0.001] | 0.00018       |

#### 4.2. Model Validation

To validate the effectiveness of the proposed model, the TCN, CNN-LSTM, TCN-BiGRU, TCN-BiGRU-SA, TSO-TCN-BiGRU-SA models and the proposed ITSO-TCN-BiGRU-SA model were compared and verified in the temperature dataset and humidity dataset, respectively. The evaluation was performed using several widely recognized metrics, including mean absolute percentage error (MAPE), root mean square error (RMSE), mean square error (MSE), mean absolute error (MAE) and coefficient of determination ( $R^2$ ). These metrics were employed to comprehensively assess the performance and accuracy of the developed models. The results are depicted in Figures 8–11. The air temperature and humidity prediction curves of the different prediction models are illustrated in Figures 8 and 9. The comparisons of fitness value curves of TSO-TCN-BiGRU-SA and ITSO-TCN-BiGRU-SA in the temperature model and humidity model are represented in Figures 10 and 11. The error indexes are given in Tables 3 and 4.

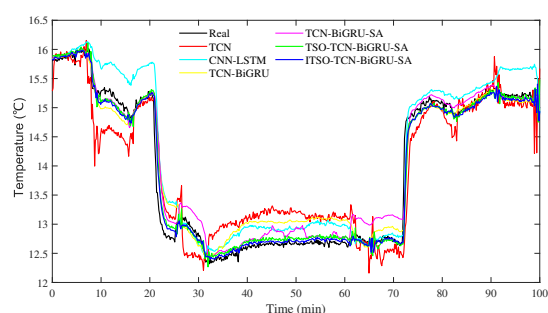


Figure 8. Comparison of air temperature model prediction curves.

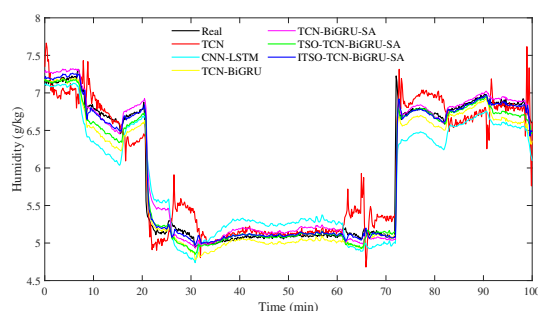


Figure 9. Comparison of air humidity model prediction curves.

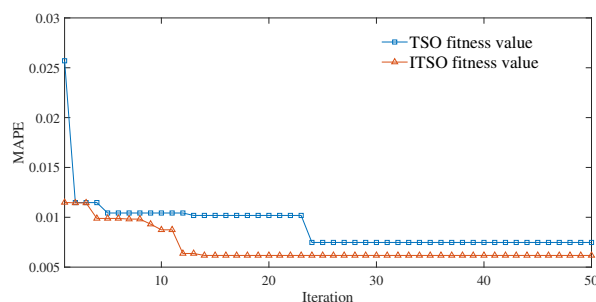
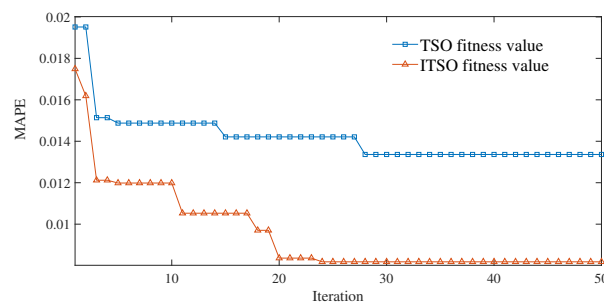


Figure 10. The fitness value curve of the air temperature model.



**Figure 11.** The fitness value curve of the air humidity model.

As illustrated in Figures 8 and 9, the TCN-BiGRU model demonstrates superior prediction accuracy for both temperature and humidity compared to the standalone TCN model. In the TCN-BiGRU architecture, the TCN component effectively captures local features, while the BiGRU component enhances the model by leveraging bidirectional contextual relationships. This dual mechanism enables the model to gain a more comprehensive understanding of sequential data, thereby significantly improving prediction performance. Similarly, the prediction accuracy of the TCN-BiGRU-SA temperature and humidity models outperforms the TCN-BiGRU model. After adding the optimization algorithm, the MAPE of the TSO-TCN-BiGRU-SA temperature model decreased by 0.487% compared with TCN-BiGRU-SA, and  $R^2$  increased by 1.606%. The MAPE of the ITSO-TCN-BiGRU-SA model decreased by 0.131% compared with TSO-TCN-BiGRU-SA model, and  $R^2$  increased by 0.495%. The MAPE of the TSO-TCN-BiGRU-SA humidity model decreased by 0.27% compared with TCN-BiGRU-SA model, and  $R^2$  increased by 0.651%. The MAPE of the ITSO-TCN-BiGRU-SA humidity model decreased by 0.518% compared with TSO-TCN-BiGRU-SA model, and  $R^2$  increased by 0.698%. As depicted in Figures 10 and 11, the ITSO algorithm exhibits a notably faster convergence speed compared to the TSO algorithm, while also achieving a smaller fitness value. This demonstrates the enhanced performance of the ITSO algorithm, which incorporates the nonlinear adaptive weights and Levy operator.

**Table 3.** Model validated errors of the air temperature prediction models.

| Models             | RMSE   | MAE    | MAPE  | $R^2$  |
|--------------------|--------|--------|-------|--------|
| TCN                | 0.4363 | 0.3519 | 2.60% | 88.70% |
| CNN-LSTM           | 0.3539 | 0.2852 | 2.05% | 92.56% |
| TCN-BiGRU          | 0.2888 | 0.2145 | 1.62% | 95.05% |
| TCN-BiGRU-SA       | 0.2347 | 0.1648 | 1.24% | 96.73% |
| TSO- TCN-BiGRU-SA  | 0.1676 | 0.103  | 0.75% | 98.34% |
| ITSO- TCN-BiGRU-SA | 0.1404 | 0.0865 | 0.62% | 98.83% |

**Table 4.** Model validated errors of the air humidity prediction models.

| Models             | RMSE   | MAE    | MAPE  | $R^2$  |
|--------------------|--------|--------|-------|--------|
| TCN                | 0.2243 | 0.1664 | 2.79% | 93.53% |
| CNN-LSTM           | 0.2912 | 0.2475 | 4.15% | 89.08% |
| TCN-BiGRU          | 0.1744 | 0.1257 | 2.06% | 96.09% |
| TCN-BiGRU-SA       | 0.1536 | 0.092  | 1.61% | 96.96% |
| TSO- TCN-BiGRU-SA  | 0.1363 | 0.0805 | 1.34% | 97.61% |
| ITSO- TCN-BiGRU-SA | 0.1147 | 0.0486 | 0.82% | 98.31% |

From Tables 3 and 4, it is obvious that compared with the TCN, TCN-BiGRU, TCN-BiGRU-SA and TSO-TCN-BiGRU-SA temperature prediction models, the MAPE of the

proposed ITSO-TCN-BiGRU-SA temperature model is reduced by 1.983%, 1.000%, 0.618%, and 0.131%, and  $R^2$  is improved by 10.127%, 3.78%, 2.101%, and 0.495%, respectively; the MAPE of the proposed ITSO-TCN-BiGRU-SA humidity model is reduced by 1.972%, 1.240%, 0.788% and 0.518%, and  $R^2$  improved by 4.778%, 2.219%, 1.349% and 0.698%, respectively. In summary, the proposed ITSO-TCN-BiGRU-SA model demonstrates superior prediction accuracy compared to the other models, thereby validating its effectiveness and robustness.

Although this study did not conduct large-scale multi-round repeated experiments, fixed random seeds were set during model training to ensure reproducible results. Observation of the training process revealed that the model converged smoothly within 100 epochs, and the loss curves of the training set and validation set were in good agreement, indicating stable model training without overfitting. For some sample points with relatively large prediction errors, verification showed that they mainly occurred during the transient stages of system startup or valve regulation, when the system was in a non-steady state, increasing the difficulty of instantaneous prediction. Once the system entered a steady state, the model's prediction accuracy recovered rapidly.

#### 4.3. Performance Investigation

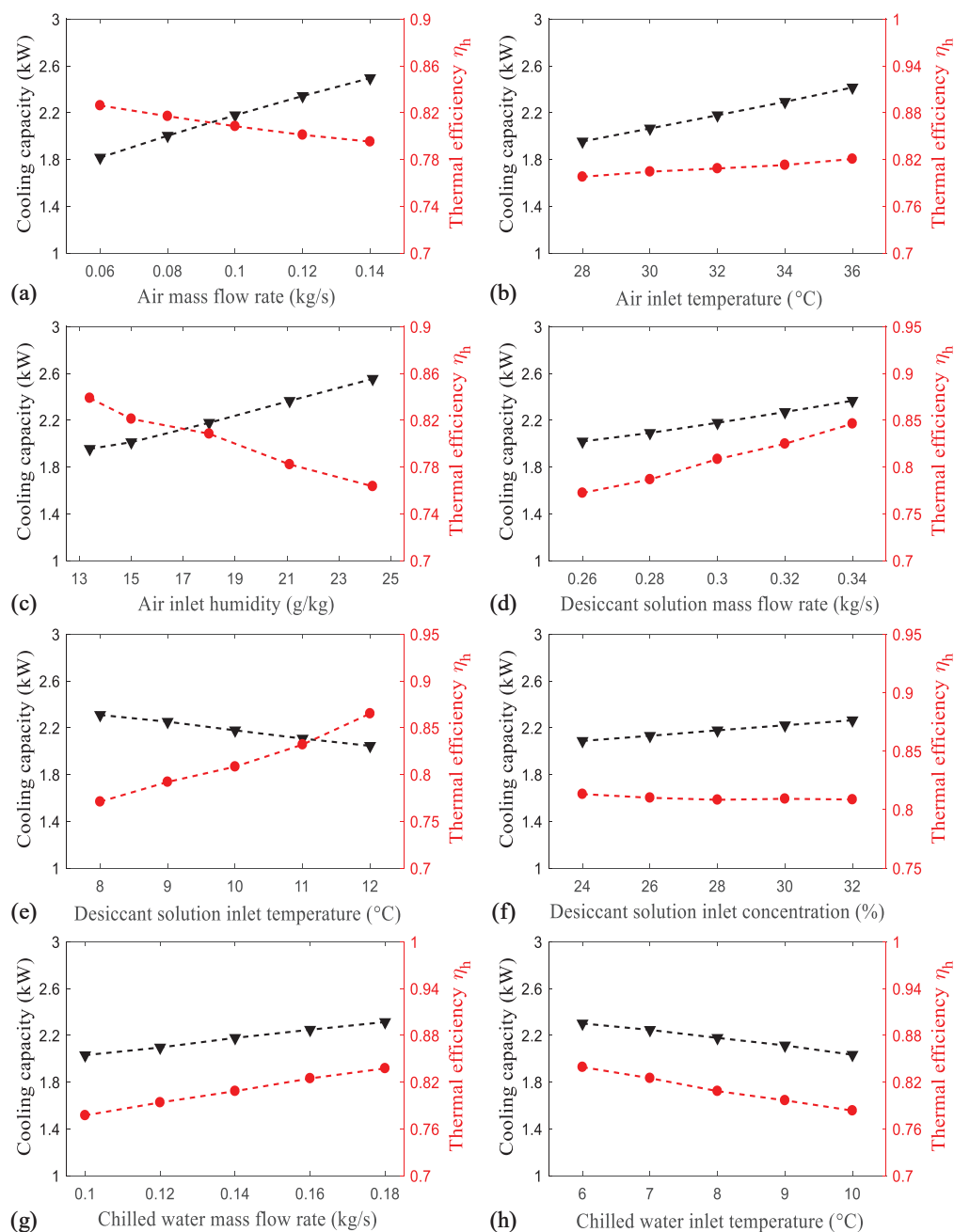
In the LDDS system, the operating variables exert varying degrees of influence on the system's cooling and dehumidification performance. To provide practical guidance for adjusting the system's adjustable parameters, an analysis of how the relevant inlet parameters affect the system's mass and heat transfer performance is conducted, based on the proposed ITSO-TCN-BiGRU-SA model. In each scenario, only one variable is modified while the others are maintained at the reference values. In the LDCD system, the cooling and dehumidification performance is affected by various inlet parameters, including the flow rate  $\dot{m}_a$ ,  $\dot{m}_s$ ,  $\dot{m}_w$ , inlet temperature  $T_{a,i}$ ,  $T_{s,i}$ ,  $T_{w,i}$ , humidity  $\omega_{a,i}$  and concentration  $\xi_i$ . The operating ranges and reference values of the inlet parameters are provided in Table 5. The cooling and dehumidification performance are systematically compared and analyzed. The impact of operating parameters on the cooling capacity and thermal efficiency is illustrated in Figure 12, while their effects on dehumidification effectiveness and moisture removal rate are shown in Figure 13.

**Table 5.** The operating parameters settings of the LDCD system.

| Parameters         |                | Reference | Range     | Unit |
|--------------------|----------------|-----------|-----------|------|
| Outdoor air        | temperature    | 32        | 28–36     | °C   |
|                    | humidity ratio | 18.0      | 13.4–24.3 | g/kg |
|                    | flow rate      | 0.10      | 0.06–0.14 | kg/s |
| Chilled water      | temperature    | 8         | 6–10      | °C   |
|                    | flow rate      | 0.14      | 0.10–0.18 | kg/s |
| Desiccant solution | temperature    | 10        | 8–12      | °C   |
|                    | flow rate      | 0.30      | 0.26–0.34 | kg/s |
|                    | concentration  | 28        | 24–32     | %    |

As illustrated in Figures 12 and 13, the air inlet humidity, mass flow rate of air, desiccant solution and chilled water exhibit a positive effect on both the moisture removal rate and cooling capacity. In contrast, the desiccant solution and chilled water inlet temperature show a negative effect on both moisture removal rate and cooling capacity. The air inlet temperature positively influences cooling capacity but has minimal effect on the moisture removal rate. Similarly, the desiccant solution inlet concentration enhances the moisture removal rate but has negligible impact on the cooling capacity. Regarding thermal effi-

ciency and moisture effectiveness, the mass flow rate of desiccant solution and chilled water demonstrates a positive influence, whereas the air mass flow rate and chilled water inlet temperature exhibit a negative effect. The air inlet temperature and desiccant solution inlet concentration have little to no effect on these metrics. The air inlet humidity negatively affects thermal efficiency but has minimal impact on moisture effectiveness. The desiccant solution inlet temperature positively influences thermal efficiency while negatively affecting moisture effectiveness.



**Figure 12.** Influences of inlet parameters on cooling performance.

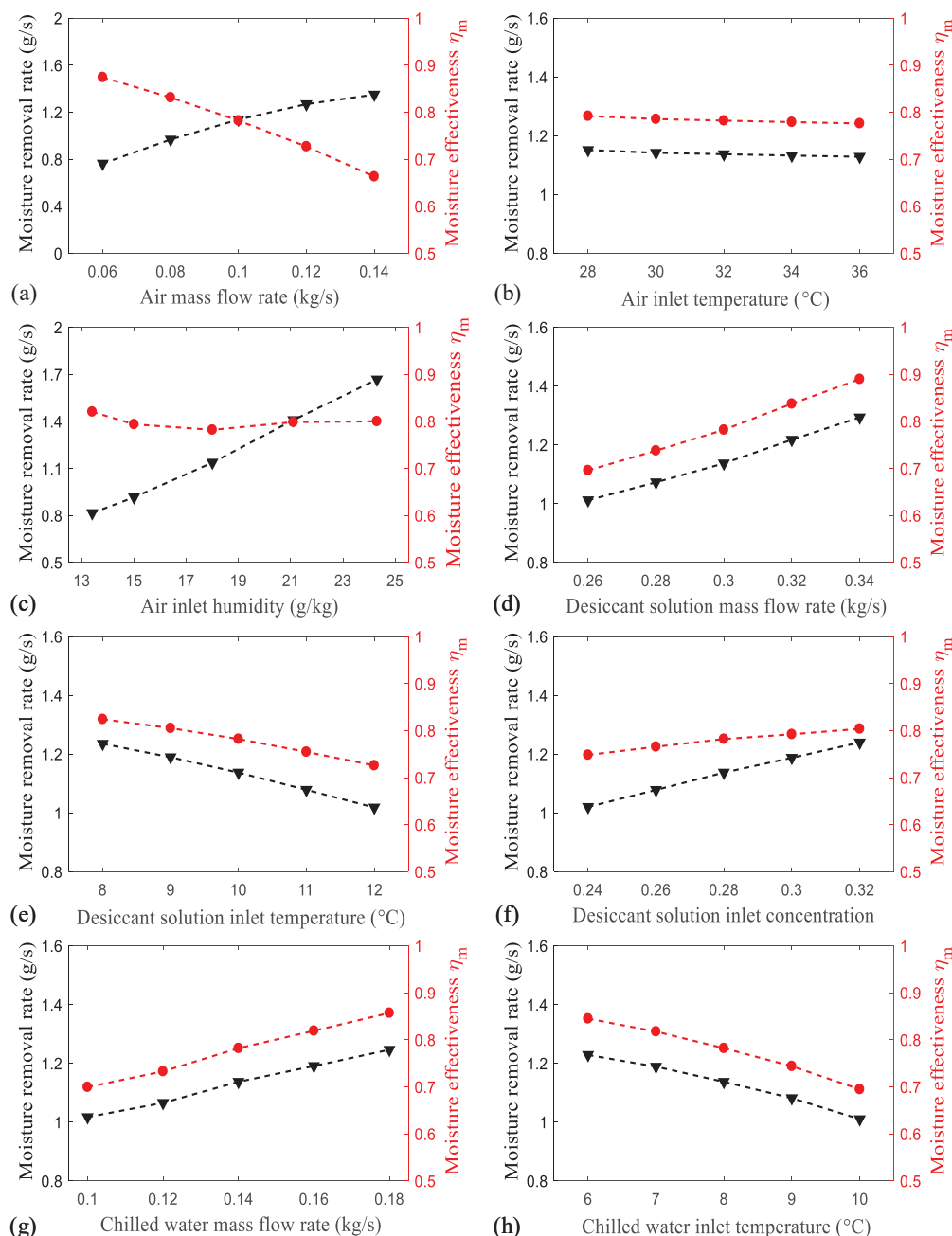


Figure 13. Influences of inlet parameters on the dehumidification performance.

## 5. Conclusions

This study aims to develop a data-driven model for the two-stage LDCD system to accurately predict the supply air temperature and humidity. The ITSO-TCN-BiGRU-SA hybrid prediction model constructed in this study integrates the TCN-BiGRU-SA network architecture with an improved TSO algorithm (incorporating the Levy operator and non-linear adaptive weights). It demonstrates excellent performance in sequence modeling and prediction tasks, with high accuracy and strong applicability for predicting the supply air temperature and humidity of the LDCD system, thus supporting the optimization of control performance and energy efficiency improvement. Based on the proposed model, this study reveals that the influence of system inlet parameters on cooling and dehumidification performance is driven by the internal thermodynamics and heat-mass transfer mechanisms of the LDCD: the air inlet temperature affects the cooling capacity through the sensible-heat temperature difference, while the solution inlet temperature influences

the dehumidification performance through the water vapor partial pressure; the air inlet humidity ratio affects both the cooling load and the mass-transfer process, and the solution concentration significantly enhances the moisture-absorption potential; and flow rate changes comprehensively act on the thermal–moistureexchange performance from the dimensions of load and efficiency. The research results provide practical guidance for the LDCD system: in design, prioritize ensuring a high-concentration solution to maintain strong dehumidification capacity; in operation, adopt a decoupled optimization strategy; and in the future, combine the parameter interaction model with advanced algorithms to achieve dynamic collaborative optimization of multi-loop parameters, minimizing the total energy consumption while meeting thermal and moisture demands.

**Author Contributions:** Methodology, X.W. and Z.W.; Software, X.W. and Z.W.; Validation, X.W. and Z.W.; Writing—original draft, X.W. and Z.W.; Writing—review & editing, X.O.; Supervision, X.O.; Funding acquisition, X.O. All authors have read and agreed to the published version of the manuscript.

**Funding:** This document is the result of the research project funded by the National Natural Science Foundation of China (No. 62303418) and the Zhejiang Provincial Natural Science Foundation of China (No. LQ22F030016).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data presented in this study are available on request from the corresponding author.

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

## References

1. Chua, K.J.; Chou, S.K.; Yang, W.M.; Yan, J. Achieving better energy-efficient air conditioning—A review of technologies and strategies. *Appl. Energy* **2013**, *104*, 87–104. [[CrossRef](#)]
2. Lof, G.O. Cooling with solar energy. In Proceedings of the Congress of Solar Energy, Tucson, AZ, USA, 31 October–1 November 1955; Volume 2, pp. 171–189.
3. Abdel-Salam, A.H.; Simonson, C.J. Capacity matching in heat-pump membrane liquid desiccant air conditioning systems. *Int. J. Refrig.* **2014**, *48*, 166–177. [[CrossRef](#)]
4. Su, B.; Han, W.; Sui, J.; Jin, H. A two-stage liquid desiccant dehumidification system by the cascade utilization of low-temperature heat for industrial applications. *Appl. Energy* **2017**, *207*, 643–653. [[CrossRef](#)]
5. Zhang, G.; Tan, J.; Xie, J.; Du, B.; Liu, H.; Liu, J. Experimental and 4NTU-Le heat and mass transfer model theoretical analysis based on a novel internally cooled liquid desiccant dehumidifier. *Energy* **2024**, *313*, 134053. [[CrossRef](#)]
6. Zhang, Y.; Zhang, H.; Yang, H.; Chen, Y.; Leung, C.W. Response surface modeling and optimization scheme of an internally cooled liquid desiccant air conditioning system. *J. Build. Eng.* **2023**, *76*, 107371. [[CrossRef](#)]
7. Wang, L.; Xiao, F.; Niu, X.; Gao, D.c. A dynamic dehumidifier model for simulations and control of liquid desiccant hybrid air conditioning systems. *Energy Build.* **2017**, *140*, 418–429. [[CrossRef](#)]
8. Cao, B.; Yin, Y.; Zhang, F.; Ji, Q.; Chen, W. Liquid desiccant-based deep dehumidifier working with a novel ionic liquid: Prediction model and performance comparison. *Int. J. Refrig.* **2023**, *146*, 74–87. [[CrossRef](#)]
9. Liu, X.; Qu, M.; Liu, X.; Wang, L.; Warner, J.; Gao, Z. Numerical modeling and performance analysis of a membrane-based air dehumidifier using ionic liquid desiccant. *Appl. Therm. Eng.* **2020**, *181*, 115754. [[CrossRef](#)]
10. Ou, X.; Cai, W.; He, X.; Zhai, D. Experimental investigations on heat and mass transfer performances of a liquid desiccant cooling and dehumidification system. *Appl. Energy* **2018**, *220*, 164–175. [[CrossRef](#)]
11. Bhowmik, M.; Naik, B.K.; Muthukumar, P.; Anandalakshmi, R. Performance assessment and optimization of liquid desiccant dehumidifier system using intelligent models and integration with solar dryer. *J. Build. Eng.* **2023**, *64*, 105577. [[CrossRef](#)]
12. Ou, X.; Cai, W.; He, X. Model-based optimization strategy for a liquid desiccant cooling and dehumidification system. *Energy Build.* **2019**, *194*, 21–32. [[CrossRef](#)]
13. Zhang, N.; Yin, S.Y.; Li, M. Model-based optimization for a heat pump driven and hollow fiber membrane hybrid two-stage liquid desiccant air dehumidification system. *Appl. Energy* **2018**, *228*, 12–20. [[CrossRef](#)]

14. Jiang, Y.; Wang, X.; Zhao, H.; Wang, L.; Yin, X.; Jia, L. Dynamic modeling and economic model predictive control of a liquid desiccant air conditioning. *Appl. Energy* **2020**, *259*, 114174. [[CrossRef](#)]
15. Liu, W.; Gong, Y.; Niu, X.; Shen, J.; Kosonen, R. Dynamic modeling of liquid-desiccant regenerator based on a state-space method. *Appl. Energy* **2019**, *240*, 744–753. [[CrossRef](#)]
16. Chen, X.; Li, Z.; Jiang, Y.; Qu, K. Analytical solution of adiabatic heat and mass transfer process in packed-type liquid desiccant equipment and its application. *Sol. Energy* **2006**, *80*, 1509–1516. [[CrossRef](#)]
17. Dong, C.; Lu, L.; Qi, R. Model development of heat/mass transfer for internally cooled dehumidifier concerning liquid film shrinkage shape and contact angles. *Build. Environ.* **2017**, *114*, 11–22. [[CrossRef](#)]
18. Babakhani, D.; Soleymani, M. An analytical solution for air dehumidification by liquid desiccant in a packed column. *Int. Commun. Heat Mass Transf.* **2009**, *36*, 969–977. [[CrossRef](#)]
19. Peng, D.; Zhang, X. An analytical model for coupled heat and mass transfer processes in solar collector/regenerator using liquid desiccant. *Appl. Energy* **2011**, *88*, 2436–2444. [[CrossRef](#)]
20. Shen, S.; Cai, W.; Wang, X.; Wu, Q.; Yon, H. Investigation of liquid desiccant regenerator with fixed-plate heat recovery system. *Energy* **2017**, *137*, 172–182. [[CrossRef](#)]
21. Wang, X.; Wang, L.; Yin, X.; Jiang, Y.; Liu, H.; Jia, L. Modeling and performance analyses of a batch-wise liquid desiccant air conditioning system. *Build. Environ.* **2019**, *154*, 1–12. [[CrossRef](#)]
22. Yon, H.R.; Cai, W.; Wang, Y.; Wang, X.; Shen, S. Dynamic model for a novel liquid desiccant regeneration system operating in vacuum condition. *Energy Build.* **2018**, *167*, 69–78. [[CrossRef](#)]
23. Wu, Q.; Cai, W.; Wang, X.; Chakraborty, A. Dehumidifier desiccant concentration soft-sensor for a distributed operating Liquid Desiccant Dehumidification System. *Energy Build.* **2016**, *129*, 215–226. [[CrossRef](#)]
24. Wu, Q.; Cai, W.; Shen, S.; Wang, X.; Ren, H. A regulation strategy of working concentration in the dehumidifier of liquid desiccant air conditioner. *Appl. Energy* **2017**, *202*, 648–661. [[CrossRef](#)]
25. Li, X.; Liu, S.; Tan, K.K.; Wang, Q.G.; Cai, W.J.; Xie, L. Dynamic modeling of a liquid desiccant dehumidifier. *Appl. Energy* **2016**, *180*, 435–445. [[CrossRef](#)]
26. Gandhidasan, P.; Mohandes, M. Artificial neural network analysis of liquid desiccant dehumidification system. *Energy* **2011**, *36*, 1180–1186. [[CrossRef](#)]
27. Mohammad, A.T.; Mat, S.B.; Sulaiman, M.; Sopian, K.; Al-Abidi, A.A. Implementation and validation of an artificial neural network for predicting the performance of a liquid desiccant dehumidifier. *Energy Convers. Manag.* **2013**, *67*, 240–250. [[CrossRef](#)]
28. Oleolo, I.; Abdullah, H.; Mustapha, I.; Mohamad, M.; Jaafar, M.N.M.; Olowolayemo, A.; Sulaiman, S. Long Short-Term Memory Neural Network Model for the Control of Temperature in A Multi-Circuit Air Conditioning System. *CFD Lett.* **2022**, *14*, 84–98. [[CrossRef](#)]
29. Elmaz, F.; Eyckerman, R.; Casteels, W.; Latré, S.; Hellinckx, P. CNN-LSTM architecture for predictive indoor temperature modeling. *Build. Environ.* **2021**, *206*, 108327. [[CrossRef](#)]
30. Wu, D.C.; Momeni, M.; Razban, A.; Chen, J. Optimizing demand-controlled ventilation with thermal comfort and CO<sub>2</sub> concentrations using long short-term memory and genetic algorithm. *Build. Environ.* **2023**, *243*, 110676. [[CrossRef](#)]
31. Guo, Z.; Moayedi, H.; Foong, L.K.; Bahiraei, M. Optimal modification of heating, ventilation, and air conditioning system performances in residential buildings using the integration of metaheuristic optimization and neural computing. *Energy Build.* **2020**, *214*, 109866. [[CrossRef](#)]
32. Liang, R.; Le-Hung, T.; Nguyen-Thoi, T. Energy consumption prediction of air-conditioning systems in eco-buildings using hunger games search optimization-based artificial neural network model. *J. Build. Eng.* **2022**, *59*, 105087. [[CrossRef](#)]
33. Zhao, Y.; Li, W.; Zhang, J.; Jiang, C.; Chen, S. Real-time energy consumption prediction method for air-conditioning system based on long short-term memory neural network. *Energy Build.* **2023**, *298*, 113527. [[CrossRef](#)]
34. Zhou, M.; Wang, L.; Hu, F.; Zhu, Z.; Zhang, Q.; Kong, W.; Zhou, G.; Wu, C.; Cui, E. ISSA-LSTM: A new data-driven method of heat load forecasting for building air conditioning. *Energy Build.* **2024**, *321*, 114698. [[CrossRef](#)]
35. Mohammad, A.T.; Mat, S.B.; Sopian, K.; Al-Abidi, A.A. Survey of the control strategy of liquid desiccant systems. *Renew. Sustain. Energy Rev.* **2016**, *58*, 250–258. [[CrossRef](#)]
36. Das, R.S.; Jain, S. Experimental investigations on a solar assisted liquid desiccant cooling system with indirect contact dehumidifier. *Sol. Energy* **2017**, *153*, 289–300. [[CrossRef](#)]
37. Abdel-Salam, A.H.; Ge, G.; Simonson, C.J. Performance analysis of a membrane liquid desiccant air-conditioning system. *Energy Build.* **2013**, *62*, 559–569. [[CrossRef](#)]
38. Yang, Z.; Zhang, K.; Hwang, Y.; Lian, Z. Performance investigation on the ultrasonic atomization liquid desiccant regeneration system. *Appl. Energy* **2016**, *171*, 12–25. [[CrossRef](#)]
39. Ashish, V. Attention is all you need. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2017; Volume 30.

40. Xie, L.; Han, T.; Zhou, H.; Zhang, Z.R.; Han, B.; Tang, A. Tuna swarm optimization: A novel swarm-based metaheuristic algorithm for global optimization. *Comput. Intell. Neurosci.* **2021**, 2021, 9210050. [[CrossRef](#)] [[PubMed](#)]
41. Zhang, J.; Wang, J.S. Improved whale optimization algorithm based on nonlinear adaptive weight and golden sine operator. *IEEE Access* **2020**, 8, 77013–77048. [[CrossRef](#)]
42. Zhan, X.; Kou, L.; Xue, M.; Zhang, J.; Zhou, L. Reliable long-term energy load trend prediction model for smart grid using hierarchical decomposition self-attention network. *IEEE Trans. Reliab.* **2022**, 72, 609–621.

**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.