agronomy

MDPI

Article

Prediction Model of Nitrogen, Phosphorus, and Potassium Fertilizer Application Rate for Greenhouse Tomatoes under Different Soil Fertility Conditions

Xiaoyu Yu [1], Yuzhu Luo [1], Bing Bai [1], Xin Chen [2], Caiyan Lu [2] and Xiuyuan Peng [2,1,*]

[1] Institute of Information, Liaoning Academy of Agricultural Sciences, Shenyang 110161, China; yxyu602@163.com (X.Y.); luoyuzhu1384218a@163.com (Y.L.); bbing202403@163.com (B.B.)
[2] Key Laboratory of Pollution Ecology and Environment Engineering, Institute of Applied Ecology, Chinese Academy of Sciences, Shenyang 110016, China; chenxin@iae.ac.cn (X.C.); lucaiyan@iae.ac.cn (C.L.)
* Correspondence: peng_xiuyuan@163.com; Tel.: +86-186-2405-6996

Abstract: To reach the target yield of crops, nutrient management is essential. Selecting the appropriate prediction model and adjusting the nutrient supply based on the actual situation can effectively improve the nutrient utilization efficiency, crop yield, and product quality. Therefore, a prediction model of the NPK fertilizer application rate for greenhouse tomatoes under the target yield was studied in this study. Under low, medium, and high soil fertility conditions, a neural network prediction model based on the sparrow search algorithm (SSA-NN), a neural network prediction model based on the improved sparrow search algorithm (ISSA-NN), and a neural network prediction model based on the hybrid algorithm (HA-NN) were used to predict the NPK fertilizer application rate for greenhouse tomatoes. The experimental results indicated that the evaluation indexes (i.e., the mean square error (MSE), explained variance score (EVS), and coefficient of determination (R^2)) of the HA-NN prediction model proposed in this study were superior than the SSA-NN and ISSA-NN prediction models under three different soil fertility conditions. Under high soil fertility, compared with the SSA-NN prediction model, the MSE of the ISSA-NN and HA-NN prediction models decreased to 0.007 and 0.005, respectively; the EVS increased to 0.871 and 0.908, respectively; and the R^2 increased to 0.862 and 0.899, respectively. This study showed that the HA–NN prediction model was superior in predicting the NPK fertilizer application rate for greenhouse tomatoes under three different soil fertility conditions. Due to the significance of NPK fertilizer application rate prediction for greenhouse tomatoes, this technique is expected to bring benefits to agricultural production management and decision support.

Keywords: ant colony algorithm; genetic algorithm; greenhouse tomatoes; inorganic fertilizer prediction; sparrow search algorithm

check for updates

Citation: Yu, X.; Luo, Y.; Bai, B.; Chen, X.; Lu, C.; Peng, X. Prediction Model of Nitrogen, Phosphorus, and Potassium Fertilizer Application Rate for Greenhouse Tomatoes under Different Soil Fertility Conditions. *Agronomy* **2024**, *14*, 1165. https://doi.org/10.3390/agronomy14061165

Academic Editor: Bertrand Hirel

Received: 19 April 2024
Revised: 27 May 2024
Accepted: 28 May 2024
Published: 29 May 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The greenhouse environment is a complex system with multivariable, nonlinear, and strong coupling [1–3]. In China, tomatoes stand as the primary crop cultivated within greenhouses, and the tomato industry plays a significant role in the development of the agricultural economy. Nutrient management has an important effect on the yield and quality of greenhouse tomatoes. Too much or too little fertilizer input can result in reduced yield and quality and ecosystem imbalance. Nutrient management needs to comprehensively consider factors such as soil fertility, the nutrient requirements of different crops, and the climatic conditions, and the nutrient supply should be consistent with crop demand to reduce nutrient waste in the field [4–7]. To sum up, crop nutrient demand, management, and rational fertilization are very significant. They can improve the efficiency of agricultural production, help protect the ecological environment, and promote

the sustainable development of agriculture. Therefore, it is necessary to carry out in-depth research and practice.

Recently, neural networks (NNs) have become a focus in establishing forecasting models [8–11]. These networks are characterized by their large-scale parallel processing frameworks, efficient operation speeds, remarkable adaptability, and exceptional self-organizing and self-learning capabilities. Such attributes have positioned NNs as pivotal tool in nonlinear system modeling, leading to their extensive application across diverse domains. Due to their proficiency in learning and simulating intricate nonlinear relationships, numerous researchers have adopted NNs for greenhouse production systems. Consequently, neural network technology has found widespread utilization in various agricultural prediction models [12–14].

At present, there are few studies on prediction models of NPK fertilizer application rate for greenhouse tomatoes. Some researchers have studied the precise fertilization model of corn crops [15], but more studies are focused on forecasting models of greenhouse tomato yields under the condition of the target fertilizer application rate. Alhnaity et al. employed long short-term memory neural networks to model the target growth parameters of crops, and machine learning and deep learning were utilized to anticipate the yield and plant-growth variations across diverse scenarios [16]. To prevent the gradient descent learning algorithm from becoming trapped in local optima, Wang adopted a wavelet neural network with swifter convergence speed, and the wavelet neural network model enhanced by a genetic algorithm was proposed to elevate the prediction accuracy of tomato yields within a Chinese solar greenhouse [17]. Belouz et al. integrated a neural network with a sensitivity analysis to forecast greenhouse tomato yields, and compared the results of the neural network model with that of a multiple linear regression model [18]. It was found that the neural network model exhibited superior predictive capabilities.

In this study, greenhouse tomatoes were selected as the subject of investigation, and the NPK fertilizer application rate for greenhouse tomatoes was predicted by a combination of a neural network model and optimization algorithm. The sparrow search algorithm (SSA) is an optimization algorithm with global search capability and the capability to escape the local optimal value [19]. The algorithm mainly implements the search process by introducing early warning behavior and a division of labor strategy [20,21]. Therefore, this study used the SSA algorithm to optimize parameters of the neural network forecasting model.

To enhance the accuracy of the SSA algorithm in predicting the NPK fertilizer application rate for greenhouse tomatoes, this study proposes an improved SSA algorithm (ISSA), which mainly improved four parts of the SSA algorithm, including initializing the population by Arnold cat mapping chaos, an adaptive adjustment strategy for the number of producers and scroungers, an improved producer position update equation, and a Lévy flight strategy to update the early warning position. Although many improvements have been made to the ISSA algorithm, it still has some limitations, such as the single search method, its ease of falling into the local optimal value, and so on. Therefore, this study introduces the genetic algorithm (GA) [22] and ant colony optimization algorithm (ACO) [23] on the basis of the ISSA Algorithm, and then proposes a hybrid algorithm (HA). The HA algorithm uses a parallel operation, and selects the optimal result after each iteration for the next iteration, so as to enhance the global search capability and the capability to escape local optima. Finally, this study proposes a neural network prediction model based on the HA algorithm, aiming to more accurately predict the NPK fertilizer application rate for greenhouse tomatoes.

The primary results of this study are as follows:

1. In this study, 390 sets of experimental data were extracted from different regions from 1999 to 2020, and based on the classification standard of the soil nutrient index for tomato production in solar greenhouse facilities, the extracted experimental data were divided into three soil fertility grades: low, medium, and high;
2. In order to solve the limitations of the SSA algorithm, such as the poor population diversity, tendency to settle into local optima, and inadequate accuracy in solving

multidimensional functions, this study improved the SSA algorithm by adopting four improvement measures and proposed the ISSA algorithm;

3. Using the ISSA algorithm as the foundation, the GA algorithm and ACO algorithm were introduced, and then an HA algorithm was proposed to enhance its global search capability and the capability to escape local optima;
4. The performance of the neural network prediction model based on the SSA algorithm (SSA–NN), the neural network prediction model based on the ISSA algorithm (ISSA–NN), and the neural network prediction model based on the HA algorithm (HA–NN) were compared in the prediction experiment of the NPK fertilizer application rate for greenhouse tomatoes under different soil fertilities, which proved that the HA-NN prediction model is the optimal and most efficient prediction model of the NPK fertilizer application rate for greenhouse tomatoes.

The remainder of this study is structured as follows: In Section 2, the materials and methods are introduced, and the sources of the data required by the model, three different prediction models, and the evaluation indexes are described. Then, in Section 3, the required parameters of the model are given, and three different prediction models of the NPK fertilizer application rate for greenhouse tomatoes under different soil fertilities are outlined. Section 4 provides a discussion to prove the effectiveness and superiority of the proposed prediction model of the NPK fertilizer application rate for greenhouse tomatoes. Finally, the content of this study is summarized in Section 5.

2. Materials and Methods

2.1. Data Source

The experimental data utilized in this study were partly derived from field trial data, and partly from Web of Science and CNKI databases [24]. The literature underwent screening based on several criteria: (1) Large tomatoes (excluding potted, processed, and cherry tomatoes); (2) solar greenhouses (excluding multi-span and cold greenhouses); (3) NPK fertilizer application data (excluding foliar spraying fertilizer); (4) yield data; (5) soil background nutrient data; (6) planting density data; (7) the effective accumulated temperature during the growth period can be calculated according to the information provided; (8) the experimental area within China.

In greenhouse tomato production, the grading standard of soil nutrients is of great significance for guiding rational fertilization and ensuring a high yield and high quality of tomatoes. Table 1 is the classification standard for soil nutrient indexes for tomato production in solar greenhouse facilities, which mainly considers the following nutrient indexes: organic matter, total nitrogen, alkaline-hydrolyzable nitrogen, available phosphorus, and available potassium. In total, 390 sets of experimental data were divided into low, medium, and high soil fertility grades (see Table 2). Then, under different soil fertility conditions, the experimental data were divided into training data and testing data according to the proportion of 8:2. The prediction model of this study uses six factors as input variables: the planting density, organic fertilizer NPK data, greenhouse tomato yield, and effective accumulated temperature data (these data were sourced from meteorological shared data and obtained according to the calculation method of the effective accumulated temperature). The predicted NPK fertilizer application rate of greenhouse tomatoes is the output variable.

Table 1. Classification standard for soil nutrient indexes for tomato production in solar greenhouse facilities.

Nutrient Index	Unit	Scoring Rules				
		Extremely High	High	Medium	Low	Extremely Low
Organic Matter	g/Kg	≥28.5	22.3–28.5	15.02–22.3	10.9–15.02	<10.9
Total Nitrogen	g/Kg	≥1.98	1.19–1.98	0.88–1.19	0.72–0.88	<0.72
Alkaline-Hydrolyzable Nitrogen	mg/kg	≥150	96–150	57.8–96	21.36–57.8	<21.36
Available Phosphorus	mg/kg	≥167	68–167	31.6–68	8.4–31.6	<8.4
Available Potassium	mg/kg	≥216	166–216	102.3–166	72.4–102.3	<72.4

Table 2. Experimental data.

Soil Fertility	Sample Number	Planting Density (plant/hm^2)	Organic Fertilizer *N* (kg/hm^2)	Organic Fertilizer *P* (kg/hm^2)	Organic Fertilizer *K* (kg/hm^2)	Effective Accumulated Temperature (°C)	Greenhouse Tomato Yield (kg/hm^2)	Inorganic Fertilizer *N* (kg/hm^2)	Inorganic Fertilizer *P* (kg/hm^2)	Inorganic Fertilizer *K* (kg/hm^2)
Low	1	34,667.00	0.00	0.00	0.00	1565.00	66,110.00	240.00	120.00	150.00
	2	33,375.00	0.00	0.00	0.00	617.20	89,173.40	480.00	240.00	300.00
	...	...	...	...	...	...	...	...	...	...
	146	60,633.00	430.32	406.56	224.40	1664.80	52,430.00	525.00	220.00	300.00
Medium	1	34,667.00	0.00	0.00	0.00	1565.00	91,100.00	180.00	90.00	112.50
	2	24,293.00	0.00	0.00	0.00	929.20	92,200.00	158.00	79.00	158.00
	...	...	...	...	...	...	...	...	...	...
	144	48,372.00	69.60	48.00	72.24	1598.40	52,141.95	570.00	438.00	738.00
High	1	41,675.00	489.00	462.00	255.00	1423.10	70,261.30	315.00	250.50	391.70
	2	53,333.00	176.25	97.50	116.25	1198.50	79,966.67	149.96	408.00	89.65
	...	...	...	...	...	...	...	...	...	...
	100	58,029.00	0.00	0.00	0.00	727.00	63,900.00	356.00	134.70	530.20

2.2. Prediction Model Based on Neural Network

The neural network model, with input, hidden, and output layers, is valued for its self-learning, adaptability, and prediction abilities [25]. Consequently, this model has been widely employed across various disciplines [26,27].

The Back Propagation (BP) neural network is a feedforward neural network, with a learning process that consists of two distinct phases: forward propagation and backpropagation [28]. During forward propagation, information is passed from the input layer to the hidden layer and ultimately to the output layer. In cases where a significant discrepancy exists between the output of the neural network and the actual target value, backpropagation is initiated. This learning approach relies on the steepest descent method to progressively fine-tune the parameters of the network, thereby enabling the predicted output of the BP neural network to move continuously closer to the actual value. The structure of the BP neural network is shown in Figure 1.

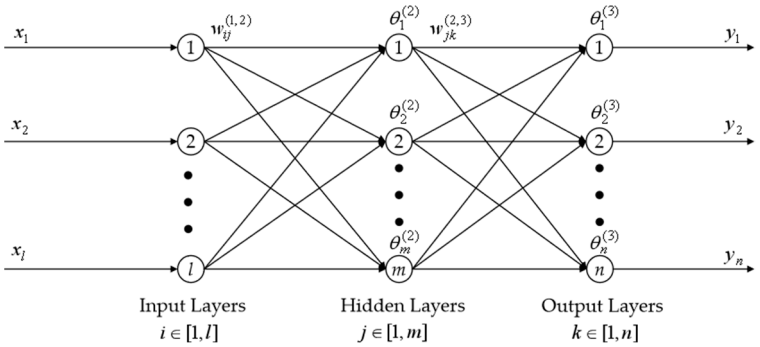


Figure 1. The structure of BP neural network.

Suppose the input $\boldsymbol{x}$ is a real vector, $\boldsymbol{x} \in R^l$, $\boldsymbol{x} = (x_1, x_2, \cdots, x_l)^{\mathrm{T}}$; the output $\boldsymbol{y}$ is a real vector, $\boldsymbol{y} \in R^n$, $\boldsymbol{y} = (y_1, y_2, \cdots, y_n)^{\mathrm{T}}$; the hidden layer has m nodes, and its output u is a real vector, $\boldsymbol{u} \in R^m$, $\boldsymbol{u} = (u_1, u_2, \cdots, u_m)^{\mathrm{T}}$; the weight from the input layer to the hidden layer is $w_{ij}^{(1,2)}$, and the threshold is $\theta_j^{(2)}$; the weight from the hidden layer to the output layer is $w_{jk}^{(2,3)}$, and the threshold is $\theta_k^{(3)}$; and they are all real numbers.

The neuron output u_j in the hidden layer and the neuron output y_k in the output layer are, respectively [25]:

$$u_j = f_1\left(\sum_{i=1}^{l} w_{ij}^{(1,2)} \cdot x_i - \theta_j^{(2)}\right) \tag{1}$$

$$y_k = f_2\left(\sum_{i=1}^{l} w_{jk}^{(2,3)} \cdot u_j - \theta_k^{(3)}\right) \tag{2}$$

where the activation function $f(\cdot)$ can be chosen from various functions based on the specific requirements of the situation.

2.3. Neural Network Prediction Model Based on Sparrow Search Algorithm

The SSA algorithm is one of many swarm intelligence algorithms. The introduction of early warning behavior and the division strategy of the SSA algorithm makes the algorithm have an excellent global search capability and the capability to escape local optima. This makes the algorithm perform well in the optimization problem and, thus, it has attracted wide attention. Therefore, in this section, we describe how the SSA algorithm optimizes the parameters of the neural network prediction model.

2.3.1. Sparrow Search Algorithm

The SSA algorithm is an intelligent optimization method inspired by the behavior of sparrow population foraging and predator avoidance. It has been widely used in many fields [29–32]. This algorithm continuously conducts local searches, selecting local operations that can improve the solution each time, and updating the current solution to achieve position optimization [33].

In the SSA algorithm, the fitness of a particle is compared to the energy of an individual sparrow, and the individuals in the population are divided into producers and scroungers according to the level of energy. The roles of producers and scroungers are constantly shifting. If the better food sources can be found, each sparrow can become a producer. When one sparrow becomes a producer, another sparrow must become a scrounger. The concept of natural enemy early warning is also included in the SSA algorithm, and some sparrows are selected as early warning sparrows, who find natural enemies and send warning signals to the whole population, thereby reducing the possibility of the algorithm becoming stuck in local optima.

Each particle represents an individual sparrow, and these individuals look for the point or position with the highest fitness according to the sparrow's movement mode. Suppose there are n sparrows in a population, then the position of n sparrows can be represented by a matrix of n rows and D columns [29]:

$$X = \begin{bmatrix} X_{1,1} & X_{1,2} & \cdots & X_{1,D} \\ X_{2,1} & X_{2,2} & \cdots & X_{2,D} \\ \vdots & \vdots & \vdots & \vdots \\ X_{n,1} & X_{n,2} & \cdots & X_{n,D} \end{bmatrix} \tag{3}$$

where D is the dimension of the variable of the problem to be optimized.

Assuming that the individual fitness function is $f(x)$, then according to Equation (3), the fitness function $F(x)$ of the population can be expressed as follows [29]:

$$F(X) = \begin{bmatrix} f([X_{1,1} & X_{1,2} & \cdots & X_{1,D}]) \\ f([X_{2,1} & X_{2,2} & \cdots & X_{2,D}]) \\ \vdots & \vdots & \vdots & \vdots \\ f([X_{n,1} & X_{n,2} & \cdots & X_{n,D}]) \end{bmatrix} \tag{4}$$

From the behavioral rules of the sparrow population, it can be seen that producers have higher energy to prioritize obtaining food, and have a wider search scope. The equation for updating the position of the producers in each iteration is [29]:

$$X_{i,d}^{k+1} = \begin{cases} X_{i,d}^{k} \cdot \exp(\frac{-i}{\alpha \cdot K}) & R_2 < ST \\ X_{i,d}^{k} + Q \cdot L & R_2 \geq ST \end{cases} \tag{5}$$

where $X_{i,d}^{k}$ represents the value of the d-dimensional variable of the ith individual in the kth iteration; α is a random number in [0,1]; R_2 is the alarm value with the range of [0,1]; ST is a warning threshold, with a range of values of [0.5,1]; K represents the maximum number of iterations; Q is a random number with a normal distribution in the range of [0,1]; L is a matrix with 1 row and D column; and all elements in the matrix are 1.

If $R_2 < ST$, it signifies that no predators have been found around and that the sparrows are in a safe environment. The producer can conduct extensive global searches. If $R_2 \geq ST$, it means that predators have appeared around the population. At this time, a sparrow sends an alarm signal to the population, and all individuals will go to a safe locality.

Because the scroungers have low energy, they will follow the footsteps of the discoverer and compete for food. The formula to update the position of scroungers is as follows [29]:

$$X_{i,d}^{k+1} = \begin{cases} Q \cdot \exp\left(\frac{X_{worst}^{k} - X_{i,d}^{k}}{i^2}\right) & i > \frac{n}{2} \\ X_{p}^{k+1} + \left|X_{i,d}^{k} - X_{p}^{k+1}\right| \cdot A^{+} \cdot L & 0 < i \leq \frac{n}{2} \end{cases} \tag{6}$$

where X_{p}^{k+1} represents the individual optimal position of the current discoverer; X_{worst}^{k} represents the current global worst position; n is the number of scroungers; A is a matrix with 1 row and D columns; and all its elements are randomly assigned to 1 or -1, $A^{+} = A^{T}\left(AA^{T}\right)^{-1}$.

While the individuals are looking for food, some of them will be on guard. In case of danger, they will go to the new location. In each iteration, SD individuals are randomly chosen for early warning behavior. These sparrows represent early warning individuals [29]:

$$X_{i,d}^{k+1} = \begin{cases} X_{best}^{k} + \beta \cdot \left|X_{i,d}^{k} - X_{best}^{k}\right| & f_i \neq f_g \\ X_{i,d}^{k} + P \cdot \frac{\left|X_{i,d}^{k} - X_{worst}^{k}\right|}{(f_i - f_w) + \varepsilon} & f_i = f_g \end{cases} \tag{7}$$

where X_{best}^{k} is the current globally optimal position; as the step size control parameter, β is a random number that follows a normal distribution with a mean of 0 and a variance of 1; P is a random number within the range of $[-1,1]$, where positive or negative represents the direction of sparrow movement, and size represents the step size control parameter; f_i is the fitness value corresponding to the current sparrow; f_g and f_w are the fitness values corresponding to sparrows in the optimal and worst positions, respectively; and ε is a smaller number to avoid the occurrence of 0 in the denominator.

2.3.2. Neural Network Prediction Model Based on Sparrow Search Algorithm

The flow of the neural network prediction model based on the SSA algorithm (SSA-NN) is shown in Figure 2.

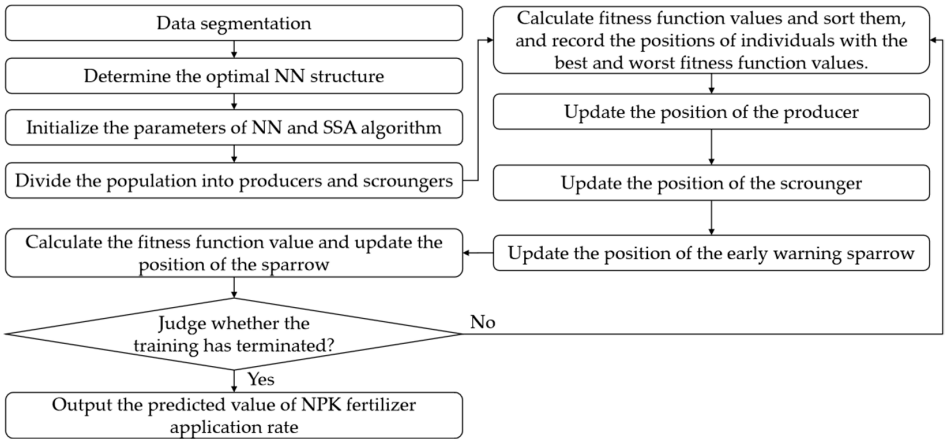


Figure 2. Flow chart of SSA-NN prediction model.

Step 1: Data segmentation. The dataset is divided into the training set (80%) and the test set (20%).

Step 2: Determine the optimal neural network structure.

Step 3: Initialize the parameters of the neural network and SSA algorithm.

Step 4: Divide the population into producers and scroungers.

Step 5: Calculate the fitness function values of the population, sort them, and record the individual positions with the best and worst fitness function values.

Step 6: Update the position of the producer based on Equation (5).

Step 7: Update the position of the scrounger based on Equation (6).

Step 8: Update the position of the early warning sparrow based on Equation (7).

Step 9: Calculate the fitness function value and update the position of the sparrow.

Step 10: Judge whether the maximum number of iterations K is reached: if so, the algorithm is terminated; otherwise, go to Step 5.

Step 11: The trained neural network prediction model is tested on the test samples.

2.4. Neural Network Prediction Model Based on Improved Sparrow Search Algorithm

To address the limitations of the SSA algorithm, such as insufficient population diversity, susceptibility to local optima, and inadequate accuracy in solving multidimensional functions, this section improves the SSA algorithm to enhance its global search capability and the capability to escape local optima. Finally, an improved SSA algorithm (ISSA) is proposed.

2.4.1. Improved Sparrow Search Algorithm

The ISSA algorithm is an improvement on the SSA algorithm, which adopts four measures: the population is initialized based on Arnold cat mapping chaos, it has an adaptive adjustment strategy for the number of producers and scroungers and an improved producer position update equation, and the position of the early warning sparrows is updated based on the Lévy flight strategy. The following four improvement measures are introduced:

(1) The population is initialized based on Arnold cat mapping chaos

When solving complex problems, the original SSA algorithm initializes the position of the sparrow population by randomly generating positions. This initialization method has problems, such as an insufficient population diversity and uneven distribution of sparrows in the search domain. To solve the above limitations, we chose to initialize the sparrow population by using a chaos operator, so that the position of the sparrow population is evenly distributed within the whole solution space.

Chaos is a nonlinear natural phenomenon, which has the advantages of ergodicity and randomness. At present, there are various chaotic maps to choose from in the field of optimization, including Logistic mapping, Tent mapping, and Arnold cat mapping [34]. The sequence of Logistic mapping is non-uniform: the sequence is characterized by a relatively uniform intermediate probability, but with a higher probability at both ends [35]. Compared with Logistic mapping, Tent mapping has superior ergodic uniformity and fast convergence speed; however, it often encounters issues such as a short cycle period and fixed point [36]. In contrast, Arnold cat mapping has a simple structure, it is more suitable for initializing the population, and ensures superior ergodic uniformity in the generated initial population [37]. Therefore, Arnold cat mapping was chosen to initialize the sparrow population in this study to improve the efficiency and accuracy of the algorithm.

Arnold Cat mapping is based on two-dimensional matrix transformation, and the expression is [37]:

$$\begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix} = \begin{bmatrix} 1 & a \\ b & ab+1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} \mathrm{mod} 1 \quad (8)$$

where a and b are arbitrary real numbers; mod1 is the decimal part of a.

According to the characteristics of the Arnold cat map, the method of generating a chaotic sequence in a feasible domain and initializing it with reverse solution is as follows:

Randomly generate a feasible solution of the current population, recorded as follows [37]:

$$\{\boldsymbol{X}_i = [x_{i1}, x_{i2}, \cdots, x_{id}, \cdots, x_{iD}]; x_{id} \in [l_{id}, u_{id}]\} \tag{9}$$

Then, the reverse solution is [37]:

$$\begin{cases} \boldsymbol{X}' = [\boldsymbol{X}'_1, \boldsymbol{X}'_2, \cdots, \boldsymbol{X}'_d, \cdots, \boldsymbol{X}'_D] \\ x'_{id} = q(l_{id} - u_{id}) - x_{id} \end{cases} \tag{10}$$

where q is a uniformly distributed real number on the interval [0,1], and l_{id} and u_{id} are the upper and lower bounds of the feasible solution, respectively.

(2) Adaptive adjustment strategy for the number of producers and scroungers

The constant ratio of producers and scroungers may lead to some problems in the operation of the algorithm. In the early iteration phase, the number of producers is limited, so it is impossible to fully search the overall situation. In the late iteration, the number of producers is relatively high; however, in fact, there is no need for more producers to conduct a global search; instead, the number of scroungers needs to be increased for accurate local searches.

In order to solve the above problem, we adopted the self-adaptive adjustment strategy for the number of producers and scroungers, which can comprehensively improve the optimization accuracy. At the beginning of iteration, the proportion of producers in the population is high. As the iteration progresses, the number of producers adaptively decreases, and the corresponding number of scroungers can adaptively increase, so that they can focus on a local accurate search. This adaptive strategy enables the algorithm to better meet to the optimization needs in different stages, so as to enhance the optimization accuracy. The adaptive adjustment equation for the number of producers and scroungers is as follows:

$$r = b(\tan(-\frac{\pi \cdot k}{4 \cdot K} + \frac{\pi}{4}) - c \cdot rand(0,1)) \tag{11}$$

$$P_{num} = r \cdot N \tag{12}$$

$$S_{num} = (1 - r) \cdot N \tag{13}$$

where P_{num} is the number of producers; S_{num} is the number of scroungers; b denotes the proportional coefficient, which is used to control the number between producers and scroungers; c denotes the disturbance deviation factor, which disturbs the nonlinear decreasing r value; k denotes the current iteration number; K denotes the maximum number of iterations; and N denotes the population of sparrows.

(3) Improved producer position update equation

In the SSA algorithm, the update of the position of the producer is only affected by the position of the previous generation producer, and the value of $\exp(\frac{-i}{\alpha \cdot K})$ will decrease adaptively in the iterative process. When its value is large, the producer is in global search mode. As the iteration progresses, its value continuously decreases, and the producer gradually adopts the local search mode, i.e., conducts a deep exploration near the optimal solution. In summary, the value of $\exp(\frac{-i}{\alpha \cdot K})$ has a significant impact on the performance of the algorithm. Therefore, according to Equation (5), this study improved the producer position update equation:

$$X_{i,d}^{k+1} = \begin{cases} X_{i,d}^{k} \cdot \frac{2}{\exp(\frac{4i}{\alpha \cdot K})^m} & R_2 < ST \\ X_{i,d}^{k} + Q \cdot L & R_2 \geq ST \end{cases} \tag{14}$$

The advantage of the improved equation is that the producer can conduct a comprehensive global search during the initial phase and is more inclined to explore the optimal position in later stages, so as to enhance the overall search performance of the algorithm.

(4) Update the position of the early warning based on the Lévy flight strategy

Lévy flight is the random walking system that has the ability to make large jumps in local positions, which plays an important role in the development of intelligent optimization algorithms [38]. Many researchers have found Lévy flight patterns in animals, such as albatrosses [39], seabirds [40], and even human behavior [41]. Based on the above characteristics of Lévy flight, researchers have conducted extensive research on it [42–44]. So far, Lévy flight has been widely used in various intelligent optimization algorithms to solve optimization problems, such as the cuckoo search algorithm [45], particle swarm optimization algorithm [46], and differential evolution algorithm [47]. In view of the characteristics of Lévy flight and its wide application, we chose Lévy flight to update the position of the early warning.

The Lévy flight position update equation is [43]:

$$x_i^{(k+1)} = x_i^{(k)} + \alpha \oplus Levy(\lambda) \quad i = 1, 2, \cdots, n \tag{15}$$

where $x_i^{(k)}$ represents the kth generation position of x_i; $\oplus$ is point-to-point multiplication; α represents the step size scaling factor; and $Levy(\lambda)$ is a random search path, which satisfies [43]:

$$Levy \sim \mu = k^{-\lambda} \quad 1 < \lambda \leq 3 \tag{16}$$

Lévy flight is essentially a random step size, and its step size conforms to the Lévy distribution. However, the Lévy distribution is very complex and cannot be realized. At present, the Mantegna algorithm stands as a prevalent method for simulating its flight trajectory; its mathematical expression is as follows:

The step size s is calculated as follows [43]:

$$s = \frac{\mu}{|\nu|^{\frac{1}{\beta}}} \tag{17}$$

where μ and ν are normal distributions, and the obeying equations are as follows [43]:

$$\mu \sim N(0, \sigma_\mu^2) \tag{18}$$

$$\nu \sim N(0, \sigma_\nu^2) \tag{19}$$

Standard deviations σ_μ and σ_ν satisfy [43]:

$$\sigma_\mu = \left\{ \frac{\Gamma(1+\beta) \cdot \sin(\frac{\pi\beta}{2})}{\Gamma(\frac{1+\beta}{2}) \cdot \beta \cdot 2^{\frac{\beta-1}{2}}} \right\}^{\frac{1}{\beta}} \tag{20}$$

$$\sigma_\nu = 1 \tag{21}$$

where $\Gamma(\cdot)$ is the gamma function, and β is usually 1.5.

Therefore, the Lévy flight position update equation can be expressed as:

$$x_i^{(k+1)} = x_i^{(k)} + \alpha \oplus s \quad i = 1, 2, \cdots, n \tag{22}$$

$$\alpha = 0.01 \cdot (x_i^{(k)} - x_{best}) \tag{23}$$

Lévy flight can generate diverse random step sizes. In the search process, a larger step size can expand the global search range, thus reducing the search accuracy and increasing the possibility of producing an unstable vibration status. However, the smaller step size

can enhance the depth optimization capability and the search accuracy. Lévy flight was applied to the ISSA algorithm to generate a random step size, which is helpful to escape from the local optima, expand the search range, and improve the optimization accuracy.

2.4.2. Neural Network Prediction Model Based on Improved Sparrow Search Algorithm

The flow of the neural network prediction model based on the ISSA algorithm (ISSA-NN) is shown in Figure 3.

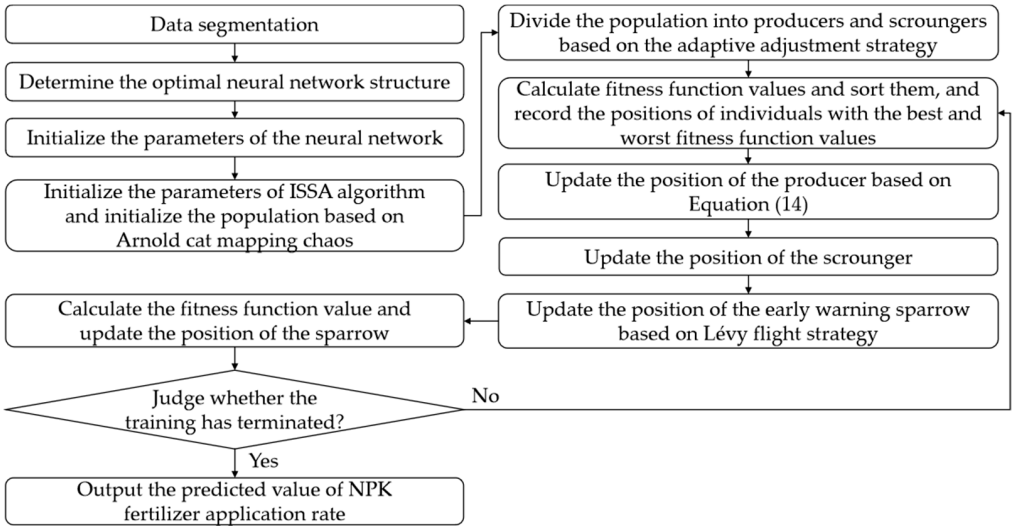


Figure 3. Flow chart of ISSA-NN prediction model.

Step 1: Data segmentation. The dataset is divided into the training set (80%) and the test set (20%).

Step 2: Determine the optimal neural network structure.

Step 3: Initialize the parameters of the neural network.

Step 4: Initialize the parameters of the ISSA algorithm and initialize the population based on Arnold cat mapping chaos.

Step 5: Divide the population into producers and scroungers based on Equations (11)–(13).

Step 6: Calculate the fitness function values of the population, sort them, and record the individual positions with the best and worst fitness function values.

Step 7: Update the position of the producer based on Equation (14).

Step 8: Update the position of the scrounger.

Step 9: Update the position of the early warning sparrow based on the Lévy flight strategy.

Step 10: Calculate the fitness function value and update the position of the individuals.

Step 11: Judge whether the maximum number of iterations K is reached: if so, the algorithm is terminated; otherwise, go to Step 6.

Step 12: The trained neural network prediction model is tested on the test samples.

2.5. Neural Network Prediction Model Based on Hybrid Algorithm

Although the SSA algorithm has been improved many times in the ISSA algorithm, the algorithm still has some limitations, such as a single search method, its ease of falling into the local optimal value, and so on. Therefore, in this study, the GA algorithm and ACO algorithm were introduced into the ISSA algorithm, so that the three algorithms can operate in parallel, and after each iteration, the best result was selected for the next iteration to improve the global search ability and the ability to jump out of the local optimum.

2.5.1. Genetic Algorithm

The GA algorithm is an excellent optimization algorithm for solving practical problems by simulating the process of biological genetic evolution in nature [22]. The GA

algorithm has the advantages of high efficiency, parallelism, and extensiveness, and is suitable for solving various types of practical problems. At present, it is widely used in engineering design, signal processing, production scheduling, and other fields to solve various optimization problems [48–51].

In nature, the chromosomes of biological individuals form a new individual through cross-recombination and mutation, thus realizing biological evolution. Similarly, the GA algorithm simulates the process of biological evolution in solving practical problems, conducts a probabilistic search in each genetic iteration, selects and retains the corresponding optimal individual, and then recombines genes through selection [52], crossover [53], mutation [54], and other operations to reconstitute a new population and pass it to the next generation, constantly cycling the above operations. When the individual meets the termination conditions required by the problem, the genetic process is stopped and the optimal solution of the actual problem is output [55].

The advantage of the GA algorithm is that it has wide applicability and can solve all kinds of optimization problems, especially complexity problems with multi-parameters and multi-objectives. It can perform global optimization in large-scale parameter space, has good robustness and parallelism, and is easily combined with other optimization methods. The flow of the neural network prediction model based on the GA algorithm (GA-NN) is shown in Figure 4.

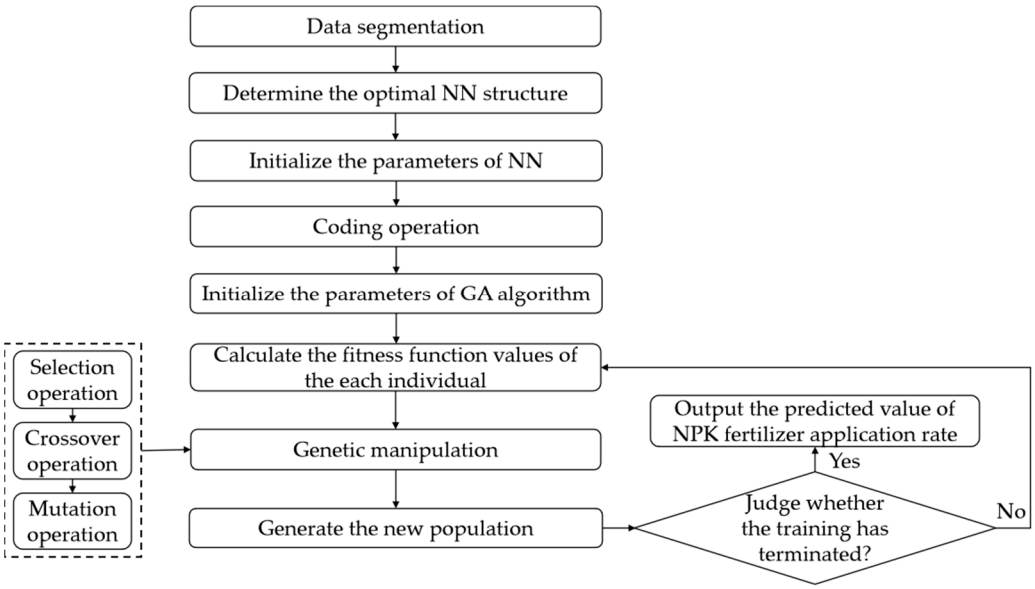


Figure 4. Flow chart of GA-NN prediction model.

Step 1: Data segmentation. The dataset is divided into the training set (80%) and the test set (20%).

Step 2: Determine the optimal neural network structure.

Step 3: Initialize the parameters of the neural network.

Step 4: Binary coding operation of the actual problem.

Step 5: Initialize the parameters of the GA algorithm.

Step 6: Calculate the fitness function values of each individual.

Step 7: A certain number of individuals are selected by the sorting selection method.

Step 8: Perform a single point crossover operation on the selected individual.

Step 9: On the basis of the crossover operation, the basic bit mutation operation is adopted.

Step 10: Generate the new population.

Step 11: Judge whether the maximum number of iterations K is reached: if so, the algorithm is terminated; otherwise, go to Step 6.

Step 12: The trained neural network prediction model is tested on the test samples.

2.5.2. Ant Colony Optimization Algorithm

The ACO algorithm is a probabilistic optimization method originating from ants finding the best path [23]. The ACO algorithm possesses notable strengths, such as robust global optimization capabilities, easy expansion, and parallelization. It has gained widespread applications in various domains, including robot path planning, the traveling salesman problem, task allocation, combinatorial optimization, and other fields [56–59].

The ACO algorithm obtains the global optimal solution by releasing pheromones on the path of the feasible solution. Ants search the path according to the pheromones. The shorter the path, the greater the amount of pheromones released by ants. With the increasing pheromone concentration on a short path, the more ants are attracted to choose this path. In conclusion, through the positive feedback mechanism, the colony can focus on the best path, i.e., the best solution to the problem to be optimized [60].

The ACO algorithm is essentially a parallel method, i.e., each ant searches for its own path. The algorithm primarily consists of the following steps:

(1) Initial pheromone

In the ACO algorithm, pheromones are represented by the symbol τ. The subscripts i and j represent the pheromone between position i and position j, and the right script 0 indicates that this is the first calculation, i.e., the initial pheromone. The initial pheromone is generally set to 1, or a smaller constant, indicating that the pheromones on each path are equal, i.e., the probability of ants climbing to each position is equal. The expression of the initial pheromone is [23]:

$$\tau_{ij}(0) = C \tag{24}$$

(2) Path selection

In the process of the path search, each ant has the opportunity to choose a path. Ants can move between positions based on the pheromone on the path and heuristic information. Therefore, the expression of the state transition probability p_{ij}^{m} of the mth ant from position i to position j at t time is as follows [23]:

$$p_{ij}^{m}(t) = \begin{cases} \dfrac{[\tau_{ij}(t)]^{\alpha} \cdot [\eta_{ij}(t)]^{\beta}}{\sum_{s \in allowed_m} [\tau_{is}(t)]^{\alpha} \cdot [\eta_{is}(t)]^{\beta}} & j \in allowed_m \\ 0 & otherwise \end{cases} \tag{25}$$

$$\eta_{ij}(t) = \frac{1}{d_{ij}(t)} \tag{26}$$

where $\tau_{ij}(t)$ represents the concentration of pheromones from position i to position j at t time; $\eta_{ij}(t)$ represents the heuristic function, which is the expected degree of ants from position i to position j at t time; $d_{ij}(t)$ represents the distance from position i to position j at t time, and adopts the Euclidean distance calculation method; α is an information heuristic factor, indicating the relative significance of the pheromone concentration, and its range of values is [0,5]; β is an expected heuristic factor, which signifies the relative significance of heuristic information in directing the colony search, and its range of values is [0,5]; $allowed_m$ represents the set of nodes where the ant can choose the next position j from the current position i; and s is the rest positions excepting position i.

(3) Pheromone update

According to the analysis of Equation (25), it can be seen that, if the volatilization of pheromones is not considered, the pheromone concentration of the relatively short path increases too fast with the increase in time, i.e., $\tau_{ij}(t)$ is too large, which can lead to the heuristic function $\eta_{ij}(t)$ having a diminishing effect on ant path selection and may lead to the loss of authenticity in solving the problem. Therefore, when the ant takes a step or completes a cycle after n times, it will update the pheromone concentration of the current path by using the pheromone update rule, which will make the next generation of ants

choose the path with a high pheromone concentration. The update rules for the pheromone concentration are [23]:

$$\tau_{ij}(t+n) = (1-\rho) \cdot \tau_{ij}(t) + \Delta\tau_{ij}(t) \tag{27}$$

$$\Delta\tau_{ij}(t) = \sum_{m=1}^{M} \Delta\tau_{ij}^{m}(t) \tag{28}$$

where ρ denotes a volatilization coefficient of pheromones on the path, the general range of which is $0 < \rho < 1$ to prevent the infinite accumulation of information; $1 - \rho$ denotes a pheromone coefficient saved after the end of the cycle; $\Delta\tau_{ij}(t)$ denotes an increment of pheromone from position i to position j at t time; $\Delta\tau_{ij}^{m}(t)$ denotes an increment of pheromone left by the mth ant from position i to position j at t time; and M denotes the total number of ants.

Due to the different calculation methods of $\Delta\tau_{ij}^{m}(t)$, the pheromone update methods are also different. As a researcher of the ACO algorithm, Dr. Dorigo M of Italy proposed three pheromone concentration updating models [61], i.e., the Ant-Cycle model, Ant-Quantity model, and Ant-Density model. The Ant-Cycle model is a global update method, and the principle is to calculate the pheromone concentration released by using the overall information of the path passed by ants (the total length of the path). The Ant-Cycle model is best in performance and is more widely used. Therefore, in this study, the Ant-Cycle model was used to update the pheromone concentration globally, and the expression of the Ant-Cycle model is as follows [61]:

$$\Delta\tau_{ij}^{m}(t) = \begin{cases} Q/L_m & \textit{m-th ant passes the path between positions i and j in a cycle} \\ 0 & \textit{otherwise} \end{cases} \tag{29}$$

where Q is a total number of pheromones, which is the positive constant, and its value has a certain influence on the convergence speed; and L_m is the total length of the path traveled by the mth ant during a cycle.

The flow of the neural network prediction model based on the ACO algorithm (ACO-NN) is shown in Figure 5.

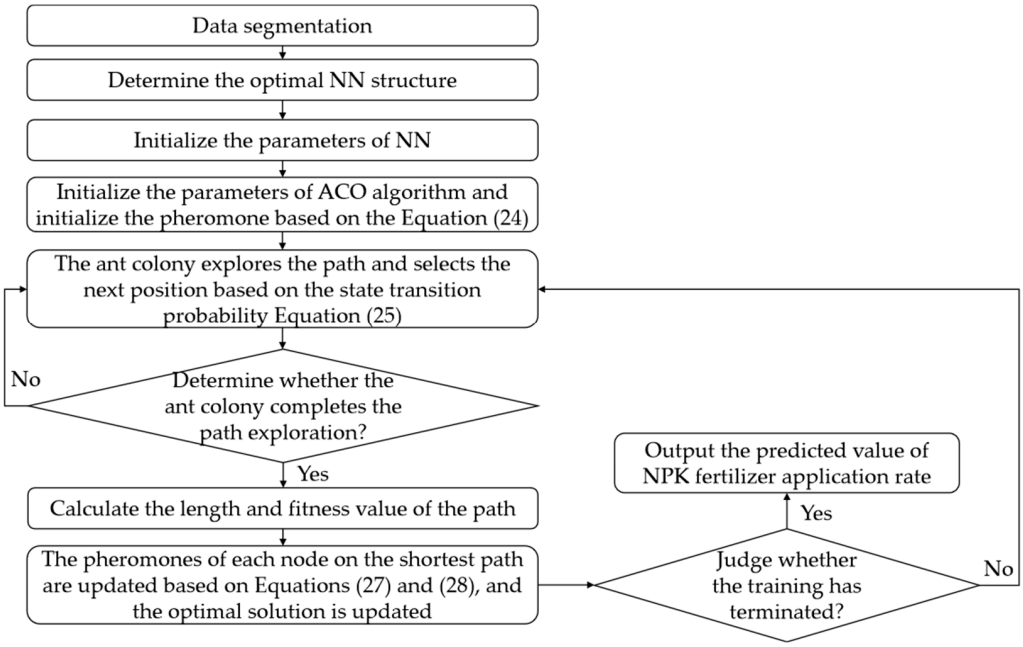


Figure 5. Flow chart of ACO-NN prediction model.

Step 1: Data segmentation. The dataset is divided into the training set (80%) and the test set (20%).

Step 2: Determine the optimal neural network structure.

Step 3: Initialize the parameters of the neural network.

Step 4: Initialize the parameters of the ACO algorithm and initialize the pheromone based on Equation (24).

Step 5: The ant colony explores the path and selects the next position based on the state transition probability Equation (25).

Step 6: Determine whether the ant colony completes the path exploration: if so, go to Step 7; otherwise, go to Step 5 until the ant colony completes the exploration.

Step 7: Calculate the length and fitness value of the path.

Step 8: The pheromones of each node on the shortest path are updated based on Equations (27) and (28), and the optimal solution is updated.

Step 9: Judge whether the maximum number of iterations K is reached: if so, the algorithm is terminated; otherwise, go to Step 5.

Step 10: The trained neural network prediction model is tested on the test samples.

2.5.3. Neural Network Prediction Model Based on Hybrid Algorithm

In fact, the HA algorithm was used to introduce the GA algorithm and the ACO algorithm into the ISSA algorithm, which makes the three algorithms operate in parallel. Following each iteration, the optimal result is selected for the next iteration, so as to enhance its global search capability and the capability to escape local optima.

The flow of the neural network prediction model based on the HA algorithm (HA-NN) is shown in Figure 6.

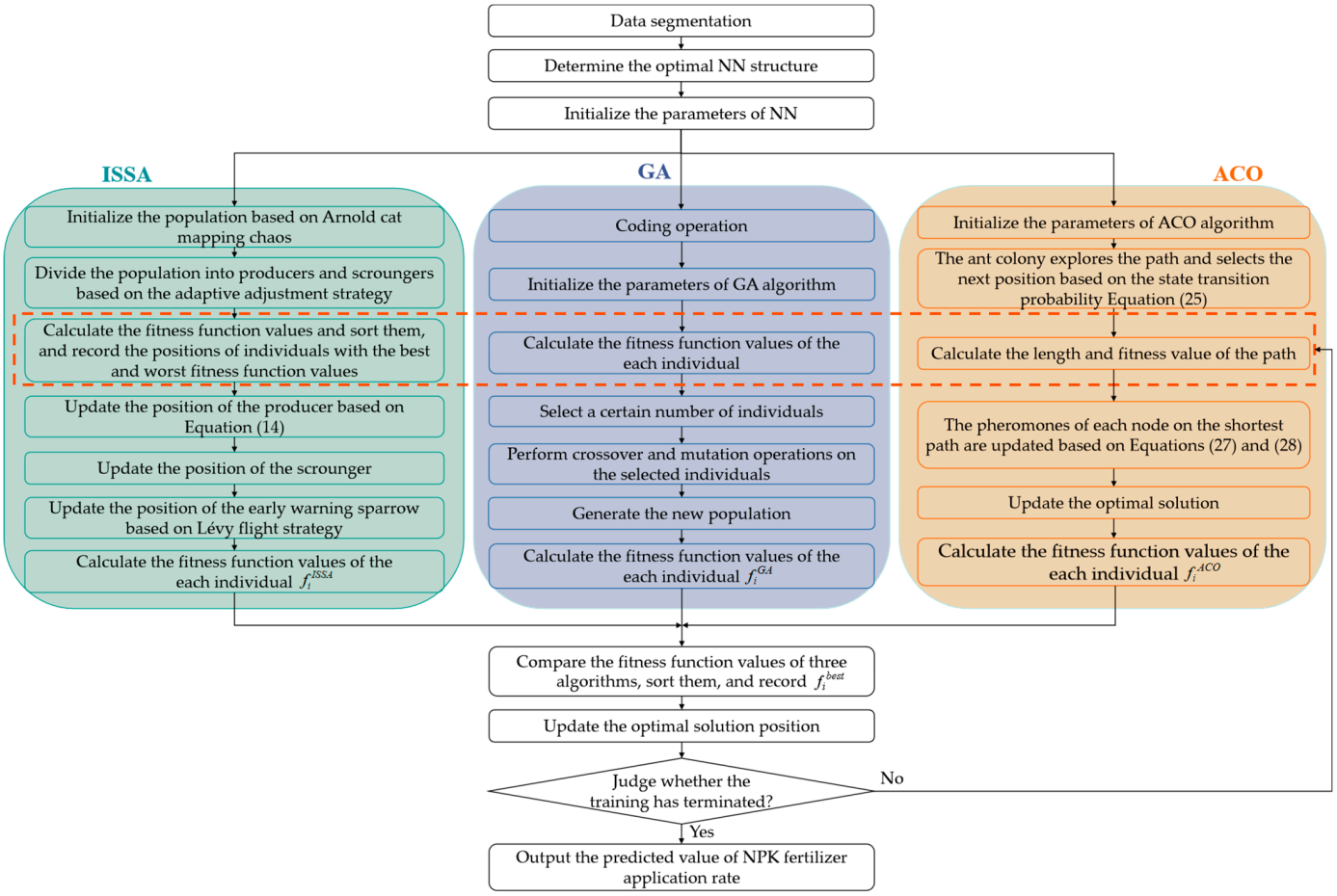


Figure 6. Flow chart of HA-NN prediction model.

Step 1: Data segmentation. The dataset is divided into the training set (80%) and the test set (20%).

Step 2: Determine the optimal neural network structure.

Step 3: Initialize the parameters of the neural network.

Step 4: The ISSA algorithm is executed, the population is initialized by Arnold cat mapping chaos, and the population is divided into producers and scroungers based on the adaptive adjustment strategy for the number of producers and scroungers. The population fitness function values are calculated and sorted, and the individual positions with the best and worst fitness function values are recorded. Update the positions of producers, scroungers, and early warning sparrows, and calculate the fitness function value f_i^{ISSA}.

Step 5: The GA algorithm is executed, a binary coding operation is performed on the practical problems, the GA algorithm parameters are initialized, and the fitness function value of each individual is calculated. Then, a certain number of individuals are selected by the sorting selection method, and the single point crossover and basic bit mutation operation are carried out to generate a new population and calculate the fitness function value f_i^{GA}.

Step 6: The ACO algorithm is executed, and the ant selects the path according to the pheromone concentration, the pheromone is updated, and the length of the path and the fitness function value f_i^{ACO} are calculated.

Step 7: The fitness function values of the three algorithms are compared and sorted, and the optimal fitness function value f_i^{best} is retained.

Step 8: Update the optimal solution position.

Step 9: Judge whether the maximum number of iterations K is reached: if so, the algorithm is terminated; otherwise, go to Step 4.

Step 10: The trained neural network prediction model is tested on the test samples.

2.6. Evaluation Indexes

In this study, the objective function is defined as the difference between the predicted values and actual values. To evaluate the output results of the three prediction models of the NPK fertilizer application rate, the mean square error (MSE), the explained variance score (EVS), and the coefficient of determination (R^2) were used to verify the effectiveness of the prediction models of the NPK fertilizer application rate for greenhouse tomatoes [62–64]. Their concepts and calculation methods are as follows:

(1) The expression of the objective function is as follows:

$$Fitness = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i) \tag{30}$$

where N is the number of samples; y_i is the actual data of the sample; and $\hat{y}_i$ is the predicted data of the model.

(2) The *MSE* is the mean value of the sum of the squares of the errors between predicted values and actual values, and its expression is as follows [62]:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{31}$$

(3) The *EVS* is an index used to evaluate the prediction accuracy of a regression model, and the *EVS* calculation method is as follows [63]:

$$EVS = 1 - \frac{\frac{1}{N} * \sum_{i=1}^{N}\left(\hat{y}_i - \frac{1}{N}\sum_{i=1}^{N} y_i\right)^2}{\frac{1}{N} * \sum_{i=1}^{N}\left(y_i - \frac{1}{N}\sum_{i=1}^{N} y_i\right)^2} = 1 - \frac{\sum_{i=1}^{N}\left(\hat{y}_i - \frac{1}{N}\sum_{i=1}^{N} y_i\right)^2}{\sum_{i=1}^{N}\left(y_i - \frac{1}{N}\sum_{i=1}^{N} y_i\right)^2} \tag{32}$$

The *EVS* value range is [0,1]. In practical applications, the closer the *EVS* value is to 1, the more data variance the model interprets, and the better the performance.

(4) R^2 refers to the coefficient of determination, and its expression is [64]:

$$R^2 = 1 - \frac{\sum_{i=1}^{N} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{N} (y_i - \overline{y})^2} = 1 - \frac{\sum_{i=1}^{N} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{N} (y_i - \frac{1}{N}\sum_{i=1}^{N} y_i)^2} \quad (33)$$

The R^2 value range is [0,1]; a larger R^2 indicates that the prediction model has a better effect on data fitting.

3. Results

3.1. Model Parameters

The three prediction models were constructed using the BP neural network. Based on the data source, the input layer of the neural network consists of six nodes, while the output layer comprises three nodes. So, a three-layer neural network structure was constructed as 6-*L*-3. According to the calculation method for hidden layer nodes, the possible range of L is from 3 to 13 [22]. After iterative training the neural network, the optimal number of hidden layer nodes was determined to be 10, i.e., the final topology chosen for the neural network was 6-10-3.

The parameters in the HA-NN prediction model were set as follows: sparrow population size $n = 30$; early warning ratio $SD = 0.1$; warning threshold $ST = 0.8$; disturbance deviation factor $c = 0.001$; population size of GA algorithm $n = 30$; crossover probability $P_c = 0.8$; mutation probability $P_m = 0.005$; ant colony size $n = 10$; information heuristic factor $\alpha = 2$; expected heuristic factor $\beta = 3$; volatilization coefficient of pheromones $\rho = 0.8$; maximum number of iterations $K = 100$.

3.2. Results and Analysis of Different Prediction Models under Low Soil Fertility

In this section, based on the data of low soil fertility (Table 2), three different prediction models are used to predict the NPK fertilizer application rate for greenhouse tomatoes. A total of 116 sets of data were selected as the training dataset for the models, and 30 sets of data were selected as the test dataset.

The comparison between the output values generated by the SSA–NN, ISSA–NN, and HA–NN prediction models, and the actual values are shown in Figure 7. To enable a more detailed analysis of the contrast between the predicted values and the actual values of the NPK fertilizer application rate, this study selected the comparative results of the first 10 groups of data.

Figure 7 shows the results of the prediction model of the NPK fertilizer application rate for greenhouse tomatoes under a low soil fertility. The high uncertainty of the test set, which comes from different data sources in different years and regions, presented a challenge. As shown in Figure 7, compared with the SSA-NN and ISSA-NN models, the HA-NN prediction model showed a higher prediction accuracy and lower error rate, and it was particularly excellent in predicting the greenhouse tomato NPK fertilizer application rate under low soil fertility. Through the comparative analysis of the prediction results of the model, it was found that the HA-NN prediction model made obvious progress in prediction accuracy. This achievement fully demonstrates the ability of the model to optimize parameters efficiently, and, at the same time, to significantly improve the performance level of the traditional SSA-NN prediction model.

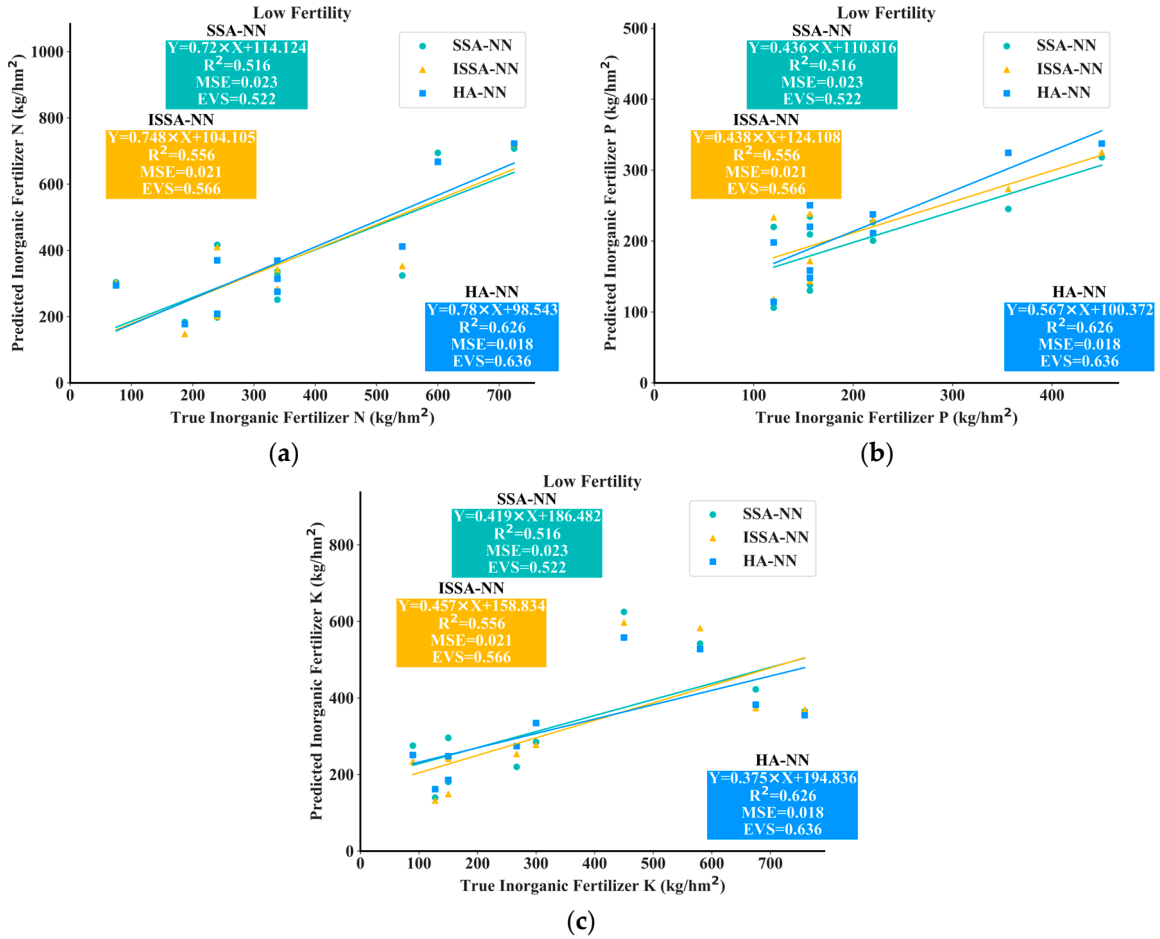


Figure 7. Comparison between predicted and actual values of prediction models under low soil fertility. (**a**) Inorganic fertilizer N; (**b**) inorganic fertilizer P; (**c**) inorganic fertilizer K.

To deeply explore the accuracy of the three forecasting models, this study used three evaluation indexes to comprehensively evaluate their forecasting performance. Through a detailed calculation according to Equations (31)–(33), the evaluation indicator results listed in Table 3 were obtained.

Table 3. Evaluation indicator data under low soil fertility.

	SSA–NN	ISSA–NN	HA–NN
MSE	0.023	0.021	0.018
EVS	0.522	0.566	0.636
R^2	0.516	0.556	0.626

Based on the definition of the evaluation index, it can be seen that the smaller the MSE value, the larger the EVS and R^2 values, which represents a better prediction effect of the prediction model. Compared with the evaluation indicator results in Table 3, the HA-NN prediction model had the lowest MSE value, and the largest EVS value and R^2 value, which showed that, compared with the SSA-NN and ISSA-NN prediction models, the HA-NN prediction model had a higher prediction accuracy and better prediction performance, and could better predict the NPK fertilizer application rate for greenhouse tomatoes under low soil fertility.

3.3. Results and Analysis of Different Prediction Models under Medium Soil Fertility

In this section, based on the data of medium soil fertility (Table 2), the three different prediction models are used to predict the NPK fertilizer application rate for greenhouse tomatoes. A total of 115 sets of data were selected as the training dataset for the models, and 29 sets of data were selected as the test datasets.

The comparison between the output values generated by the SSA–NN, ISSA–NN, and HA–NN prediction models, and the actual values are shown in Figure 8. To enable a more detailed analysis of the contrast between the predicted values and the actual values of the NPK fertilizer application rate, this study selected the comparative results of the first 10 groups of data.

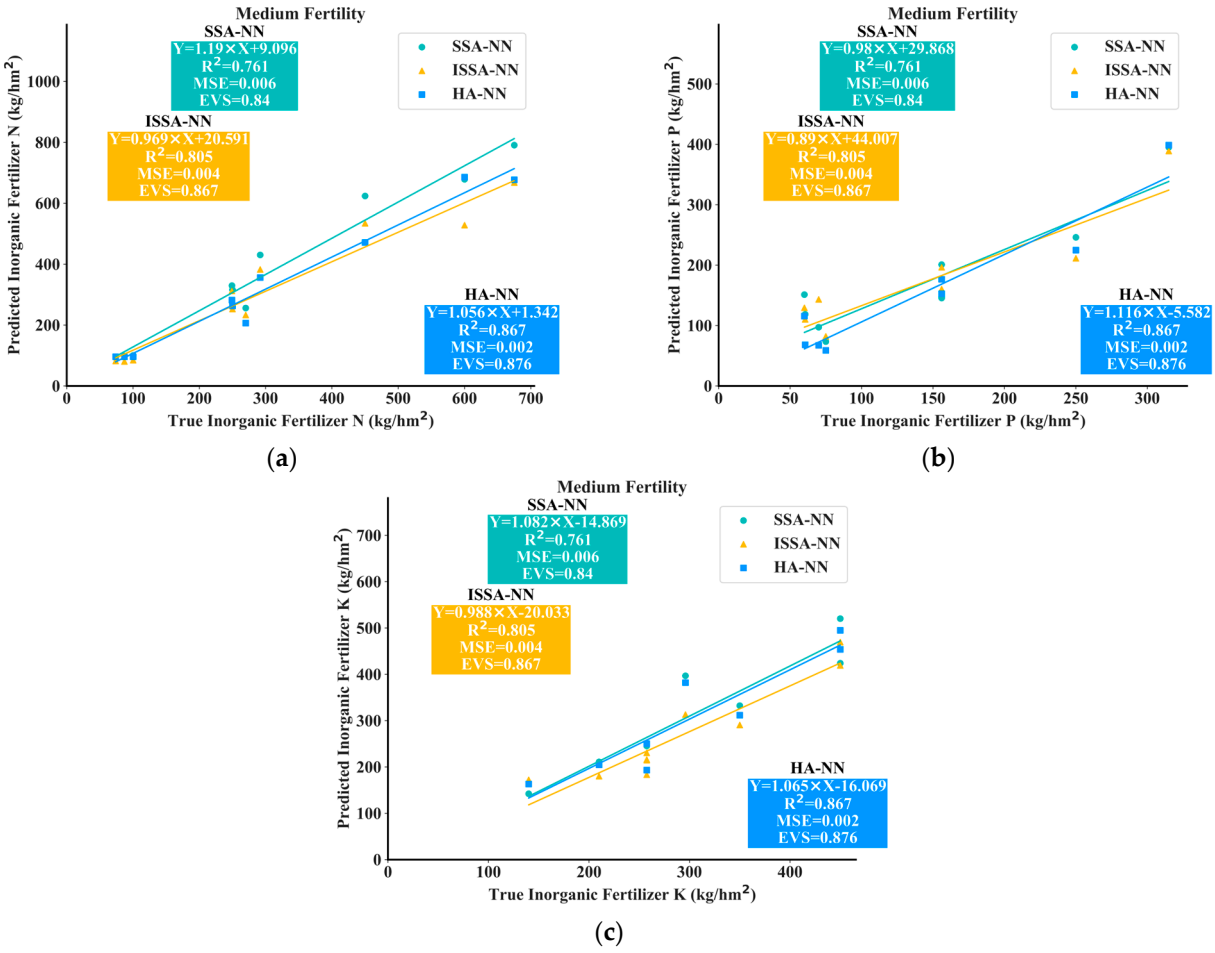


Figure 8. Comparison between predicted and actual values of prediction models under medium soil fertility. (**a**) Inorganic fertilizer N; (**b**) inorganic fertilizer P; (**c**) inorganic fertilizer K.

As shown in Figure 8, the prediction accuracy of the HA-NN prediction model proposed in this study was better than that of the SSA-NN and the ISSA-NN prediction models. The output value of the prediction model was closer to the actual value, indicating that it had a higher prediction accuracy and better prediction performance in predicting the greenhouse tomato NPK fertilizer application rate under medium soil fertility.

To deeply explore the accuracy of the three forecasting models, this study used three evaluation indexes to comprehensively evaluate their forecasting performance. Through a detailed calculation according to Equations (31)–(33), the evaluation indicator results listed in Table 4 were obtained.

Table 4. Evaluation indicator data under medium soil fertility.

	SSA–NN	ISSA–NN	HA–NN
MSE	0.006	0.004	0.002
EVS	0.840	0.867	0.876
R^2	0.761	0.805	0.867

Based on the analysis of the prediction performance of the three prediction models according to the characteristics of the evaluation indexes (MSE, EVS, and R^2), it was not difficult to find that all the evaluation indexes of the HA-NN prediction model proposed in this study were the best, which showed that, compared with the SSA-NN and ISSA-NN prediction models, the HA-NN prediction model had a higher prediction ability in predicting the greenhouse tomato the NPK fertilizer application rate under medium soil fertility. This, once again, proved that the HA-NN prediction model proposed in this study had a high accuracy in predicting the NPK fertilizer application rate for greenhouse tomatoes, and the predicted NPK fertilizer application rate for greenhouse tomatoes was closer to the actual values.

3.4. Results and Analysis of Different Prediction Models under High Soil Fertility

In this section, based on the data of high soil fertility (Table 2), the three different prediction models are used to predict the NPK fertilizer application rate for greenhouse tomatoes. A total of 80 sets of data were selected as the training dataset for the models, and 20 sets of data were selected as the test dataset.

The comparison between the output values generated by the SSA–NN, ISSA–NN, and HA–NN prediction models, and the actual values are shown in Figure 9. To enable a more detailed analysis of the contrast between the predicted values and the actual values of the NPK fertilizer application rate, this study selected the comparative results of the first 10 groups of data.

According to the comparison between the predicted output values and actual values in Figure 9, it can be seen that, compared with the SSA-NN and ISA-NN prediction models, the HA-NN prediction model proposed in this study had a higher prediction accuracy and could better approximate the actual values. The simulation results showed that the HA-NN prediction model had an excellent performance in predicting the greenhouse tomato NPK fertilizer application rate under high soil fertility and had a better prediction performance.

To deeply explore the accuracy of the three forecasting models, this study used three evaluation indexes to comprehensively evaluate their forecasting performance. Through a detailed calculation according to Equations (31)–(33), the evaluation indicator results listed in Table 5 were obtained.

Table 5. Evaluation indicator data under high soil fertility.

	SSA–NN	ISSA–NN	HA–NN
MSE	0.008	0.007	0.005
EVS	0.861	0.871	0.908
R^2	0.839	0.862	0.899

From the evaluation index data in Table 5, it can be seen that the MSE value gradually decreased, moving closer to 0. In addition, the EVS and R^2 values gradually increased, and were closer to 1, which meant that, compared with the SSA-NN and the ISSA-NN prediction models, the HA-NN prediction model proposed had the best prediction accuracy, fully reflecting the good prediction performance of the HA-NN prediction model in predicting the greenhouse tomato NPK fertilizer application rate under high soil fertility.

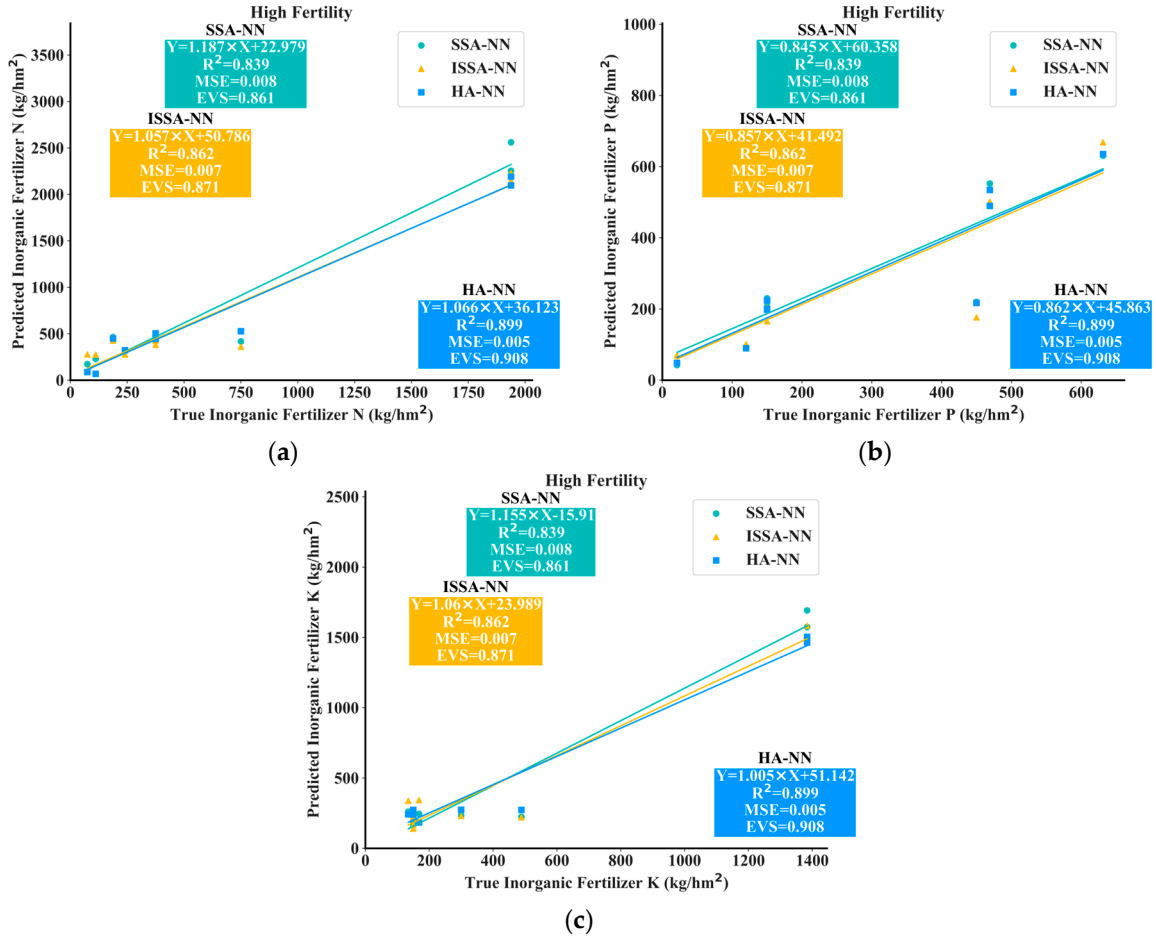


Figure 9. Comparison between predicted and actual values of prediction models under high soil fertility. (**a**) Inorganic fertilizer N; (**b**) inorganic fertilizer P; (**c**) inorganic fertilizer K.

4. Discussion

Facility tomato production is an important part of modern agriculture and is widely used around the world. Compared with traditional open-field cultivation, facility tomato production has obvious advantages, such as a controllable environment, an improved land utilization rate and output rate, and an enhanced comprehensive production capacity. However, there are still some problems in this production mode, such as the low nutrient utilization rate, inaccurate fertilization, and unbalanced soil ecology. Precision fertilization aims at improving the crop yield and quality and reducing resource waste and environmental pollution through scientific and reasonable fertilization methods. It is a challenging task to determine the optimal amount of fertilizer applied to crops. This is mainly due to the complicated and nonlinear relationship between crop fertilization and soil fertility. Traditional fertilization methods, such as the nutrient balance method or fertilizer effect function method, can guide fertilization to a certain extent, but it is often difficult to accurately describe this complex relationship.

The emergence of neural network methods provides a solution to this problem. The precision fertilization model based on a neural network integrates big data, machine learning methods, and optimization algorithms, and constructs a nonlinear mapping relationship between crop growth and fertilization amount, thus improving the accuracy and practicability of fertilization. Based on a neural network, three forecasting models of the NPK fertilizer application rate for greenhouse tomatoes under low, medium, and high soil fertility conditions were constructed, and the predicted values of the models were compared with actual values. As can be seen from Figures 7–9, the HA-NN prediction model had a superior prediction accuracy, which successfully proved that the HA-NN

prediction model proposed in this study had a good ability to predict the NPK fertilizer application rate for greenhouse tomatoes under any soil fertility conditions. To facilitate a more intuitive evaluation of the prediction performance among the three forecasting models, this study used the MSE, EVS, and R^2 as evaluation indicators, and conducted a deep comparison and analysis of the forecasting results of the three models. As shown in Figures 10–12, these evaluation indicators comprehensively reflect the prediction effect of each model. Based on the definitions of the evaluation indicators, it is evident that a lower MSE and higher EVS and R^2 indicate more optimal evaluation indicators, signifying the superior prediction performance of the forecasting model.

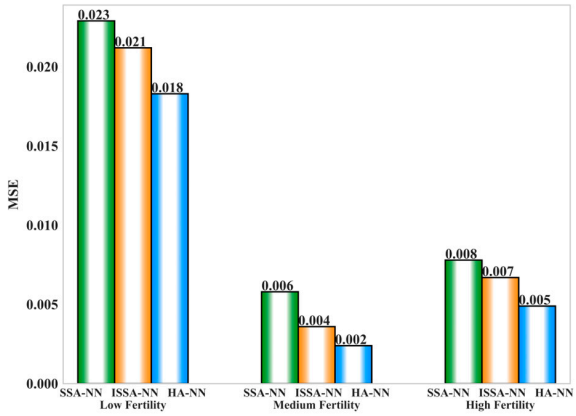


Figure 10. Comparison of the MSE evaluation indicator for the three prediction models.

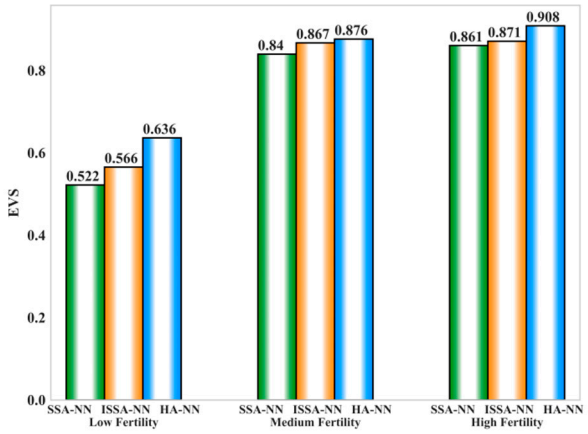


Figure 11. Comparison of the EVS evaluation indicator for the three prediction models.

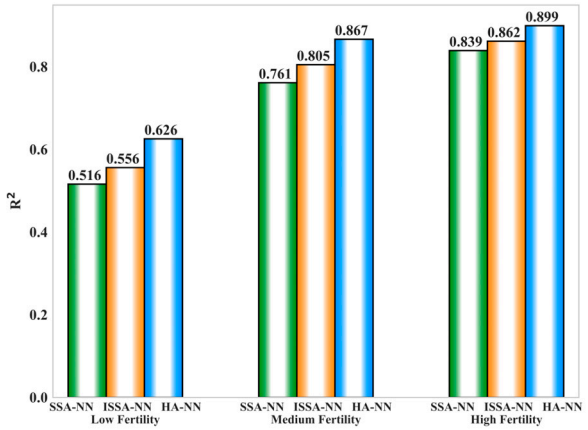


Figure 12. Comparison of the R^2 evaluation indicator for the three prediction models.

It can be clearly seen from Figures 10–12 that the MSE of the SSA-NN, ISSA-NN, and HA-NN prediction models showed a gradual downward trend, while the EVS and R^2 showed an upward trend under low, medium, and high soil fertility conditions. This showed that the evaluation indicators of the HA-NN forecasting model were the optimal values under any soil fertility condition, i.e., the HA-NN forecasting model had the lowest MSE and the highest EVS and R^2.

In related research, Zhao proposed a fertilization model combining fuzzy C-means clustering and an RBF neural network, and the model evaluation index, i.e., the MSE, was 0.2 [15], while Dong constructed a maize fertilization method based on a wavelet-BP neural network, and the model evaluation index, i.e., the root-mean-square error (RMSE), was 200 [65]. By comparison, the MSE evaluation indicator values for the HA-NN prediction model proposed in this study under low, medium, and high soil fertility conditions were 0.018, 0.002, and 0.005, respectively. This fully proved the superiority of the HA-NN forecasting model in predicting the NPK fertilizer application rate for greenhouse tomatoes, and showed the rationality and feasibility of applying the HA-NN forecasting model to predict the NPK fertilizer application rate for greenhouse tomatoes. According to the prediction results of the HA-NN forecasting model, effective guidance can be provided for nutrient and production management in facility tomato production.

5. Conclusions

The greenhouse environment, the soil properties, the physiological characteristics of the crops, and the target yield are all closely related to the NPK fertilizer application rate for greenhouse tomatoes. To achieve an accurate prediction of the NPK fertilizer application rate for greenhouse tomatoes, this study utilized the significant advantages of neural network models in dealing with nonlinear relationships and multivariable predictions. We chose the neural network model as the basic model and propose the ISSA-NN and HA-NN prediction models to solve the problem associated with the ordinary SSA algorithm, i.e., it easily falls into the local optimal value, which further improved the prediction accuracy and stability. The primary conclusions of this study can be summarized as follows:

1. Based on the grading standard of the soil nutrient index for tomato production in solar greenhouses, experimental data collected in different years and regions were classified into low, medium, and high soil fertility datasets;
2. To enhance the prediction accuracy of the NPK fertilizer application rate for greenhouse tomatoes, first, the ISSA algorithm was used to improve the SSA algorithm in four aspects, so that it could jump out of local optima in a timely manner. Then, the ISSA algorithm, GA algorithm, and ACO algorithm were operated in parallel, and the HA algorithm was proposed to enhance the global search capability and the capability to escape local optima;
3. Under the low, medium, and high soil fertility conditions, the SSA-NN prediction model, the ISSA-NN prediction model, and the HA-NN prediction model were used to predict the NPK fertilizer application rate for greenhouse tomatoes, and the prediction results were evaluated by the MSE, EVS, and R^2. The experimental results showed that the evaluation indexes of the HA-NN prediction model were superior to those of the SSA-NN prediction model and ISA-NN prediction model. The HA-NN prediction model had a higher prediction accuracy and better fitting ability, and could more accurately predict the NPK fertilizer application rate for greenhouse tomatoes.

To sum up, the HA-NN prediction model proposed in this study could provide reliable prediction results for the NPK fertilizer application rate for greenhouse tomatoes and had a high reliability in predicting the NPK fertilizer application rate for greenhouse tomatoes. This research will help agricultural producers to make decisions on nutrient management and production management in the process of tomato production in solar greenhouses and provide a reference for the prediction of the fertilizer application rate for other crops. The development and validation of this model were based on soil fertility data from different regions in China, so its prediction results have a high practical application

value in agricultural production in China. The application of the model is expected to provide a scientific fertilizer management scheme for greenhouse tomato planting in China and promote the sustainable development of greenhouse agriculture. However, there are still many problems to be further studied, such as how to obtain more data sources to enhance the prediction accuracy and generalizability of the model, how to improve existing optimization methods to enhance the optimization performance of the model, and whether we can find a more efficient objective function to elevate the precision of the fertilizer application prediction model. These are our future research directions.

Author Contributions: Conceptualization, X.Y., Y.L. and X.P.; methodology, X.Y. and X.P.; data curation, B.B.; writing–original draft preparation, X.Y. and X.P.; writing–review and editing, X.C., C.L. and X.P.; visualization, X.Y. and Y.L.; supervision, X.C. and C.L.; funding acquisition, X.C. and X.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Key R&D Program of China (grant numbers 2021YFD1500100), the National Natural Science Foundation of China (grant numbers 42177005, 41877098, and 41671290), the Science and Technology Program of Shenyang (22-317-2-01), the China Postdoctoral Science Foundation (2023M733676), and the Liaoning Province Applied Basic Research Program (Youth Special Project) (2023020288-JH2/1016).

Data Availability Statement: Publicly available datasets were analyzed in this study. The data sources are indicated in the text.

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

References

1. Chen, X.; Jiang, Z.; Tai, Q.; Shen, C.; Rao, Y. Construction of a photosynthetic rate prediction model for greenhouse strawberries with distributed regulation of light environment. *Math. Biosci. Eng.* **2022**, *19*, 12774–12791. [CrossRef] [PubMed]
2. Gao, M.; Wu, Q.; Li, J.; Wang, B.; Zhou, Z. Temperature prediction of solar greenhouse based on NARX regression neural network. *Sci. Rep.* **2023**, *13*, 1563. [CrossRef] [PubMed]
3. Jung, D.; Kim, H.S.; Jhin, C.; Kim, H.; Park, S.H. Time-serial analysis of deep neural network models for prediction of climatic conditions inside a greenhouse. *Comput. Electron. Agric.* **2020**, *173*, 105402. [CrossRef]
4. Havlin, J.; Heiniger, R. Soil Fertility Management for Better Crop Production. *Agronomy* **2020**, *10*, 1349. [CrossRef]
5. Srivastava, A.; Wu, Q.; Mousavi, S.; Hota, D. Integrated Soil Fertility Management in Fruit Crops: An Overview. *Int. J. Fruit Sci.* **2021**, *21*, 413–439. [CrossRef]
6. Amirahmadi, E.; Ghorbani, M.; Moudrý, J.; Konvalina, P.; Kopecký, M. Impacts of Environmental Factors and Nutrients Management on Tomato Grown under Controlled and Open Field Conditions. *Agronomy* **2023**, *13*, 916. [CrossRef]
7. Sulaiman, S.; Sadiq, S. Influence of greenhouse shading and different nutrient management practices on alleviating heat stress, improving plant nutrients status, flowering growth and yield of tomato. *Iraqi J. Agric. Sci.* **2020**, *51*, 1001–1014.
8. Shabani, A.; Ghaffary, K.A.; Sepaskhah, A.R.; Kamgar-Haghighi, A.A. Using the artificial neural network to estimate leaf area. *Sci. Hortic.* **2017**, *216*, 103–110. [CrossRef]
9. Salazar, R.; López, I.; Rojano, A.; Schmidt, U.; Dannehl, D. Tomato yield prediction in a semi-closed greenhouse. *Acta Hortic.* **2015**, *1107*, 263–270. [CrossRef]
10. Hara, P.; Piekutowska, M.; Niedbała, G. Selection of Independent Variables for Crop Yield Prediction Using Artificial Neural Network Models with Remote Sensing Data. *Land* **2021**, *10*, 609. [CrossRef]
11. Zhuang, L.; Liu, H.; Zhu, J.; Wang, S.; Song, Y. Comparison of forecasting methods for power system short-term load forecasting based on neural networks. In Proceedings of the 2016 IEEE International Conference on Information and Automation (ICIA), Ningbo, China, 1–3 August 2016.
12. Ehret, D.L.; Hill, B.D.; Helmer, T.; Edwards, D.R. Neural network modeling of greenhouse tomato yield, growth and water use from automated crop monitoring data. *Comput. Electron. Agric.* **2011**, *79*, 82–89. [CrossRef]
13. Gopal, P.; Bhargavi, R. A novel approach for efficient crop yield prediction. *Comput. Electron. Agric.* **2019**, *165*, 104968. [CrossRef]
14. Khaoula, A.; Gabsi, K.; Mercatoris, B.; Khemis, C.; Amami, R. Prediction of organic potato yield using tillage systems and soil properties by artificial neural network (ANN) and multiple linear regressions (MLR). *Soil Tillage Res.* **2019**, *190*, 202–208.
15. Zhao, S.; Chen, G.; Fu, S.; Xiao, E. Study on precision fertilization model based on fusion algorithm of cluster and RBF neural network. In Proceedings of the 11th International Conference on Computer and Computing Technologies in Agriculture (CCTA), Jilin, China, 12–15 August 2017.
16. Alhnaity, B.; Pearson, S.; Leontidis, G.; Kollias, S. Using Deep Learning to Predict Plant Growth and Yield in Greenhouse Environments. *Acta Hortic.* **2020**, *1296*, 425–431. [CrossRef]
17. Wang, Y.; Xiao, R.; Yin, Y.; Liu, T. Prediction of Tomato Yield in Chinese-Style Solar Greenhouses Based on Wavelet Neural Networks and Genetic Algorithms. *Information* **2021**, *12*, 336. [CrossRef]

18. Belouz, K.; Nourani, A.; Zereg, S.; Bencheikh, A. Prediction of greenhouse tomato yield using artificial neural networks combined with sensitivity analysis. *Sci. Hortic.* **2022**, *293*, 110666. [CrossRef]
19. Xue, J.; Shen, B. A novel swarm intelligence optimization approach: Sparrow search algorithm. *Syst. Sci. Control Eng.* **2020**, *8*, 22–34. [CrossRef]
20. Yan, S.; Liu, W.; Li, X.; Yang, P.; Wu, F.; Yan, Z. Comparative Study and Improvement Analysis of Sparrow Search Algorithm. *Wirel. Commun. Mob. Comput.* **2022**, *2022*, 4882521. [CrossRef]
21. Yue, Y.; Cao, L.; Lu, D.; Hu, Z.; Xu, M.; Wang, S.; Li, B.; Ding, H. Review and empirical analysis of sparrow search algorithm. *Artif. Intell. Rev.* **2023**, *56*, 10867–10919. [CrossRef]
22. Zhai, L.; Feng, S. A novel evacuation path planning method based on improved genetic algorithm. *J. Intell. Fuzzy Syst.* **2022**, *42*, 1813–1823. [CrossRef]
23. Chen, A.; Tan, H.; Zhu, Y. Ant colony optimization algorithm and its application. In Proceedings of the 2nd International Conference on Applied Mathematics, Modelling, and Intelligent Computing (CAMMIC), Kunming, China, 25–27 March 2022.
24. Peng, X.; Yu, X.; Luo, Y.; Chang, Y.; Lu, C.; Chen, X. Prediction Model of Greenhouse Tomato Yield Using Data Based on Different Soil Fertility Conditions. *Agronomy* **2023**, *13*, 1892. [CrossRef]
25. Xiao, Z.; Yan, Z. Radar Emitter Identification Based on Feedforward Neural Networks. In Proceedings of the 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Chongqing, China, 12–14 June 2020.
26. Sun, X. Kinematics model identification and motion control of robot based on fast learning neural network. *J. Ambient Intell. Humaniz. Comput.* **2020**, *11*, 6145–6154. [CrossRef]
27. Zheng, Y.; Xue, X.; Zhang, J. Research on Fault Diagnosis of Hydraulic System of Fast Erecting Device Based on Fuzzy Neural Network. *Int. J. Fluid Power* **2022**, *23*, 141–160. [CrossRef]
28. Feng, W.; Feng, F. Research on the Multimodal Digital Teaching Quality Data Evaluation Model Based on Fuzzy BP Neural Network. *Comput. Intell. Neurosci.* **2022**, *2022*, 7893792. [CrossRef] [PubMed]
29. Wu, D.; Yuan, C. Threshold image segmentation based on improved sparrow search algorithm. *Multimed. Tools Appl.* **2022**, *81*, 33513–33546. [CrossRef] [PubMed]
30. Zhang, Z.; He, R.; Yang, K. A bioinspired path planning approach for mobile robots based on improved sparrow search algorithm. *Adv. Manuf.* **2022**, *10*, 114–130. [CrossRef]
31. Li, X.; Gu, J.; Sun, X.; Li, J.; Tang, S. Parameter identification of robot manipulators with unknown payloads using an improved chaotic sparrow search algorithm. *Appl. Intell.* **2022**, *52*, 10341–10351. [CrossRef]
32. Yao, Y.; Liao, H.; Liu, M.; Yang, X. Coverage Optimization Strategy for 3-D Wireless Sensor Networks Based on Improved Sparrow Search Algorithm. *IEEE Sens. J.* **2023**, *23*, 23721–23733. [CrossRef]
33. Zhang, C.; Chen, P.; Jiang, F.; Xie, J.; Yu, T. Fault Diagnosis of Nuclear Power Plant Based on Sparrow Search Algorithm Optimized CNN-LSTM Neural Network. *Energies* **2023**, *16*, 2934. [CrossRef]
34. Jia, J.; Han, Z.; Liu, L. A Novel Nonlinear Chaotic Mapping Based on Cosine-Exponent. In Proceedings of the 2022 IEEE 2nd International Conference on Data Science and Computer Application (ICDSCA), Dalian, China, 28–30 October 2022.
35. Guo, Y.; Yang, J.; Liu, B. Application of Chaotic Encryption Algorithm Based on Variable Parameters in RFID Security. *EURASIP. J. Wirel. Commun. Netw.* **2020**, *1*, 155. [CrossRef]
36. Chen, R.; Wang, S.Y. An optimization method for an integrated energy system scheduling process based on NSGA-II improved by tent mapping chaotic algorithms. *Processes* **2020**, *8*, 426. [CrossRef]
37. Giri, K.; Dadheech, P.; Gupta, A.K. A Efficient Approach for Securing Images Using RSA and Arnold Technique Metamorphose. In Proceedings of the 2nd International Conference on Information Management and Machine Intelligence (ICIMMI), Jaipur, India, 24–25 July 2020.
38. He, Q.; Liu, H.; Ding, G.; Tu, L. A modified Lévy flight distribution for solving high-dimensional numerical optimization problems. *Math. Comput. Simul.* **2023**, *204*, 376–400. [CrossRef]
39. Reynolds, A.M. Olfactory search behaviour in the wandering albatross is predicted to give rise to Lévy flight movement patterns. *Anim. Behav.* **2012**, *83*, 1225–1229. [CrossRef]
40. Reynolds, A.M.; Paiva, V.H.; Cecere, J.G.; Focardi, S. Lévy patterns in seabirds are multifaceted describing both spatial and temporal patterning. *Front. Zool.* **2016**, *13*, 29. [CrossRef] [PubMed]
41. Baronchelli, A.; Radicchi, F. Lévy flights in human behavior and cognition. *Chaos Solitons Fractals* **2013**, *56*, 101–105. [CrossRef]
42. Lenin, K. Levy Flight-Based White Wolf Algorithm for Solving Optimal Reactive Power Problem. In Proceedings of the International Conference on Advanced Communication and Computational Technology (ICACCT), Kurukshetra, India, 6–7 December 2019.
43. Nasri, D.; Mokeddem, D. Optimisation of multi-objective problems using an efficient Levy flight grasshopper algorithm. *Int. J. High Perform. Syst. Archit.* **2022**, *11*, 26–35. [CrossRef]
44. Li, J.; An, Q.; Lei, H.; Deng, Q.; Wang, G. Survey of Lévy Flight-Based Metaheuristics for Optimization. *Mathematics* **2022**, *10*, 2785. [CrossRef]
45. Panda, S.K.; Ray, P.; Mishra, D.P.; Salkuti, S.R. Short-term load forecasting of the distribution system using cuckoo search algorithm. *Int. J. Power Electron. Drive Syst.* **2022**, *13*, 159–166. [CrossRef]

46. Liu, X.; Wang, G.; Wang, L. Lsfqpso: Quantum particle swarm optimization with optimal guided Lévy flight and straight flight for solving optimization problems. *Eng. Comput.* **2021**, *38*, 4651–4682. [CrossRef]
47. Wang, W.; Xu, L.; Chau, K.; Liu, C.; Ma, Q.; Xu, D. Cε-LDE: A lightweight variant of differential evolution algorithm with combined ε constrained method and Lévy flight for constrained optimization problems. *Expert Syst. Appl.* **2023**, *211*, 118644. [CrossRef]
48. Li, L.; Sun, L.; Guo, J.; Li, S.; Jiang, P. Identification of crop diseases based on improved genetic algorithm and extreme learning machine. *Comput. Mater. Contin.* **2020**, *65*, 761–775. [CrossRef]
49. Luo, X.; Qian, Q.; Fu, Y.F. Improved genetic algorithm for solving flexible job shop scheduling problem. *Procedia Comput. Sci.* **2020**, *166*, 480–485. [CrossRef]
50. Li, W.; Cheng, L.; Hu, J. Research on stealthy UAV path planning based on improved genetic algorithm. In Proceedings of the 2022 International Conference on Artificial Intelligence and Computer Information Technology (AICIT), Yichang, China, 16–18 September 2022.
51. Wang, S.; Zhang, J.; Liu, M.; Liu, B.; Wang, J.; Yang, S. Large-Signal Behavior Modeling of GaN P-HEMT Based on GA-ELM Neural Network. *Circuits Syst. Signal Process.* **2022**, *41*, 1834–1847. [CrossRef]
52. Shang, L.; Shang, Y.; Hu, L.; Li, J. Performance of genetic algorithms with different selection operators for solving short-term optimized reservoir scheduling problem. *Soft Comput.* **2020**, *24*, 6771–6785. [CrossRef]
53. Misevičius, A.; Verenė, D. A Hybrid Genetic-Hierarchical Algorithm for the Quadratic Assignment Problem. *Entropy* **2021**, *23*, 108. [CrossRef] [PubMed]
54. Song, Y.; Yan, F. An Improved Mutation Operator Which Can Improve the Performance of Genetic Algorithm. In Proceedings of the 2021 2nd International Conference on Artificial Intelligence and Computer Engineering (ICAICE), Hangzhou, China, 5–7 November 2021.
55. Lei, L.; Liu, N.; Zhou, J. Nonlinear Function Optimization Based on Adaptive Genetic Algorithm. In Proceedings of the 4th International Conference on Computer Science and Application Engineering (CSAE), Virtual, Online, 20–22 October 2020.
56. Miao, C.; Chen, G.; Yan, C.; Wu, Y. Path planning optimization of indoor mobile robot based on adaptive ant colony algorithm. *Comput. Ind. Eng.* **2021**, *156*, 107230. [CrossRef]
57. Yang, K.; You, X.; Liu, S.; Pan, H. A novel ant colony optimization based on game for traveling salesman problem. *Appl. Intell.* **2020**, *50*, 4529–4542. [CrossRef]
58. Cao, R.; Li, S.; Ji, Y.; Zhang, Z.; Xu, H.; Zhang, M. Task assignment of multiple agricultural machinery cooperation based on improved ant colony algorithm. *Comput. Electron. Agric.* **2021**, *182*, 105993. [CrossRef]
59. Shi, M.; Xiao, S.; Feng, X. An Adaptive Ant Colony Algorithm Based on Local Information Entropy to Solve Distributed Constraint Optimization Problems. *Intern. J. Pattern Recognit. Artif. Intell.* **2023**, *37*, 2359013. [CrossRef]
60. Shi, Z.; Kumar, R.; Tomar, R. Multi-Objective Optimization of Smart Grid Based on Ant Colony Algorithm. *Electrica* **2022**, *22*, 395–402. [CrossRef]
61. Dorigo, M.; Maniezzo, V.; Colorni, A. Ant system: Optimization by a colony of cooperating agents. *IEEE Trans. Syst. Man Cybern. B Cybern.* **1996**, *26*, 29–41. [CrossRef] [PubMed]
62. Upadhyay, S.; Kapoor, S.R.; Choudhary, R. Fault Classification and Detection in Transmission Lines Using ANN. In Proceedings of the 2018 International Conference on Inventive Research in Computing Applications (ICIRCA), Coimbatore, India, 11–12 July 2018.
63. Nivitha, K.; Madhavan, V.; Megha, V.; Dinesh Kumar, S.; Jothi Lakshmi, S. An Ensemble Approach for Flight Delay Prediction Through Spatiotemporal Parameters. In Proceedings of the 2023 9th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 17–18 March 2023.
64. Zhang, D.; Chu, Y.; Ma, S.; Sheng, Z.; Chen, F. Three-dimensional elemental imaging of material surface using image-assisted laser-induced breakdown spectroscopy. *Appl. Surf. Sci.* **2020**, *534*, 147601. [CrossRef]
65. Dong, Y.; Fu, Z.; Peng, Y.; Zheng, Y.; Yan, H.; Li, X. Precision fertilization method of field crops based on the Wavelet-BP neural network in China. *J. Clean. Prod.* **2020**, *246*, 118735. [CrossRef]

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.