Prediction of twisting Machine Speed based on improved Particle Swarm Optimization algorithm

Yannian Wang¹, Weixun Zhai², Xiongfei Li³ and Zheng Zhong⁴

¹,²,³,⁴College of Electronics and Information, Xi’an Polytechnic University, Xi’an, Shaanxi, 710048, China
²Corresponding author’s e-mail: 423014262@qq.com

Abstract: In order to predict the conversion of process parameters to roller and spindle speed in the production of twisting machine in textile mill, the improved particle swarm optimization algorithm is used to optimize the prediction model based on BP neural network, the optimization algorithm is used to find the optimal solution of neural network weight and bias, and the BP neural network algorithm is used to predict the speed value of roller and spindle. The improved single neural network is easy to fall into the problem of local minimum and slow convergence speed. The experimental data show that the prediction data on BP neural network based on improved particle swarm optimization algorithm are accurate and the error is small. It can be seen that the prediction model can meet the needs of speed prediction in twisting machine production.

1. Introduction
The twisting machine is the main equipment used in the yarn production of the textile mill, and the rotating speed value setting of the front and back rollers, the core gauze, the hollow spindle and the ring spindle in the production of the traditional twisting machine is often based on the usual experience of the technical personnel. At the same time, the rotation speed value of the component is set in combination with the data on the known yarn product, and then the debugging setting is carried out according to the condition of the production small sample, and finally until the reciprocating circulation reaches the requirement. The traditional production mode speed setting efficiency is low, and not only the process is complicated, but also the test can not achieve the ideal effect after many tests, which is not beneficial to the development of the spinning enterprise.

For the complexity of the speed setting, it is difficult to establish an accurate mathematical model, and the efficiency of the measured method is low and the production efficiency is influenced, so the prediction algorithm is widely used in the estimation of the rotating speed. In this paper, the improved particle swarm algorithm is used to optimize the prediction model of the traditional neural network algorithm in recent years, and the disadvantages of the traditional algorithm are improved and the prediction model is optimized and improved by the improved particle swarm optimization algorithm[5].

2. Process analysis of twisting machine
Yarn consists of core yarn, fixed yarn, decorative yarn and so on. The process parameters, such as twist, overfeeding ratio, traction multiple and tension coefficient, are the main factors affecting the speed of roller and spindle of twisting machine. Twist refers to the measure of changing yarn structure by using the relative angular displacement between the cross sections of cotton yarn and tilting the original straight parallel fiber and yarn shaft; Overfeeding ratio is the ratio of finishing yarn speed to
core yarn speed in the production process. If the yarn is produced at a constant value, it is overfed yarn, and if the pattern is constantly changed and overfed, it is controlled yarn; the traction multiple is the ratio of the product of the feeding and the super-feeding ratio of the decorative yarn and the weight of the decorative yarn in the output yarn; Tension coefficient is adjusted by tensioner or roller, and tension coefficient is also a factor that directly affects yarn quality and pattern.

In this paper, a neural network speed prediction model of twisting machine is established by using four process parameters to update the weights and biases continuously after many times of training. 120 sets of data are used as the original data set, and ten different kinds of yarns are used for prediction model training.

3. BP neural network based on particle swarm optimization

3.1 BP neural network
BP-neural network\(^2\) is a learning algorithm that emphasizes the backward propagation of error in the network. It has strong self-adaptation and generalization ability, and is especially suitable for solving the complex problem of internal mechanism. Its network structure is shown in figure 1. In recent years, the BP neural network algorithm is used in the prediction of the rotational speed of the twisting machine, but the BP algorithm is a local search method, and in the speed prediction of the twisting machine, the global extreme value of the complex non-linear function is solved. Therefore, the algorithm is likely to fall into the local extreme value, resulting in a large deviation of the training output result. At the same time, the optimal convergence rate of weight\(^3\) and bias\(^4\) in the traditional algorithm will gradually slow down in the course of network training, this will continue to prolong the training time of the prediction model. Therefore, the BP neural network algorithm needs to be combined with other artificial intelligence algorithms to effectively improve the performance of the algorithm. In this paper, an improved particle swarm algorithm is proposed to optimize the disadvantages of the traditional neural network algorithm.

$$\begin{align*}
\text{Input layer} & \rightarrow \text{Hidden layer} \\
& \rightarrow \text{Output layer}
\end{align*}$$

Figure 1. Schematic diagram of neural network structure.

3.2 Improvement Particle Swarm optimization
Particle swarm optimization (Particle Swarm Optimization) is an intelligent global random search algorithm proposed by simulating the migration and clustering behavior of birds in the process of foraging. In this algorithm, each particle can be regarded as a search individual in the multidimensional search space, and the current position of the particle is a solution to the corresponding problem. Particles have only two properties: speed and position, speed represents the speed and speed of particle movement, and position represents the direction of particle movement. The optimal solution searched by each particle is called individual extreme value, and the optimal individual extreme value in particle swarm optimization is regarded as the current global optimal
solution. Finally, the optimal solution satisfying the termination condition is obtained by continuous iteration, updating speed and position.

The modified particle swarm optimization algorithm is used to dynamically adjust the original inertia weight and the learning factor of the particle swarm algorithm to further improve the global search performance, convergence performance and stability of the particles. Dynamic adjustment can improve the population search space that is shrinking in the iteration so that the particles can be searched in a wider range. The improved algorithm is not easy to fall into the local optimal value while maintaining the diversity of the population, and finally improves the possibility of finding the best weight and bias of the algorithm.

As shown in figure 2, this paper makes full use of the global search characteristics of the improved particle swarm optimization algorithm to improve the neural network, and preserves the individual with the highest fitness function by the particle position corresponding to the weight and bias of the neural network, so as to achieve the purpose of optimizing the network model. The fusion of the two algorithms can overcome the dependence of BP network on the initial value and effectively improve the performance and convergence speed of the network.

![Optimization of BP neural network by IPSO algorithm](image)

Figure 2. A flow chart of improved particle swarm optimization for BP neural network.

4. Speed prediction model for twisting machine

4.1 Multi-layer neural network establishment

1) The neural network prediction model of twister is established, and the network structure is set as
input layer, hidden layer and output layer. Using different twist machine process parameters as the input layer of BP network, the data set is twist, overfeed ratio, tension coefficient and traction multiple. The input layer is \( I_1, I_2, I_3, I_4 \). The rotation speed values such as front roller, core yarn roller, rear roller, hollow spindle and ring spindle are used as the network output layer, corresponding to the expected output value of \( O_1, O_2, O_3, O_4, O_5 \). General formula for selecting the number of nodes according to the hidden layer:

\[ l = \sqrt{j + k + z} \tag{1} \]

Wherein \( j \) and \( k \) are the number of neurons in the input layer and the output layer, \( z \) is a constant between 0 and 10, and the expected convergence effect can be achieved when the number of the hidden layer nodes is 9 through repeated experiments.

2) The weight values of the input layer of the three-layer network of the three-layer network of the prediction model of the fancy twisting machine to the hidden layer and the hidden layer to the output layer are \( W_{ab} \) and \( W_{bc} \), the offset of the hidden layer is set to \( \theta_b \). The offset of the output layer is set to \( \Phi_4 \). Initialize the weight and bias, where \( a \) is the output layer unit number, \( b \) is the hidden layer, and \( c \) is the output layer.

### 4.2 Improved particle swarm optimization

1) The population size, the number of iteration, the learning factor and the inertia weight of the particle population and its population size, the number of iteration, the learning factor and the inertia weight are initialized, and the position and the speed are taken at random in the defined area. Generate a particle swarm consisting of \( n \) particles \( X = (X_1, X_2, \cdots, X_i, \cdots, X_n) \), a search space for a given m-dimension in a population, and \( X_i = (x_{i1}, x_{i2}, \cdots, x_{im})^T \) represents the position of the first particle in m-dimensional space.

\[ m = n \cdot j + j \cdot k + j + k \tag{2} \]

2) The weight and offset in the BP neural network correspond to the dimension components of the individual particles. In this way the weight and bias are optimized by improving the particle swarm algorithm.

3) Let the velocity of the first particle be \( V_i = (V_{i1}, V_{i2}, \cdots, V_{im})^T \), \( \theta_i = (\theta_{i1}, \theta_{i2}, \cdots, \theta_{im})^T \) is the individual extremum of the i-th particle. In each optimization, \( \theta_g = (\theta_{g1}, \theta_{g2}, \cdots, \theta_{gm})^T \) is the global extreme value. Then the speed and position of the particles will be constantly refreshed according to the algorithm in each search, with the formula:

\[
V_{im}^\prime = w V_{im} + c_1 r_1 (\theta_{im} - x_{im}) + c_2 r_2 (\theta_{gm} - x_{im})
\]

\[
x_{im}^\prime = x_{im} + V_{im}^\prime
\]

Where \( w \) represents the inertia weight and , \( V_{im}^\prime \) and \( x_{im}^\prime \) represent the speed and position of the particle after the update, \( c_1 \) and \( c_2 \) are acceleration factors greater than zero, \( r_1 \) and \( r_2 \) are random numbers of \([0,1]\). In order to limit the search range of the particles, the intervals of the speed and position are defined in \([-X_{max}, X_{min}],[v_{max}, v_{min}]\).

4) Improve the inertia weight. According to the strategy that the inertia weight decreases linearly according to the number of iterations, the inertia weight is recalculated.

\[
w(\lambda) = \frac{\lambda_{max} - \lambda}{\lambda_{max} \cdot (w_{max} - w_{min}) + w_{min}} \tag{4} \]

Where the \( \lambda \) is iterations, and \( w \) is inertia weight.

5) Improvement of dynamic factor, and the acceleration factors of \( c_1 \) and \( c_2 \) are improved to improve the convergence speed and accuracy of the algorithm.
\[
c_2 = c_{\text{max}} - (c_{\text{max}} - c_{\text{min}}) \frac{\lambda_{\text{max}} - \lambda}{\lambda_{\text{max}}}
\]

\[
c_1 = 4 - c_2
\]

\(c_{\text{max}}, c_{\text{min}}\) are the initial and final values of \(c_1\). \(0 < c_{\text{min}} < c_{\text{max}} \leq 4\).

6) Whether the weights and offsets corresponding to the velocity and position of particles meet the requirements, that is, the mean square error is used as the fitness function of particles. Then the particle fitness function is:

\[
F(t) = \frac{1}{N} \sum_{t=1}^{N} (\bar{y}(t) - y(t))^2
\]

where \(N\) is the number of samples in the PSO training, \(\bar{y}(t)\) is the expected output, and \(y(t)\) is the network training output.

7) Calculate the fitness values corresponding to each particle and the average population fitness values. When compared with the population, the state is invariant if it is less than the average fitness of the population, whereas the particle position is updated.

8) Recalculating the particle fitness continuously updates the individual extremum and global extremum, and finally finds the corresponding optimal weights and biases for output.

4.3 Rotation Speed Forecast of Twister

According to the setting of the multi-layer neural network and the optimization of the improved particle swarm algorithm, the optimal weight and the bias can be obtained, and the network training and the rotation speed prediction can be carried out. According to the setting, the output value of the hidden layer of the prediction model can be calculated:

\[
H_b = f \left( \sum_{i=1}^{d} W_{ab} I_a - \theta_b \right)
\]

\(H_b\) is the hidden layer output, where \(a\) is the output layer unit number, \(b\) is the hidden layer, and \(c\) is the output layer. The function \(f\) is the activation equation sigmoid function:

\[
f(x) = \frac{1}{1 + e^{-ax}}
\]

Calculate whether the error \(e\) between the predicted speed value and the actual output speed value \(e\) meets the demand. The learning rate is \([0,1]\) random value for the learning rate. If not, the updating weight and bias formula are as follows:

\[
W_{ab} = W_{ab} + \eta \times e_b \times I_a
\]

\[
W_{bc} = W_{bc} + \eta \times e_c \times H_b
\]

\[
\Phi_{bc} = \Phi_{bc} + \eta \Phi_c
\]

\[
\theta_b = \theta_b + \eta \theta_c
\]

If satisfied, then predict the output, and the speed prediction value \(C\) of the output layer is based on the nonlinear activation of the \(H_b\), hidden layer and the output layer weight \(W_{bc}\) and the output layer bias \(\Phi_c\) again. Where \(C\) is a collection of output layers \(O_1, O_2, O_3, O_4, O_5\), then the formula is:

\[
C = f \left( \sum_{i=1}^{b} W_{bc} H_b - \Phi_c \right)
\]

Under the condition that the precision requirement is finally met, the twist, overfeed ratio, tension coefficient and traction multiple of the twisting machine are combined with the optimal weight and bias in the neural network by \(I_1, I_2, I_3, I_4\), that is, the twist, overfeeding ratio, tension coefficient and traction multiple of the twisting machine which are \(W_{ab}, W_{bc}, \theta_b, \Phi_c\). Calculation by Formula 3-10,
We can get the predicted values $O_1$, $O_2$, $O_3$, $O_4$, $O_5$. That is, the speed value of the front roller, the core yarn, the back roller, the hollow spindle and the ring spindle of the twisting machine.

5. Experiments and Data Analysis

Table 1. Raw data set section data sheet.

| Process parameters | Yarn Type number |
|--------------------|------------------|
|                    | 1    | 2    | 3    | 4    | 5    | 6    | 7    | 8    | 9    | 10   |
| Yarn twist         | 319.28| 10.21| 53.571| 51.820| 51.821| 54.622| 58.451| 63.755| 71.713| 84.974|
| Over-feeding       | 1.3   | 1.6   | 2.1   | 2.4   | 2.4   | 3.6   | 3.1   | 2.9   | 3.3   | 3.6   |
| Tension coefficient| 1.05  | 1.05  | 1.05  | 1.05  | 1.06  | 1.06  | 1.07  | 1.07  | 1.07  | 1.08  |
| Traction multiple  | 3.2   | 3.5   | 3.8   | 4.3   | 4.3   | 4.7   | 4.0   | 5.0   | 5.5   | 5.8   |

Table 2. Normalization data table of process parameters and predicted values.

| Process parameters | Yarn Type number |
|--------------------|------------------|
|                    | 1    | 2    | 3    | 4    | 5    | 6    | 7    | 8    | 9    | 10   |
| Yarn twist         | 0.243 | 0.312 | 0.407 | 0.393 | 0.413 | 0.445 | 0.483 | 0.542 | 0.648 | 0.842 |
| Over-feeding       | 0.287 | 0.356 | 0.478 | 0.547 | 0.594 | 0.713 | 0.676 | 0.761 | 0.834 | 0.885 |
| Tension coefficient| 0.956 | 0.95  | 0.964 | 0.964 | 0.973 | 0.973 | 0.982 | 0.982 | 0.981 | 0.981 |
| Traction multiple  | 0.457 | 0.502 | 0.544 | 0.615 | 0.672 | 0.715 | 0.787 | 0.830 | 0.858 | 0.930 |
| Front roller speed | 0.916 | 1.000 | 0.954 | 0.878 | 0.834 | 0.852 | 0.668 | 0.612 | 0.507 | 0.354 |
| Core yarn rotation speed | 1.000 | 0.544 | 0.627 | 0.505 | 0.438 | 0.377 | 0.314 | 0.258 | 0.189 | 0.122 |
| Rear Rolla speed   | 1.000 | 1.000 | 0.878 | 0.714 | 0.625 | 0.604 | 0.426 | 0.348 | 0.294 | 0.192 |
| Hollow spindle speed| 1.0000 | 0.938 | 0.879 | 0.817 | 0.786 | 0.688 | 0.629 | 0.564 | 0.505 | 0.438 |
| Ring spindle speed  | 1.000 | 0.752 | 0.700 | 0.817 | 0.756 | 0.688 | 0.626 | 0.284 | 0.500 | 0.438 |

The data in Table 1 is used as the original data, and the data in Table 2 is the normalization of the input value and the predicted output value of Table 1. The training data are normalized to convert the experimental data all to the number between [0,1]. In order to increase the prediction accuracy and reduce the error, it can facilitate the operation of the data before improving the particle swarm optimization and neural network modeling training, and it is more suitable for the training of the neural network, and can make the convergence speed of the program faster when running. The formula is as follows:

$$x' = \frac{x - x_{\text{min}}}{x_{\text{max}} - x_{\text{min}}}$$  \hspace{1cm} (11)
Of which normalized converted values, \( x \) is the value before normalization, \( x_{\text{max}} \) and \( x_{\text{min}} \) are the maximum and minimum values for the dataset.

![Figure 3. Front roller Prediction Contrast Chart of Traditional Algorithm.](image1)

![Figure 4. Front roller Predictive Contrast Chart of Optimization Algorithm.](image2)

![Figure 5. Ring spindle prediction Contrast Chart of traditional algorithm.](image3)

![Figure 6. Ring spindle prediction Contrast Chart of Optimization Algorithm.](image4)

The experimental results are as follows: the speed of front roller and ring ingot after data normalization are taken as examples. Before figure 3 and figure 4, the prediction of the traditional roller algorithm is compared with that of the previous roller optimization algorithm. figure 5 and figure 6 compare the prediction of ring spindle optimization algorithm with the prediction of ring ingot traditional algorithm. The deviation between the predicted value and the real value of BP neural network can be obtained by comparing the data with the image, and after the improved particle swarm optimization algorithm proposed in this paper optimizes the weight and bias of the neural network, the predicted value error is smaller and more accurate, and the fitting degree with the actual speed value is high.

6. Conclusion
This paper introduces the twisting machine technology and puts forward the speed prediction model, and analyzes the traditional BP neural network speed prediction model. The improved particle swarm algorithm is proposed to optimize the weight and bias of BP neural network and to improve the convergence speed of the whole prediction system and avoid the problem of BP neural network falling
into the local minimum value. Through the analysis and comparison of the experimental data, the advantages of the optimization algorithm over the traditional algorithm can be demonstrated. The experimental results show that the algorithm has the characteristics of high accuracy and fast convergence speed.

Project funds:
1. Science and Technology Plan Project of Xi'an Science and Technology Bureau “Design and Research of Textile Factory Management and Equipment Control System Based on Intelligent Manufacturing”, No: 201805030YD8CG14 (1).
2. Shaanxi Provincial Science and Technology Department's general project in the industrial field “Research on Key Technologies of Intelligent Air-conditioning Control System in Textile Mills”, No: 2019GY-109.
3. Keqiao Textile Industry Innovation Research Institute Project of Xi'an Engineering University, No: 19KQYB02.

References
[1] Huang W., Wang J. (2017) Effect and Analysis of Composite Yarn Structure on Strength Utilization of Single Yarn. Journal of Textiles, 38(2):99-105.
[2] Wang L., Li X.H. (2019) Research on distortion correction of BP neural network based on improved genetic simulated annealing algorithm. Computer Measurement and Control, 27(5):77-81.
[3] Li S.J., Zhao C., Su X.H., et al. (2017) Optimization of PID controller parameters for control of temperature and humidity in environmental test module. Journal of Nanjing University of Technology, (004).
[4] Huang D.Z., Gong R.X., Gong S. (2015) Prediction of Wind Power by Chaos and BP Artificial Neural Networks Approach Based on Genetic Algorithm. Journal of electrical engineering and technology, 10(1):41-46.
[5] Song G.F., Liang C.Y., Liang Y., et al. (2014) Improved genetic algorithm for optimizing daily passenger flow prediction in tourist attractions with BP neural network. Minicomputer System, 35(9):2136-2141.