Abstract—Financial market dynamics forecasting has long been a focus of economic research. A hybridizing functional link artificial neural network (FLANN) and improved particle warm optimization (PSO) based on wavelet mutation (WM), named as IWM-PSO-FLANN, for forecasting the CSI 300 index is proposed in this paper. In the training model, it expands a wider mutation range while apply wavelet theory to the PSO, in order to exploring the solution space more effectively for better parameter solution. In the stimulating experiment, we use five benchmark functions to test the proposed method, and the results shows that IWM-PSO has greater convergence accuracy than WM-PSO and PSO. The empirical research is performed in testing the predictive effects of CSI 300 index in the proposed model compared with the back propagation functional link neural network (BP-FLANN), PSO-FLANN and WM-PSO-FLANN. The experiment utilizes two expansion functions, Chebyshev functions and trigonometric functions, to map the input data to higher dimension. The results show that the prediction performance of the proposed model displays a better performance in financial time series forecasting than other three models. Moreover, the accuracy of the input with trigonometric functions is higher, and it suggests that trigonometric function is more suitable for this kind of data type.

Keywords—Stock index; Forecasting; Functional link artificial neural network (FLANN); Improved wavelet mutation (IWM); Particle warm optimization (PSO)

I. INTRODUCTION

The CSI 300 index is composed of the 300 largest and most liquid A-shares listed on the two stock exchanges, launched on April 8th, 2005, and it aims to reflect the price fluctuation and performance. The CSI index has been used as the basis for many financial products around the world and is also used by investors to develop and benchmark their portfolios [1].

Predicting the stock market index is of prime importance in private and institution investors when making investment decision because successful prediction of stock prices may be guaranteed benefits. Their main question is to forecast, to determine the appropriate time to buy, hold or sell. The investors assume that the future trend of the stock market is based at least in part on present and past events and data. However, the price variation of stock market is a dynamic system and the disordered behavior of the stock price movement duplicates complication of the price prediction and the highly non-linear, dynamic complicated domain knowledge inherent in the stock market makes it very difficult for investors to make the right investment decisions. Many researchers in the past have applied various computing techniques to predict the movement of the stock markets.

Artificial neural network have gained its popularity due to their inherent capability to approximate any none-linear function, less sensitive to error, tolerate noise, and chaotic components. To improve predicting precision, various network architectures and learning algorithms have been developed in the papers [2-4].

Further it is well known that the artificial neural network (ANN) suffers from local minima trapping, saturation, weight interference, initial weight independence and over fitting, make ANN training difficult. Additionally, it is also very difficult to fix the parameters like number of neurons in a layer, and number of hidden layers in a network, thereby deciding a proper architecture is not that easy. Thus to overcome from this functional link artificial neural network (FLANN) based simple network may be used for the prediction the stock index data with less computational overhead than the ANN [5]. FLANN is proposed by Y.H.PAO and Y. Takefji [6] contains a single layer neural network in which nonlinearity is introduced as a functional block, thus giving rise to higher dimension input space. In FLANN, higher-order correlations among input components can be used to construct a higher –order network to perform non– linear mapping using only a single layer of units.

The researchers proposed different FLANN models to predict the stock market indices, such as DJIA, S&P 500 stock indices[7,8] Indian stocks[9,10], and there is no research about the CSI 300 index.

Particle swarm optimization (PSO) is a population-based stochastic optimization algorithm which inspired by the social behaviors of animals like fish schooling and bird flocking proposed by Kennedy J and Eberhart R [11]. The PSO has comparable or even superior search performance for many hard optimization problems with faster and more stable convergence rate. The PSO has been widely used in parameter learning of neural network [12-14]. We can harness the power of PSO to train our model to reduce the possibility to be trapped in local optimal and speed up the convergence.
The rest of the paper is organized as follows: in section 1, we will give a brief introduction of the study. Section 2 deals with model development using FLANN structure using the improved wavelet mutation based PSO algorithm. The details of experiment setup including process are discussed in section 3. The simulation study and the results obtained from experiment are provided in section 4. In the final section the conclusion has been made.

II. MODEL DEVELOPMENT

A. The FLANN Model

Depending on the complexities of the problems, number of layers and number of neurons in the hidden layer need to be changed. As the number of layers and the number of neurons in the hidden layer associated with multi-layer neural network increases, training the model becomes more complex. In FLANN, each element of the input data undergoes functional expansion through a set of basis functions to enhance the input pattern with nonlinear functional expansion. This enables the FLANN to solve complex problems by generating non-linear decision boundaries.

The architecture consists of two parts, namely, transformation part and learning part. The transformation deals with the input feature vector to hidden layer by approximate transformable method. A simple FLANN model with a pattern of two features is shown in Fig. 1.

\[ X = [(Ch_0(I_1), Ch_0(I_2) \cdots Ch_{m1}(I_1)), \]
\[ (Ch_0(I_2), Ch_0(I_2) \cdots Ch_{m1}(I_2))]^T \]  \hspace{1cm} (2)

where \( Ch_0(x) = 1 \), \( Ch_1(x) = x \), \( Ch_{m1}(x) = 2xCh_0(x) - Ch_1(x) \), and \( n \) is the order of the polynomial chosen.

The output of hidden neuron is given by:

\[ \hat{y} = \tanh(\sum_{i=1}^{m} w_i x_i - \theta) \]  \hspace{1cm} (3)

Where \( \theta \) is the threshold of neuron in the output layer.

B. Improved Wavelet Mutation Based Particle Swarm Optimization

In [15], it proposed an improved version of the PSO, where the constriction inertia weight factors are introduced, the velocity \( v_{ij} \) and the position \( x_{ij} \) of the element the \( j \)th of the particle \( i \)th at the \( t \)th iteration can be calculated using (4)(5):

\[ v_{ij}^{t+1} = k \left( (w v_{ij}^{t} + \phi_1 r_1) (pbest_{ij}^{t} - x_{ij}^{t}) + \phi_2 r_2 (gbest_{ij}^{t} - x_{ij}^{t}) \right) \]  \hspace{1cm} (4)

\[ x_{ij}^{t+1} = x_{ij}^{t} + v_{ij}^{t+1} \]  \hspace{1cm} (5)

where \( pbest_{ij} \) is the best position in the history of particle \( i \)th at the \( t \)th iteration; \( gbest \) is the global best position in the history at the \( t \)th iteration; \( r_1 \) and \( r_2 \) are uniform random numbers in the range of \([0,1]\); \( \phi_1 \) and \( \phi_2 \) are acceleration constants is a constriction factor derived from the stability analysis of to ensure the system to be converged but not prematurely. Mathematically, \( k \) is a function of \( \phi_1 \) and \( \phi_2 \) as reflected in (6):

\[ k = \frac{2}{2 - \phi - \sqrt{\phi^2 - 4\phi}} \]  \hspace{1cm} (6)

Where \( \phi = \phi_1 + \phi_2 \), and \( \phi > 0 \)

\( w \) is inertia weight. Generally can be dynamically set with (7):

\[ w = \frac{V_{max} - (V_{max} - V_{min})t}{T_{max}} \]  \hspace{1cm} (7)

Where \( t \) is the current iteration number, \( T_{max} \) is the total number of iteration, and \( V_{max} \) and \( V_{min} \) are the upper and lower limits of the inertia weight.

However, observation reveals that the PSO sharply converges in the early stages of the searching process, but saturated or even terminates in the later stages. In [11], it proposed the operation of the hybrid PSO with a wavelet mutation. Every particle element of the swarm will have a chance to mutate that is governed by a probability of
mutation $p_m$, which is defined by the user. For each particle element, a random number between 0 and 1 will be generated such that if it is less than or equal to $p_m$, a mutation will take place on that element.

The element of particle is randomly selected for the mutation (the value of $x'_i$ is inside the particle element’s boundaries $[par_{i_{max}}, par_{i_{max}}]$), the resulting particle is given by (8):

$$\text{mut}(x'_i) = \begin{cases} x'_i + \sigma \cdot (par_{i_{max}} - x'_i), & \sigma > 0 \\ x'_i + \sigma \cdot (x'_i - par_{i_{max}}), & \sigma \leq 0 \end{cases}$$ (8)

The Morlet wavelet integrates to zero. Over 99% of the total energy of the function is contained in the interval of [-2.5, 2.5]. By using Morlet wavelet in as the mother wavelet.

$$\sigma = 1 - e\left(\frac{\theta}{\pi}\right)^2 \cdot \cos\left(5 \cdot \frac{b}{a}\right)$$ (9)

Hence, the overall positive mutation and the overall negative mutation throughout the evolution are nearly the same. This property gives better solution stability. $\sigma = 1$, $\text{mut}(x_i) = par_{i_{max}}, \sigma = 1$, $\text{mut}(x_i) = par_{i_{max}}$. A larger value of $|\sigma|$ gives a larger searching space for fine-tuning. $b$ can be randomly generated form [-0.25a, 0.25a]. A monotonic increasing function governing $a$ and $t/T_{max}$ is proposed as follows:

$$a = e^{-g(t/T_{max})} \cdot \left(1 + \frac{\text{mut}(x)}{\max}\right) \cdot \text{mut}(x)$$ (10)

where $\xi_{\text{tran}}$ is the shape parameter of the monotonic increasing function, and $g$ is the upper limit of the parameter $a$.

However, after mutation, the value of the particle element is between $[par_{i_{max}}, par_{i_{max}}]$. If the particle did not go by the best solution, it still has the possibility of being trapped in the local optima. So we propose the particle swarm optimization based on an improved wavelet mutation by expand the mutation range during each generation, so that the particle would be more likely to fly near the global best optima. The equation of the mutation can be changed into (11):

$$\text{mut}(x'_i) = \begin{cases} x'_i + \sigma \cdot (\eta_{i_{max}} \cdot par_{i_{max}} - x'_i), & \sigma > 0 \\ x'_i + \sigma \cdot (x'_i - \eta_{i_{min}} \cdot par_{i_{max}}), & \sigma \leq 0 \end{cases}$$ (11)

Where $\eta_{i_{max}}$ and $\eta_{i_{min}}$ is defined by users to expand the mutation range.

### III. ANALYSIS OF DATASETS

#### A. Stimulation Study

<table>
<thead>
<tr>
<th>Function expression</th>
<th>$\xi_{\text{tran}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f_1 = \sum_i x_i^4$</td>
<td>2</td>
</tr>
<tr>
<td>$f_2 = \sum_i \left(\sum_j x_j^4\right)^{\text{random}[0,1]}$</td>
<td>4</td>
</tr>
<tr>
<td>$f_3 = \frac{1}{4000} \sum_{i=0}^{n-1} x_i^4 + \sum_{i=0}^{n-1} \cos\left(\frac{x_i}{\sqrt{i+1}}\right) + 1$</td>
<td>4</td>
</tr>
<tr>
<td>$f_4 = \sum_{i=0}^{n-1} (x_i^2 - 10\cos(2\pi x_i) + 10)$</td>
<td>6</td>
</tr>
<tr>
<td>$f_5 = 0.5 + \sin(\sqrt{x_1^2 + x_2^2} - 0.5) \left(0.1 + 0.001(x_1^2 + x_2^2)^2\right)^2$</td>
<td>6</td>
</tr>
</tbody>
</table>

A suite of five benchmark test functions is used to test the performance of the proposed model. Many different kinds of optimization problems are covered by these benchmark test functions. These five benchmark functions can be divided into three categories. The first one is the category of the unimodal functions, which is a symmetric model with one single minimum, $f_1$ and $f_2$. The second one is the category of multimodal functions with some local minima, $f_3$ and $f_4$. The third category is low dimension function, $f_5$. The benchmark function expressions and their parameter $\xi_{\text{tran}}$ setting for different functions are shown in Table I.

#### B. Empirical Dataset

Empirical study is carried out using the date set CSI 300 from 2008/1/2 to 2017/3/15 up to 2236 trading days. Following 7:3 ratios, we use 1566 days for training and remaining 670 days for validating the model. Choose five kinds of stock prices as the input values in the input layer: daily highest price, daily lowest price, daily closing price, change rate and turnover, and the output layer is the closing price of the next trading day. The entire data is normalized to values between 0 and 1. The normalization formula in (12) to express the data in terms of the minimum and maximum value of the dataset.

$$\text{input} = \frac{x - \text{Min.}}{\text{Max.} - \text{Min.}}$$ (12)

Where input and $x$ represents normalized and actual value respectively.

We use both Chebyshev functions and trigonometric functions to map the data into high dimension. In trigonometric functions, we set the N is 2, and the swarm dimension would be 26, the size of the swarm we set is 60. In trigonometric functions, we set the n is 3, and the swarm
dimension would be 16, the size of the swarm we set is 40. And the $p_m$ and $\varphi_{im}$ we set is 0.2.

To compare the forecasting performance of four considered forecasting models, IWM-PSO-FLANN, WMPSO-FLANN, PSO-FLANN and BP-FLANN, we use the following criteria: the mean absolute error (MAE)(13), the root mean absolute error (RMSE)(14) and the mean absolute percentage error (MAPE)(15), the corresponding definition are given as follows:

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^{N} |y_i - \hat{y}_i|$$

(13)

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

(14)

Where $y$ and $\hat{y}$ represents the actual and forecast values; $N$ is the total number of the data. Noting that MAE, RMSE, and MAPE are measures of the deviation between the prediction values and the actual values, the prediction performance is better when the values of these evaluation criteria are smaller. However, if the results are not consistent among these criteria, we choose the MAPE as the benchmark since MAPE is relatively more stable than other criteria.

Figure 2. Convergence curves of benchmark function for different PSO method
IV. RESULTS AND DISCUSSION

A. Benchmark Test Functions

The results for the five benchmark test functions are in Table II and Fig. 2. From Fig. 2, the IWM-PSO will get the same accuracy as other two models in fewer iteration times. And from Table II, results of simulating experiment in terms of the mean cost values and the best cost value of the IWM-PSO are much better than those of the other methods. Also, the standard deviation is much better, which means that the searched solutions are more stable. From what mentioned above, we can conclude that the IWM-PSO will have better and more stable solution compared to other models.

<table>
<thead>
<tr>
<th>Test Function</th>
<th>IWM-PSO</th>
<th>WM-PSO</th>
<th>PSO</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f_1$ Mean</td>
<td>-34.7321</td>
<td>-22.7162</td>
<td>3.6758</td>
</tr>
<tr>
<td>$f_1$ Best</td>
<td>-47.1855</td>
<td>-38.8609</td>
<td>2.6199</td>
</tr>
<tr>
<td>$f_1$ Std Dev</td>
<td>-68.0077</td>
<td>-44.0165</td>
<td>1.2541</td>
</tr>
<tr>
<td>$f_2$ Mean</td>
<td>-8.8846</td>
<td>-8.3512</td>
<td>4.3211</td>
</tr>
<tr>
<td>$f_2$ Best</td>
<td>-28.8609</td>
<td>-10.5959</td>
<td>3.4875</td>
</tr>
<tr>
<td>$f_2$ Std Dev</td>
<td>-18.9078</td>
<td>-12.5434</td>
<td>0.16147</td>
</tr>
<tr>
<td>$f_3$ Mean</td>
<td>-2.7140</td>
<td>1.7642</td>
<td>2.1664</td>
</tr>
<tr>
<td>$f_3$ Best</td>
<td>1.3407</td>
<td>1.2022</td>
<td>1.9377</td>
</tr>
<tr>
<td>$f_3$ Std Dev</td>
<td>2.3953</td>
<td>2.8338</td>
<td>2.9743</td>
</tr>
<tr>
<td>$f_4$ Mean</td>
<td>1.6944</td>
<td>1.7642</td>
<td>2.1664</td>
</tr>
<tr>
<td>$f_4$ Best</td>
<td>1.3407</td>
<td>1.2022</td>
<td>1.9377</td>
</tr>
<tr>
<td>$f_4$ Std Dev</td>
<td>2.4953</td>
<td>2.8338</td>
<td>2.9743</td>
</tr>
<tr>
<td>$f_5$ Mean</td>
<td>-1.2728</td>
<td>-1.1299</td>
<td>-0.6316</td>
</tr>
<tr>
<td>$f_5$ Best</td>
<td>-3.1565</td>
<td>-2.0125</td>
<td>-1.4325</td>
</tr>
<tr>
<td>$f_5$ Std Dev</td>
<td>-2.0186</td>
<td>-1.7167</td>
<td>-1.3677</td>
</tr>
</tbody>
</table>

B. CSI 300 Index Prediction

Fig. 3. Shows the predictive values on the test set for CSI 300 ((a) uses trigonometric functions and (b) uses Chebyshev functions). The empirical research shows that the proposed the IWM-PSO-FLANN model has the best performance. When the stock market is relatively stable, the forecasting result is near to the actual value. At the same time, we can see that the large fluctuation period forecasting is relatively not that accurate from these four models. But the IWM-PSO-FLANN has the best performance than other three models in this period. In Table III, the results show that the forecasting results of the proposed model are almost smaller than those by other models and these can conclude that the proposed IWM-PSO-FLANN model is better than the three other models. Moreover, the model expanded with trigonometric functions has smaller values of MAE, RMSE, and MAPE, which mean that trigonometric functions is more suitable for this kind of data type.

<table>
<thead>
<tr>
<th>Test Function</th>
<th>IWM-PSO-FLANN</th>
<th>WM-PSO-FLANN</th>
<th>PSO-FLANN</th>
<th>BP-FLANN</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f_1$ Mean</td>
<td>0.018266</td>
<td>0.050491</td>
<td>0.025934</td>
<td>0.039624</td>
</tr>
<tr>
<td>$f_1$ Std Dev</td>
<td>0.032258</td>
<td>0.111879</td>
<td>0.050256</td>
<td>0.088087</td>
</tr>
<tr>
<td>$f_1$ MAPE</td>
<td>3.766379</td>
<td>9.09154</td>
<td>5.126772</td>
<td>7.784697</td>
</tr>
<tr>
<td>$f_2$ Mean</td>
<td>0.298131</td>
<td>0.298132</td>
<td>0.213812</td>
<td>0.098014</td>
</tr>
<tr>
<td>$f_2$ Std Dev</td>
<td>0.438211</td>
<td>0.456479</td>
<td>0.414253</td>
<td>0.143458</td>
</tr>
<tr>
<td>$f_2$ MAPE</td>
<td>6.255943</td>
<td>41.097287</td>
<td>38.053276</td>
<td>39.455323</td>
</tr>
</tbody>
</table>

ACKNOWLEDGMENT

The preferred spelling of the word “acknowledgment” in America is without an “e” after the “g”. Avoid the stilted expression, “One of us (R.B.G.) thanks . . .” Instead, try “R.B.G. thanks”. Put applicable sponsor acknowledgments here; DO NOT place them on the first page of your paper or as a footnote.

REFERENCES


