Wind Speed Prediction using Artificial Neural Networks Based on Grey Wolf Optimizer

Weerachai Jonburom[1], Niwat Angkawisittpan [2,*], Adisorn Nuan-On[2] and Pei Cheng Ooi[3]

[1] Department of Electrical and Computer Engineering, Faculty of Engineering, Mahasarakham University, Kantarawichai, Maha Sarakham, Thailand 44150

[2,*] Research Unit for Computational Electromagnetics and Optical Systems (CEMOS), Faculty of Engineering, Mahasarakham University, Kantarawichai, Maha Sarakham, Thailand 44150

[3] Department of Electrical and Electronic Engineering, Faculty of Engineering, University of Nottingham Malaysia, Malaysia

niwat.a@msu.ac.th[*] (corresponding author)

Abstract. *This paper presented the optimization of Multi-Layer Perceptron (MLP) Artificial Neural Networks (ANNs) using the Grey Wolf Optimizer (GWO) algorithm. The objective was to develop a prediction model for wind signal using artificial neural networks by using the principle of numerical statistical prediction and time-series data from air pressure, temperature, and wind speed. For accuracy and efficiency of the developed prediction model, the model consisted of the time series data divided into two sets as 70% for the learning data set and 30% for the test data set of the total data for the model to learn from the actual data set. The results obtained were Model 3-9-3-1 using input data with 3 nodes, 1 hidden layer with 9 nodes using tansig activation function, 2 hidden layers with 3 nodes using the tansig activation function and the output layer had 1 node using purelin activation function. The mean squared error in the prediction was obtained at 0.0053768. It can be concluded that the prediction model could be used to forecast wind speeds very well. In the future work, the training algorithms by the optimizer may be used to achieve the least erroneous results.*

Received by	3 October 2021
Revised by	13 October 2021
Accepted by	18 October 2021

Keywords:

prediction model, wind speed, artificial neural network.

1. Introduction

The science of forecasting influences plans to deal with economic situation [1-2], agriculture [3], transportation [4], logistics [5], disaster prevention [6] and various epidemics [7]. In data science, there is a branch called predictive analytics, where appropriate prescriptive analytics is used for the future prediction of accurate results in the management to plan for risk and risk management more efficiently. The challenge of climate prediction is that the variability of the forecast data is complex. Due to the advancement of the Internet of Things (IoT), it has become easier to establish a weather station, so the amount of data is large, and the climate prediction data is time series data [8], which can be continuous data streams. In climate data, time series data have autocorrelation or serial correlation.

Artificial intelligence (AI) has been used for more than 50 years. Advances in computing power, the availability of huge amounts of data, and new algorithms have led to major breakthroughs in AI in recent years. AI is considered the core of the digital transformation of society. It is the ability of machines to demonstrate human abilities such as reasoning, learning, planning, and creativity. It allows systems to perceive their environment, process what they perceive, solve problems, and take actions to achieve specific objectives. An Artificial Neural Network (ANN) is a part of computer systems designed to simulate how the human brain analyses and processes data or information [9]. It is the foundation of AI and solves problems that have proven difficult by human [10-19]. From the advances of ANN, and the complex and continuous data streams in the climate data, this is a challenge to apply ANN to forecast the future climate data.

A wind gust is a short, sudden increase in wind speed. These sudden bursts in wind speed are often dramatic, causing trees to tip over, weather stations to be destroyed, and other types of damage. When monitoring severe thunderstorms, the wind gust is always referenced.

The objective of this study is to use ANN to develop a prediction model of wind speeds and wind gusts based on meteorological data from the Sakon Nakhon weather station in Thailand between 2015 and 2019. The data collected includes air pressure, temperature and wind speed. The accuracy efficiency of the developed predictive model is defined using the ANN with an optimizer called GWO [15].

The organization of this paper is as follows: Section 2 presents explanation of the application tools. Section 3 provides the details of dataset. The explanation of the proposed method is given in Section 4. Result and Discussion are carried out in Section 5. Finally, the conclusions of this study are described in Section 6.

2. Literature Review

2.1 Multi-Layer Perceptron Neural Network (MLP-NN)

MLP-NN consists of 3 network layers: the input layer, hidden layer, and output layer. A hidden layer may have more than one layer as shown in Fig. 1.

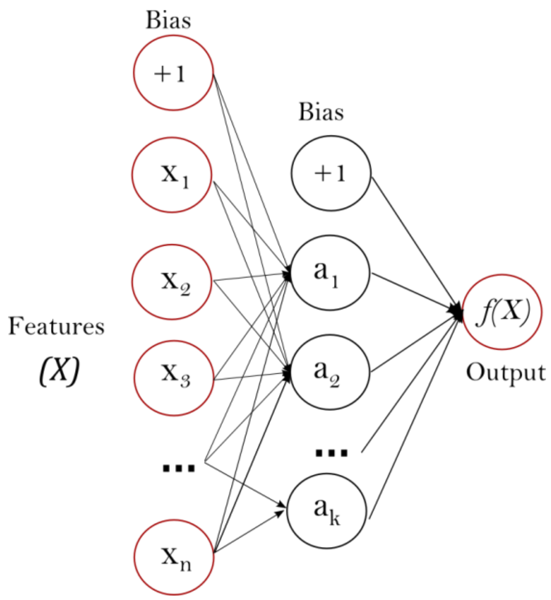

Fig. 1 One hidden layer MLP

A MLP using back-propagation learning was presented by Rumelhart in 1985 [16-17]. The principle of the learning process of the model is changing the weights of each connection to optimize the model results for the most accurate values. Back-propagation learning was used to adjust the weighting as in Eq. (1).

$$\Delta\omega_{ij}(n+1) = \eta\delta(n)y_i(n) + \alpha\Delta\omega_{ij}(n) \qquad (1)$$

where X_i is the input value at node is i, w_i is the weighted value at node i, Δw_{ij} is the weighting adjustment value between nodes i and j, η is the learning rate, α is the momentum, δ_i is the difference between the actual value and the calculated value as a derivative of the transfer function of node j, y_i is the result of the model at the node j and n, and $n+1$ is the value representing the revision loops at n or $n+1$.

In the training neural networks, several activation functions can be selected for example as shown in Fig. 2, which shows the logsig activation function. Fig. 3, shows the tansig activation function and Fig. 4, shows the purelin activation function.

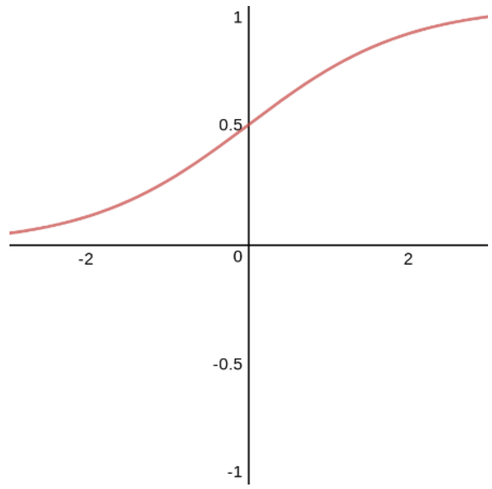

Fig. 2 Log-sigmoid function

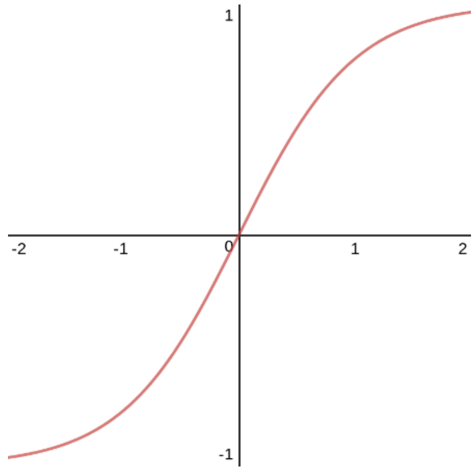

Fig. 3 Tan-sigmoid function

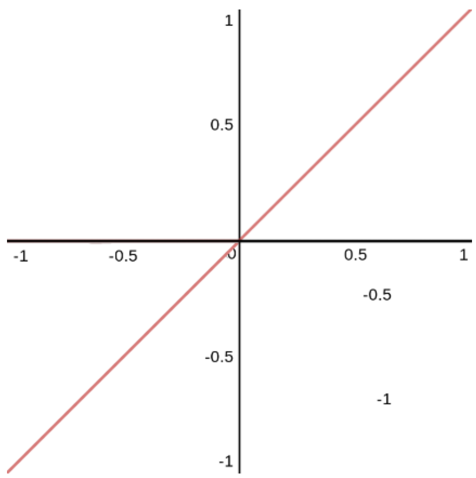

Fig. 4 Pure-linear function

2.2 Grey Wolf Optimizer (GWO)

The GWO is an algorithm that mimics the social leadership and predatory behavior of grey wolf in nature. This is a new concept introduced by Long in 2014 [20]. In this algorithm, the population is divided into 4groups as Alpha (α), Beta (β), Delta (δ) and Omega (ω). The first three wolves α, β, and δ point the prey to the last wolf (ω). Then the last wolf goes to the most likely areas to find the prey. During the search for the prey, wolves will improve their position around as in Eq. (2), and Eq. (3).

$$\vec{D} = |\vec{C}\cdot\vec{X}_p(t) - \vec{X}(t)| \tag{2}$$

$$\vec{X}(t+1) = \vec{X}_p(t) - \vec{A}\cdot\vec{D} \tag{3}$$

Where t is the position of the current prey, $\vec{A} = 2a\cdot\vec{r}_1 a, \vec{c} = 2\cdot\vec{r}_2, \vec{X}_p$ is the position vector of the prey, $\vec{X}$ is the position vector of the wolf reduced in a straight line from 2 to 0 and $\vec{r}_1$, $\vec{r}_2$ dom vector in [0,1]. Theran concept of position improvement using equations (2) and (3) are illustrated in Fig. 3. The wolf in the (*X*, *Y*) position can move themselves around their prey as proposed in the equations (2) and (3). However, the 7th position of the locations can be illustrated as shown in Fig. 3, the random parameters *A* and *C* only allow the wolf to move to any position in a continuous space around the prey.

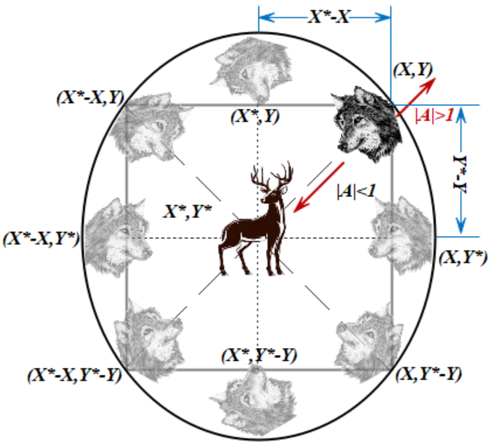

Fig. 5 Algorithm for improving prey search position

The GWO algorithm always assumes that α, β, and δ will be the closest to the best prey. The best solution optimization are the three positions obtained consisting of α, β, and δ respectively. Then, the ω can be repositioned with respect to α, β, and δ respectively. The proposed mathematical model to re-adjust the position of ω wolf is shown in Eq. (4), Eq. (5), and Eq. (6).

$$\vec{D}_\alpha = |\vec{C}_1\cdot\vec{X}_\alpha - \vec{X}| \tag{4}$$

$$\vec{D}_\beta = |\vec{C}_2\cdot\vec{X}_\beta - \vec{X}| \tag{5}$$

$$\vec{D}_\delta = |\vec{C}_3\cdot\vec{X}_\delta - \vec{X}| \tag{6}$$

where $\vec{X}_\alpha$ is the position of α, $\vec{X}_\beta$ is the position of β, $\vec{X}_\delta$ is the position of δ and $\vec{C}_1$, $\vec{C}_2$, $\vec{C}_3$ are random vectors and $\vec{X}$ indicates the current position as shown in Eq. (4), Eq. (5), and Eq. (6) which can calculate the approximate distance between the current position and α, β, and δ respectively. The last position can be calculated using by the Eq. (7), Eq. (8), Eq. (9), and Eq. (10).

$$\vec{X}_1 = \vec{X}_\alpha - \vec{A}_1\cdot(\vec{D}_\alpha) \tag{7}$$

$$\vec{X}_2 = \vec{X}_\beta - \vec{A}_2\cdot(\vec{D}_\beta) \tag{8}$$

$$\vec{X}_3 = \vec{X}_\delta - \vec{A}_3\cdot(\vec{D}_\delta) \tag{9}$$

$$\vec{X}(t+1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} \tag{10}$$

where $\vec{X}_\alpha$ is the position of α, $\vec{X}_\beta$ is the position of β, $\vec{X}_\delta$ is the position of δ and $\vec{A}_1, \vec{A}_2, \vec{A}_3$ are random vectors and t is the number of iteration as shown in Eq. (4), Eq. (5), and Eq. (6) which define the direction of the ω towards α, β, and δ, respectively.

The Eq. (7), Eq. (8), Eq. (9), and Eq. (10), are the result of the last position of the ω wolf .The vectors $\vec{A}$ and $\vec{C}$ are random and aligned vectors for use in the survey and to reach the goal of the GWO algorithm as shown in Fig. 4. A survey or search occurs when $\vec{A}$ is greater than 1 or less than -1. The target is done when $|A|<1$ and $C<1$. It should be noted that A decreases linearly during optimization to highlight utilization while increasing search cycles. However, the C is generated randomly throughout optimization cycle to highlight exploration as a very useful mechanism for resolving target failures.

2.3 GWO-based MLP Trainer

Mirjalili, S., presented the effect on the effectiveness of training multilayer neural networks with GWO [21, 22]. The process of training a multilayer perceptron using meta-heuristics, which is a method for finding rational answers, has been very popular in finding answers today in processing speed and solving complex problems. The most important step in MLP training are weight and bias. The trainers should find a dataset of weight and bias values that provide the highest accuracy for prediction. Therefore, the variables here are weights and bias following the GWO algorithm accepting variables as vectors as shown in Eq. (11).

$$\vec{V} = \{\vec{W}, \vec{\theta}\} = \{w_{1,1}, w_{1,2}, \ldots w_{n,m}, h, \theta_1, \theta_2, \ldots \theta_h\} \tag{11}$$

where n is the number of input nodes, w_{ij} is the connection weight from input layer i to j node in hidden layer, θ_j is the bias (threshold) of the hidden layer node. After that, the variable is defined in the target function. The purpose of training MLP is access to classification and prediction for training and testing samples of the most common metric. The evaluation of the MLP performance is the mean square error (MSE) as in Eq. (12).

$$MSE = \sum_{i=1}^{m} (o_i^k - d_i^k)^2 \tag{12}$$

where m is the number of outputs, d_i^k is the desired output of the input at i, when using training samples at k, o_i^k is the actual output of the input at i when training at k appears in the input.

The MLP should adapt itself to the entire training set to be effective. Therefore, the performance of the MLP is assessed based on the average of the MSE across all training as in Eq. (13).

$$MSE = \sum_{k=1}^{8} \frac{\sum_{i=1}^{m} (o_i^k - d_i^k)^2}{s} \tag{13}$$

Where s is number of training sample, m is the number of outputs, d_i^k is the desired output of the input unit i, when training sample at k, and o_i^k is the actual output of the input unit i when training samples at k appear in the input.

After that, the MLP training problem can be defined with variables and MSE for the GWO algorithm as shown in Eq. (14).

$$Minimize : F(\vec{V}) = MSE \tag{14}$$

In Fig. 6, shows the overall process of training an MLP using the GWO algorithm which uses weight and bias values and the MLP to send the average MSE for training. The GWO algorithm adjusts to weight/bias to reduce MSE.

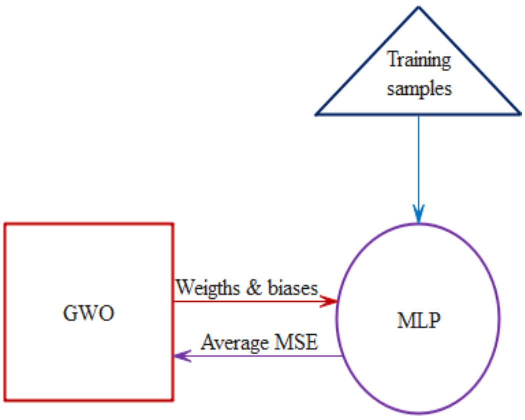

Fig. 6 The overall process of training the MLP using the GWO

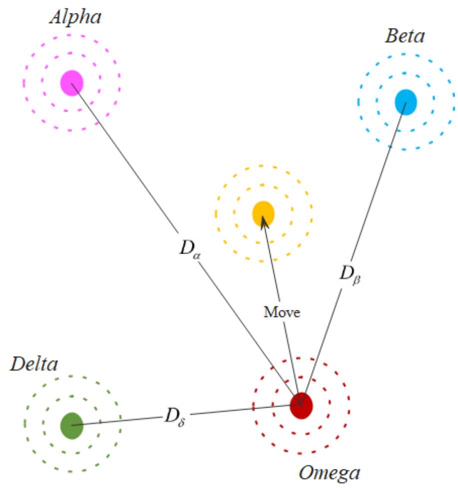

Fig. 7 The concept of training MLP by GWO search agent

The methods used for goal finding and MLP training with the GWO algorithm are illustrated in Fig. 7. It can be seen that the ω wolf tends to be higher than α, β, and δ, respectively. The next position will be close to α and β. Therefore, the involvement of the α wolves in weights determination and the prediction of ω wolves is the highest level as shown by the number of weights and bias that turned purple. When the ω wolf movement is higher than β and δ, a less engagement of β and δ in the next position of ω is clearly highlighted in blue and green and it can be noticed even if similar colors are used. Incomplete weight and bias values are replaced by, α and β, respectively.

Therefore, the high tendency of learning and approaching the goals of the MLP improved with each round. In this way, there is no absolute guarantee for finding the most suitable MLP for the dataset using the GWO. Due to the random behavior of the algorithm, the best possible MLP is used in each cycle. However, the MSE decreases during Epochs of learning iteration or the GWO converge values.

3. Dataset

The dataset used for prediction was from meteorological station at Sakon Nakhon weather station in Thailand between 2015 and 2019. It was daily climate report data including air pressure data between 0-1,100 hPa, temperature data between 0-42 °C, and wind speed data between 0-50 mph. The dataset was a nonlinear time series of 1,825 datasets divided into the datasets used for learning (Training Set) accounting for 70% or 1,277 datasets so that the model could learn from the actual datasets, while the test datasets accounted for 30% or 548 datasets.

4. Proposed Method

The proposed method is divided into three processes; data transformation, development of the prediction model, and the training and testing of the prediction model.

4.1 Data Transformation

The scope of data was adjusted to a range suitable for use in training neural networks to learn using the principle of the neural network. The field values were be converted using data normalization methods which reduced the value of the data to the extent suitable for the process in the MATLAB program and the functionality of the neural network. [23, 24]. Data handling could lead to missing data causing incorrect results [25]. In this paper, the average method was used to replace missing data such as deleted data, average, random, and etc.

4.2 Development of the Prediction Model

There are four steps for development of prediction model as follows.

1) when $X(t)=\{x_1, x_2, \ldots, x_n\}$ is a dataset of input data at time t, where n is the number of input data, $Y(t+1)=\{y_t\}$ is a dataset of output data at time $t+1$, divided into 2 sets consisting of the training set and the test set.

2) the pre-processing for the datasets is prepared by converting the datasets to be in the range [-1, 1] using the data normalization method.

3) neural network is trained using input $x(t)$ and output data $Y(t+1)$.

4) prediction using test datasets and measurement of the performance of the prediction model with MSE are performed.

4.3 Training and Testing of the Prediction Model

1) This research is used to develop a prediction model using the MATLAB program by choosing a multi-layer feed-forward neural network and applying the reverse learning propagation model to the study. The model is chosen because it is a network that can be learned by adjusting the weight to reduce the discrepancy between the output values and the target values which result in an error of higher output than the set value. Therefore, the network must be adjusted.

2) the trainlm (Levenberg Marquardt Algorithm) and the GWO-based MLP trainer is a training function which is fast in processing and suitable for solving complex problems.

3) the LearnGDM (Gradient descent with momentum weight/bias learning function) is a learning function.

4) a number of layers in finding output count from the hidden layer to the output layer. This is not stable until the most accurate result is obtained.

5) function is activated for transferring the data.

6) Measurement of the model performance with MSE is performed [26].

Fig. 8 shows the training process of the neural network model. The datasets are used to test the processing capabilities of trained neural networks.

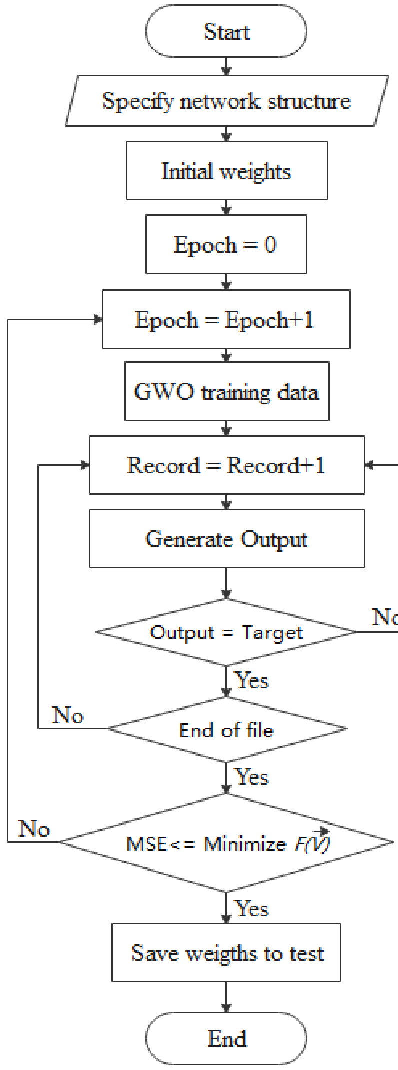

Fig. 8 The flowchart of training of the neural networks

5. Results and Discussions

The prediction model for wind speeds was developed through artificial neural networks using input data as time (t) and the next output data time($t+1$) as result data. The neural network in a multi-layer perceptron was trained and a reverse learning propagation model was used to increase network efficiency with the GWO. The prediction model provided an input layer with three nodes: hidden layer and an output layer using activation functions consisting of the logsig, tansig, and purelin.

The datasets were used to test the processing capabilities of trained neural networks. The results of the prediction model 3-9-3-1 had 3 nodes in the input layer, 9 nodes in the 1st hidden layer using the tansig activation function, and the 2nd hidden layer had 3nodes using the tansig activation function. The output layer had 1node using purelin activation functions, whose MSE was 0.0053768. This was the model with the least error compared to the others as shown in Table 1 (page 52).

After obtaining the optimal prediction model, the next step was to adjust the parameters for the model to find the parameters that gave the prediction results with the least error as shown in Table 2 (page 52).

The optimal parameters for learning were used at a learning rate in the hidden layer and output layer of 0.2/0.2/0.2, while the hidden and output layer momentum parameters were adjusted at 1.2/1.2/1.2 for the activation function. The learning in the hidden layer used tansig function and the output layer used purelin function. The epochs were 250 and the learning network target error value was 0.001. The result of learning information and the model's data learning rate of adjustment trend are shown in Fig. 9 and Fig. 10.

The wind speed prediction was tested. The predicted data and actual data are shown in Fig. 11.

6. Conclusions

A prediction model for wind speeds was developed using artificial neural networks training by the GWO. The wind speed prediction test was performed using the datasets from the meteorological station at Sakon-Nakhon province in Thailand during 2015-2019 and the predicted results were compared with the actual datasets. It was found that model 3-9-3-1 gave the least predictive error. The MSE was 0.0053768, which yielded satisfactory results.

The suggestions for future research may be to apply techniques to optimize the result datasets to increase the accuracy of the forecast and reduce the learning time in the future.

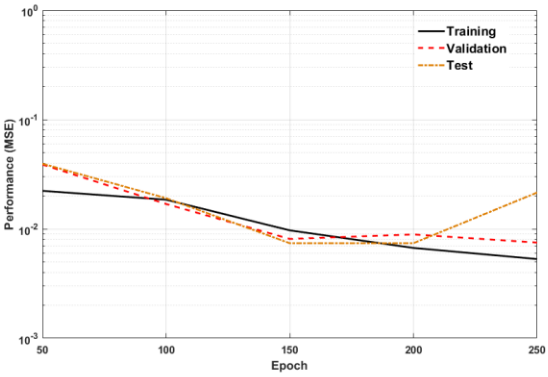

Fig. 9 The performance of the test data

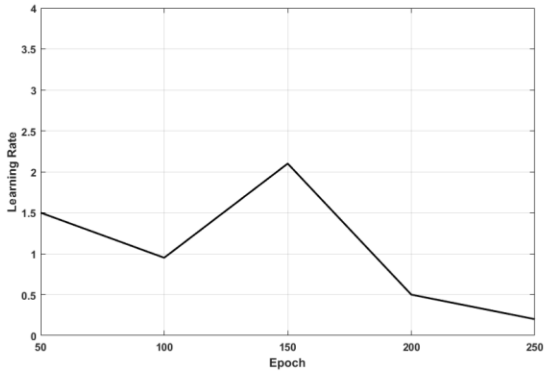

Fig. 10 The learning rate of the neural networks

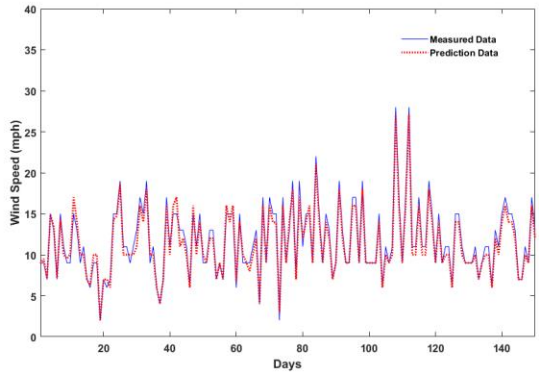

Fig. 11 Comparison of the prediction data and the measured data

Acknowledgment

This work was financially supported by Mahasarakham University, Maha Sarakham, Thailand. Gratitude is extended to Computational Electromagnetics and Optical Systems (CEMOS), Faculty of Engineering, Mahasarakham University, Maha Sarakham, Thailand and Research and Publication Clinic, Institute of Research and Development, Rajamangala University of Technology Isan, Nakhon Ratchasima, Thailand.

References

[1] E. Giovanis,"Study of discrete choice models and adaptive neuro-fuzzy inference system in the prediction of economic crisis periods in USA" *Economic Analysis and Policy*, vol. 42, no. 1, pp. 79-96, 2012.

[2] J. Uthayakumar, N. Metawa, K. Shankar, and S. Lakshmanaprabu, "Intelligent hybrid model for financial crisis prediction using machine learning techniques" *Information Systems and e-Business Management*, vol. 18, no. 44, pp. 617-645, 2020.

[3] M. T. Shakoor, K. Rahman, S. N. Rayta and A. Chakrabarty, "Agricultural production output prediction using supervised machine learning techniques", *2017 1st international conference on next generation computing applications (NextComp)*, *IEEE*, pp. 182-187, 2017.

[4] K. Kumar, M. Parida, and V. K. Katiyar,"Short term traffic flow prediction for a non urban highway using artificial neural network", *Procedia-Social and Behavioral Sciences*, vol. 104, pp. 755-764, 2013.

[5] H. Guo, C. Guo, B. Xu, Y. Xia, and F. Sun, "MLP neural network based regional logistics demand prediction", *Neural Computing and Applications*, vol. 33, pp. 3939-3952, 2021.

[6] R. Jemli, M. Chtourou, and R. Feki,"Insurability challenges under uncertainty: An attempt to use the artificial neural network for the prediction of losses from natural disasters", *Panoeconomicus*, vol. 57, no. 1, pp. 43-60, 2010.

[7] X. Zhu, et al, "Attention-based recurrent neural network for influenza epidemic prediction", *BMC bio informatics*, vol. 20, pp. 1-10, 2019.

[8] D. Do, and J. Choi, "Hourly Average Wind Speed Simulation and Forecast Based on ARMA Model in Jeju Island", *Korea Journal of Electrical Engineering and Technology*, vol. 11, no. 6, pp. 1548-1555, 2016.

[9] A. Sarasiri, "Techniques for learning the funda mentals of neural networks", *article published*, 2017.

[10] S. Arampongsanuwat, "Development of a Prediction Model of PM10 in Bangkok Using Artificial Neural Networks", *National Conference on Computing and Information Technology*, vol. 40, pp.104-109, 2010.

[11] T. Chaipimonplin, "The Roles of Artificial Neural Network for Geographical Research in Thailand", *Social Science Journal Srinakharinwirot University*, vol. 17, pp. 315. 2014.

[12] M. Alexiadis, P. Dokopoulos, H. Sahsamanoglou, and I. Manousaridis, "Short-Term Forecasting of Wind Speed and Related Electrical Power", *Solar Energy*, vol. 63, no. 1, pp.61-68, 1998.

[13] P. Pinson, and G. Kariniotakis,"Wind Power Forecasting Using Fuzzy Neural Networks Enhanced with On-line Prediction Risk Assessment", *IEEE Bologna Power Tech Conference*, vol. 2, pp. 23-26, 2003.

[14] G. Bai, Y. Ding, M. B. Yildirim, and Y. Ding, "Short- term prediction models for wind speed and wind power", *The 2014 2nd International Conference on Systems and Informatics (ICSAI 2014)*, pp. 180-185, 2014.

[15] S. Mirjalili, S. Mirjalili, and A. Lewis, "Grey Wolf Optimizer", *Advances in Engineering Software*, vol. 69, pp. 46-61, 2014.

[16] Y. Cun, et al, "Handwritten digit recognition with a back-propagation network", *In Advances in neural information processing systems*, pp. 396-404, 1990.

[17] P. Kow, et al, "Seamless integration of convolutional and back-propagation neural networks for regional multi-step-ahead PM2.5 forecasting", *Journal of Cleaner Production*, vol. 261, 2020.

[18] M. Sopa, and N. Angkawisittpan, "An application of cuckoo search algorithm for series system with cost and multiple choices constraints", *Procedia Computer Science*, vol. 86, pp. 453-456, 2016.

[19] M. Sopa, and N. Angkawisittpan, "An improved ant system algorithm for maximizing system reliability in the compatible module", *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, no. 4, pp. 3232-3240, 2019.

[20] W. Long, and S. Xu, "A Novel Grey Wolf Optimizer for Global Optimization Problems", *Institute of Electrical and Electronics Engineers (IEEE)*, pp. 1266-1270, 2016.

[21] M. Hassanin, A. Shoeb, and A. Hassanien, "Grey Wolf Optimizer-based Back-propagation Neural Network Algorithm", *Institute of Electrical and Electronics Engineers (IEEE)*, pp. 213-218, 2016.

[22] S. Mirjalili, "How effective is the Grey Wolf optimizer in training multi-layer perceptrons", *Applied Intelligence*, vol. 43, pp. 150-161, 2015.

[23] M. Shamisi, A. Assi, and H. Hejase, "Using MATLAB to Develop Artificial Neural Network Models for Predicting Global Solar Radiation in Al Ain City UAE", *In book: Engineering Education and Research Using MATLAB*, 2011.

[24] S. Zhou, J. Yuan, Z. Song, J. Tang, and L. Kang, "Wind Signal Forecasting Based on System Identification Toolbox of MATLAB", *International Conference on Intelligent System Design and Engineering Application*, vol. 3, pp. 1614-1617, 2013.

[25] H. Kang,"The prevention and handling of the missing data", *Korean Journal of Anesthesiology*, vol. 64, no. 5, pp. 402–406, 2013.

[26] T. Kaur, S. Kumar, and R. Segal,"Application of artificial neural network for short term wind speed forecasting", *2016 Biennial international conference on power and energy systems: towards sustainable energy (PESTSE). IEEE*, pp. 1-5, 2016.

Biographies

Weerachai Jonburom was born in Roi Et Province, Thailand on August, 1, 1981. He received a B.Sc. from Rajabhat Mahasarakham University, Maha Sarakham, Thailand in 2004, and M.Eng. from Mahasarakham University (MSU), Mahasarakham, Thailand in 2021. He has been with the Research Unit for Nano Photonic Research Group (NPRG), Faculty of Industry and Technology, Raagamanggala University of Technology Isan Sakon Nakhon Campus, Sakon Nakhon, Thailand as a lecturer. His current research interests include machine learning and data science.

Niwat Angkawisittpan was born was born in Khon Kaen, Thailand. He received his B.Eng. in Electrical Engineering with honors from Khon Kaen University, Thailand in 1997. He received his M.Sc. in Electrical & Computer Engineering from Purdue University, Indiana, USA in 2003. Also he received his Ph.D. in Electrical Engineering from University of Massachusetts Lowell, Massachusetts, USA in 2009. Since 2009, he has been with the Department of Electrical Engineering, Faculty of Engineering, Mahasarakham University, Maha Sarakham, Thailand as a Lecturer. He has authored or co-authored a number of papers in scientific journals and conference proceedings. His research interests include compact microstrip devices, Metamaterial applications for RF and microwave circuits, and electromagnetic material characterization.

Adisorn Nuan-on was born in Mukdahan, Thailand. He received his B.Eng. in Electrical Engineering with honors from Ubon Ratchathani University, Thailand in 2011. He received his Master of Engineering in Electrical & Computer Engineering from Mahasarakham University, Thailand in 2015. Since 2015, he has been with the Department of Electrical Engineering, Faculty of Engineering, Mahasarakham

University, Maha Sarakham, Thailand, as a Lecturer. He has authored or co-authored a number of papers in scientific journals and conference proceedings. Her research interests include RF circuits.

Pei Cheng Ooi was born was born in Malaysia. She received her M.Sc. in Electrical & Computer Engineering from Purdue University, Indiana, USA in 2003. Also she received her Ph.D. in Electrical Engineering from The University of Nottingham. She has been with the Department of Electrical and Electronic Engineering, The University of Nottingham Malaysia Campus. She has authored or co-authored a number of papers in scientific journals and conference proceedings. Her research interests include microstrip devices, Metamaterial applications for RF and microwave circuits.

Learning Algorithm = trainlm, Train Algorithm = GWO, Epoch = 250, Goals = 0.001			
Network Model		**Transfer Function**	**Performance (MSE)**
1	3-6-1	tansig-purelin	0.020179
2	3-6-1	logsig-purelin	0.022877
3	3-9-1	tansig-purelin	0.024679
4	3-9-1	logsig-purelin	0.019521
5	3-9-1	tansig-tansig	0.017948
6	3-24-1	logsig-purelin	0.018145
7	3-9-3-1	tansig-logsig-purelin	0.0063117
8*	3-9-3-1	tansig-tansig-purelin	0.0053768
9	3-9-3-1	Logsig-tansig-purelin	0.0090696

Table 1 Prediction Model

ID	**Operation function**	**Parameter values**
1	learning rate of hidden layer and output layer	0.2/0.2/0.2
2	Momentum value of hidden layer and output layer	1.2/1.2/1.2
3	Activation function of hidden layer	tansig
4	Transfer function of output layer	logsig
5	Epochs	250
6	Learning network target error	0.001

Table 2 Optimal Parameters