Functional Link Neural Network – Artificial Bee Colony for Time Series Temperature Prediction

Yana Mazwin Mohmad Hassim and Rozaida Ghazali

Faculty of Computer Science and Information Technology
Universiti Tun Hussein Onn Malaysia (UTHM),
86400 Batu Pahat, Johor, Malaysia
{yana,rozaida}@uthm.edu.my

Abstract. Higher Order Neural Networks (HONNs) have emerged as an important tool for time series prediction and have been successfully applied in many engineering and scientific problems. One of the models in HONNs is a Functional Link Neural Network (FLNN) known to be conveniently used for function approximation and can be extended for pattern recognition with faster convergence rate and lesser computational load compared to ordinary feedforward network like the Multilayer Perceptron (MLP). In training the FLNN, the mostly used algorithm is the Backpropagation (BP) learning algorithm. However, one of the crucial problems with BP learning algorithm is that it can be easily gets trapped on local minima. This paper proposed an alternative learning scheme for the FLNN to be applied on temperature forecasting by using Artificial Bee Colony (ABC) optimization algorithm. The ABC adopted in this work is known to have good exploration and exploitation capabilities in searching optimal weight especially in numerical optimization problems. The result of the prediction made by FLNN-ABC is compared with the original FLNN architecture and toward the end we found that FLNN-ABC gives better result in predicting the next-day ahead prediction.

Keywords: Temperature prediction, Functional Link Neural Network, Artificial Bee Colony Algorithm.

1 Introduction

Artificial Neural Networks (ANNs) have been known to be successfully applied in a variety of real world tasks includes prediction, classification, signal processing, image recognition and especially in industry, business and science [1, 2]. The most common architecture of ANNs is the Multi-layer feed forward network known as Multilayer perceptron (MLP). Since the MLP has multilayered structure, the network requires excessive training time for learning [3]. This is because, the number of weight and the training time will increase as the number of layers and the nodes in layer increases [3, 4]. In order to overcome the drawback of MLP, another type of network known as Higher Order Neural Networks (HONNs) have been introduced [5]. HONNs are a type of feed forward neural network which have single layer trainable weights that can help

B. Murgante et al. (Eds.): ICCSA 2013, Part I, LNCS 7971, pp. 427–437, 2013.

in speeding up the training process. They are simple in their architecture and also able to reduce the number of required training parameters in the network. One of the models in HONNs is a Functional Link Neural Network (FLNN) known to be conveniently used for function approximation and can be extended for pattern recognition with faster convergence rate and lesser computational load [6].

FLNN is a class of higher order neural network (HONNs) that utilized higher combination of it inputs[6, 7]. FLNN can capture non-linear input-output mapping, provided that, they are fed with an adequate set of functional inputs[8]. Pao [6] pointed out that FLNN may be conveniently used for function approximation and can be extended for pattern recognition with faster convergence rate and lesser computational load. In the training of FLNN the mostly used algorithm is the Backpropagation (BP) learning algorithm. However, one of the crucial problems with BP-learning algorithm is that it can easily get trapped in local optima [8]. To improve this, the Artificial Bee Colony (ABC) optimization algorithm is proposed in this work to be used to optimize the weight in FLNN instead of the BP-learning algorithm.

In this paper, we described an overview of FLNN for time series prediction task particularly on the training of the network and the proposed Artificial Bee Colony (ABC) optimization as learning algorithm in order to achieve better learning for the network. The rest of this paper is organized as follows: A model description regarding the FLNN and Artificial Bee Colony optimization technique are given in section 2. The proposed FLNN-ABC for the learning scheme is detailed in section 3. The implementation of FLNN-ABC on time series temperature prediction is presented in section 4. Finally, the paper is concluded in section 5.

2 Functional Link Neural Network-Artificial Bee Colony

In this section, the properties and learning scheme of FLNN and ABC optimization are briefly discussed.

2.1 Functional Link Neural Network

Functional Link Neural Network is a class of HONNs created by Pao [7] and has been successfully used in many applications such as system identification [9-14], channel equalization [3], classification [15-18], pattern recognition [19, 20] and prediction [21, 22]. In this paper, we would discuss on the FLNN for the prediction task. FLNN is much more modest than MLP since it has a single-layer network compared to the MLP whilst able to handle a non-linear separable classification and functions approximation tasks. The FLNN architecture is basically a flat network without any hidden layer which has make the learning algorithm used in the network less complicated [23]. In FLNN, the input vector is extended with a suitably enhanced representation of the input nodes, thereby artificially increasing the dimension of the input space [6, 7].

In this work we focused on Functional link neural networks with generic basis architecture. This model uses a tensor representation. Pao [7], Patra [10], Namatamee [24] has demonstrated that this architecture is very effective for classification task.

Fig. 1 depicts the FLNN structure up to second order with 3 inputs. The first order consist of the 2 inputs x_1, and x_2 while the second order of the network is the extended input based on the product unit of x_1x_2. The learning part of this architecture on the other hand, consists of a standard Backpropagation as the training scheme.

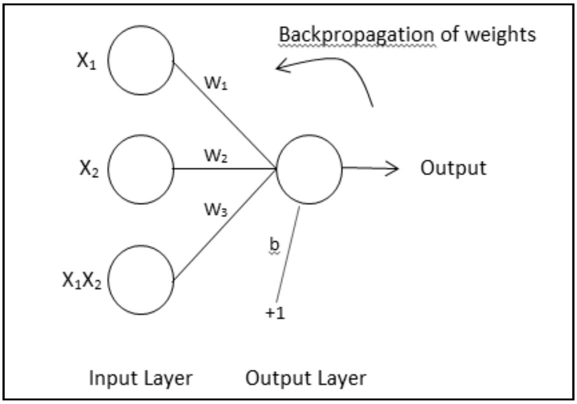

Fig. 1. The 2nd order FLNN structure with 2 inputs

2.2 FLNN Learning Scheme

In most previous researches, the learning algorithm used for training the FLNN is the Backpropagation (BP) [8, 16, 22, 23, 25-27]. BP learning is developed by Rumelhart [28] in which the network is provided with examples of the inputs and desired outputs to be computed, and then the error (difference between actual and expected results) will be calculated. The idea of the backpropagation algorithm is to reduced error, until the networks learned the training data. The training began with initializing random weights for the FLNN network with the goal to adjust these weights set in order to achieve the minimal error through the learning phase.

Even though BP is the mostly used algorithm in training the FLNN, the algorithm however, has several limitations which affect the performance of FLNN-BP network. The FLNN-BP network is prone to get trapped in local minima especially for the training on non-linearly separable classification problems. This is caused by the gradient descent method used by BP-learning algorithm in which the algorithm itself is strictly depends on the shape of the error surface. Since a common error surface on non-linearly separable classification problems may have many local minima and are multimodal, this has typically makes the algorithm susceptible to get stuck in some local minima when moving along the error surface during the training phase.

Another limitation of BP-learning algorithm inherit by FLNN-BP is that, the network is very dependent on the choices of initial values of the weights set as well as the parameters in the algorithm such as the learning rate and momentum [8] which make it not very easy to meet the desired convergence criterion during the training.

For these reasons, a further investigation to improve a learning algorithm used in tuning the learnable weights in FLNN is desired.

2.3 Artificial Bee Colony Optimization

The Artificial Bee Colony algorithm is an optimization tool, which provides a population-based search procedure [29]. The algorithm simulates the intelligent foraging behaviour of a honey bee swarm for solving multidimensional and multimodal optimization problem [30]. In population-based search procedure, each individual population called foods positions are modified by the artificial bees while the bee's aim is to discover the places of food sources with high nectar amount and finally the one with the highest nectar.

In this model, the colony of artificial bees consists of three groups, which are employed bees, onlookers and scouts [31]. For each food source there is only one artificial employed bee. The number of employed bees in the colony is equal to the number of food sources around the hive. Employed bees go to their food source and come back to the hive with three information regarding the food source; 1) the direction 2) its distance from the hive and 3) the fitness. The employed bees then perform waggle dance to let the colony evaluate the information. Onlookers watch the dances of employed bees and choose food sources depending on the dances. After waggle dancing on the dance floor, the dancer goes back to the food source with follower bees that were waiting inside the hive. This foraging process is called local search method as the method of choosing the food source is depending on the experience of the employed bees and their nest mates [30]. The employed bee whose food source has been abandoned becomes a scout and starts to search for finding a new food source randomly without using experience. If the nectar amount of a new source is higher than that of the previous one in their memory, they memorize the new position and forget the previous one [30]. This exploration managed by scout bees is called global search methods.

Several studies done by [30-32] has described that the Artificial Bee Colony algorithm is very simple, flexible and robust as compared to the existing swarm based algorithms: Genetic Algorithm (GA) , Differential Evolution (DE) and Particle Swarm Optimization (PSO) in solving numerical optimization problem. As in classification task in data mining, ABC algorithm also provide a good performance in gathering data into classes [33]. Hence motivated by these studies, the ABC algorithm is utilized in this work as an optimization tool to optimize FLNN learning for a prediction task.

3 FLNN-ABC Learning Scheme

Inspired by the robustness and flexibility offered by the population-based optimization algorithm, we proposed the implementation of the ABC algorithm as the learning scheme to overcome the disadvantages caused by backpropagation in the FLNN training. The proposed flowchart is presented in Fig. 2. In the initial process, the FLNN architecture (weight and bias) is transformed into objective function along with the training dataset. This objective function will then fed to the ABC algorithm in order to search for the optimal weight parameters. The weight changes are then tuned by the ABC algorithm based on the error calculation (difference between actual and expected results).

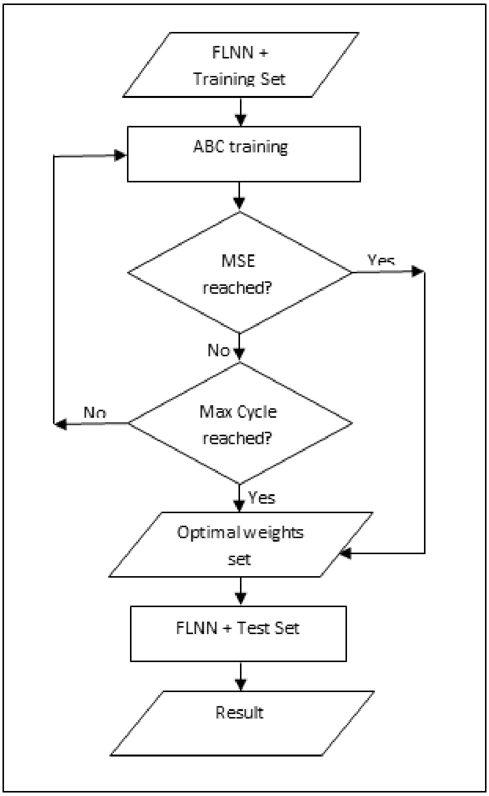

Fig. 2. The Proposed training scheme for FLNN-ABC

Based on the ABC algorithm, each bee represents the solutions with a particular set of weight vector. The ABC algorithm for training the FLNN is summarized as follow:

1) Cycle 0:
2) Initialize optimization parameter of FLNN, $j = 1,2, .. D$.
   where D are the number of weights and biases in FLNN (D= weights + biases)
3) Initialize a population of scout bee with random solution x_i , $i = 1,2, .. SN$.
   where SN denotes the size of population (Solution Numbers).
4) evaluate fitness of the population
5) cycle 1:
   i. form new population v_i for the employed bees using:

$$v_{ij} = x_{ij} + \phi_{ij}(x_{ij} - x_{kj})$$

where k is a solution in the neighbourhood of i, Φ is a random number in the range [-1, 1] and evaluate them.

ii. Apply the greedy selection process between x_{ij} and v_{ij}

iii. Calculate the probability values p_i for the solutions x_i using:

$$p_i = \frac{fit_i}{\sum_{n=1}^{SN} fit_n}$$

iv. Produce the new solutions υ_i for the onlookers from the solutions x_i elected depending on p_i and evaluate them:

v. Apply the greedy selection process for onlookers

vi. Determine the abandoned solution for the scout, if exists, and replace it with a new randomly produced solution x_i using:

$$x_i^j = x_{\min}^j + \mathrm{rand}(0,1)(x_{\max}^j - x_{\min}^j)$$

vii. Memorize the best solution

6) cycle=cycle+1
7) Stop when cycle = Maximum cycle.

4 FLNN-ABC for Time series Temperature Prediction

Weather forecasting is the application of science and technology on predicting the state of the atmosphere of a location for monitoring seasonal changes in the weather. In meteorology domain, weather is measureable in terms of temperature, atmospheric pressure, humidity, wind speed and direction and also cloudiness [34]. Temperature forecast is very important in ensuring the successful of weather forecast and also has significant impact especially in the agriculture activities and water resources [35].

4.1 Experiment Setting

In this study, we used a univariate time series data of daily temperature forecasting for the location of Batu Pahat, Johor ranging from the year 2005 to 2009. The data was obtained from Malaysian Meteorological Department, Malaysia. The simulation experiments were carried out on a 1.66 GHz Core 2 Duo Intel Workstation with 1GB RAM. The comparison of standard FLNN-BP training and FLNN-ABC algorithms is discussed based on the simulation results implemented in Matlab 2010a software for both FLNN-BP and FLNN-ABC network models. Both FLNN-BP and FLNN-ABC will be compared with MLP network trained with BP learning (MLP-BP) as to evaluate in term of network complexity.

The temperature data series is partition into three parts; 50% for training set, and 25% for each validation set and Test set. Both network models were trained for 1000 epochs/cycles on the training set. Training is stopped when the minimum error of 0.0001 reached; or when the maximum of 1000 epochs/cycles reached. We also implemented an early stopping measurement to avoid overfitting. For early stopping measurement, the error on the validation set is measure after 15 epochs/cycles of training. The training is stopped when the validation phase detected an increase error in the validation set for both FLNN-BP and FLNN-ABC. Meanwhile, the test set is used for evaluating the network performance on the unseen data.

Table 1. Parameters setting

	MLP-BP	FLNN-BP	FLNN-ABC
Network structure	5-4-1	15-1	15-1
Optimization parameters/dimensions	29	16	16
Learning Rate	0.5	0.5	-
Momentum	0.5	0.5	-
Colony Size	-	-	50

The number of input node for FLNN was set to 5 up to 2nd order of input enhancement. This was done through trial-and-error procedure between 4 and 8 number of nodes. The 5 inputs with 2nd order inputs enhancement was selected as it gave better output result with less number of trainable parameters (weights + bias) for the FLNN network. The same trial-and-error procedure also performed on the selection of MLP network and the best MLP network structure selected was 5 input nodes with single hidden layer of 4 nodes. The parameters setting for the experiment are presented as Table 1.

As for forecasting horizon, we have chosen a one-step-ahead prediction since the main target is to predict the upcoming measure of daily temperature. The learning rate for MLP-BP and FLNN-BP was set to 0.05 with momentum value of 0.5, while the colony size of 50 bees with weight range of [-2, 2] was set for the FLNN-ABC architecture. The average results of 10 simulations were determined for both FLNN-BP and FLNN-ABC. The Mean Squared Error (MSE), Mean Absolute Error (MAE) and Normalized Mean Squared Error (NMSE) were used to evaluate each network performance.

5 Simulation Results

The comparison of simulation results for MLP-BP and FLNN-BP and FLNN-ABC on the time series temperature prediction data is presented in Table 2 below. Comparison of MLP and FLNN network in term of network complexity showed that the FLNN network required less numbers of parameters (trainable weights + biases) than MLP. The less numbers of parameters indicate that the network required less computational load as there are small numbers of weight and bias to be updated at every epoch or cycle. It can be seen from Table 2 that, training by FLNN-ABC resulted the lowest MSE which is 0.0063 as compared to FLNN-BP and MLP-BP with both are 0.0069 and 0.0075 respectively. When performing on the unseen data, FLNN-ABC also gives better MSE result which is 0.0066 thus outperform both FLNN-BP and MLP-BP with the difference of 0.0002 between FLNN-BP and FLNN-ABC and 0.0004 between

MLP-BP and FLNN-ABC on testing set. FLNN-ABC also gained lower MAE rather than FLNN-BP and MLP-BP which is 0.0633 for FLNN-ABC, 0.0641 for FLNN-BP and 0.0663 for MLP-BP. The lower MAE value indicated that FLNN-ABC was able to produce close forecast to the actual temperature data by outperforming the standard FLNN-BP with the ratio of 1.2 x 10^{-2}.Results from Table 2, also shows that the FLNN-ABC gives lower NMSE compared to both traditional FLNN-BP and standard MLP-BP which shows that the predicted and the actual values obtained by the FLNN-ABC are better in term of measuring the overall deviations of scatter between the prediction and the actual values. On the whole, the performance of training the FLNN network with ABC gives a better prediction result on unseen data when compare to FLNN-BP model and also with less network complexity as compared to MLP-BP.

Table 2. Performance Evaluations on Test Set

	MLP-BP	**FLNN-BP**	**FLNN-ABC**
Number of trainable nodes	29	16	16
MAE	0.0663	0.0641	0.0633
NMSE	0.8343	0.8026	0.6431
MSE Training	0.0075	0.0069	0.0063
MSE Testing	0.0070	0.0068	0.0066

The temperature forecast made by FLNN-ABC and standard FLNN-BP on test sets are graphically presented as in Fig. 3 and Fig. 4. The blue line represents the actual values while the black line refers to the predicted values. From both figures, it is shown that the FLNN-ABC has the ability to follow the actual trend as compared to standard FLNN-BP with minimum error forecast. Hence it can be seen that training scheme by ABC algorithm has facilitate the FLNN with better learning by providing a good exploration and exploitation capabilities in searching optimal weights set in the FLNN weights space as compared to BP learning [31, 36].

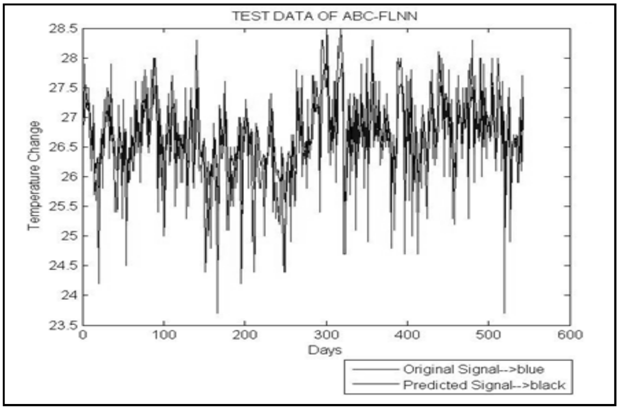

Fig. 3. Temperature forecast made by FLNN-ABC on Test set

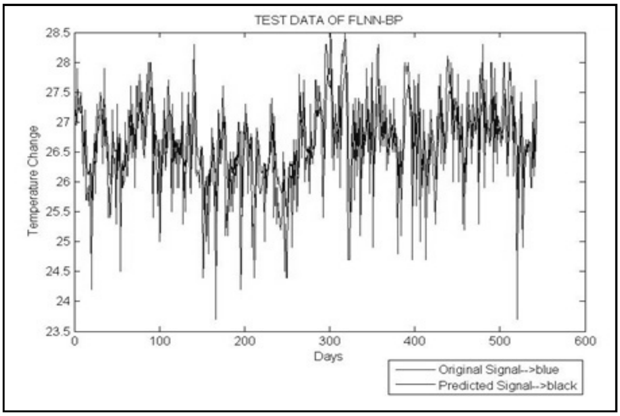

Fig. 4. Temperature forecast made by FLNN-BP on Test set

6 Conclusion and Future Work

In this work, the experiment has demonstrated that the FLNN-ABC performs the temperature prediction task quite well. Implementing the ABC algorithm as a learning scheme for FLNN has shown a significant higher results than backpropagation during experiment in terms of the lowest MSE, NMSE, RMSE and MAE. Since ABC algorithm combine the exploration and exploitation process in it search strategy, it can successfully avoid local minima trapping and provide the FLNN network with better ability in searching for optimal weights set during the training phase. Thus, ABC algorithm can be considered as an alternative learning scheme for training the Functional Link Neural Network instead of standard BP learning algorithm with better scheme in finding minimal error. As for future works, we are considering investigating the use of multivariate data to expand the FLNN-ABC ability for weather forecasting.

Acknowledgement. The authors wish to thank the Ministry of Higher Education Malaysia and Universiti Tun Hussein Onn Malaysia for the scholarship given in conducting these research activities.

References

1. Zhang, G.P.: Neural networks for classification: a survey. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews 30(4), 451–462 (2000)
2. Liao, S.-H., Wen, C.-H.: Artificial neural networks classification and clustering of methodologies and applications – literature analysis from 1995 to 2005. Expert Systems with Applications 32(1), 1–11 (2007)
3. Patra, J.C., Pal, R.N.: A functional link artificial neural network for adaptive channel equalization. Signal Processing 43(2), 181–195 (1995)

4. Chen, A.-S., Leung, M.T.: Regression neural network for error correction in foreign exchange forecasting and trading. Computers & Amp; Operations Research 31(7), 1049–1068 (2004)
5. Giles, C.L., Maxwell, T.: Learning, invariance, and generalization in high-order neural networks. Applied Optics 26(23), 4972–4978 (1987)
6. Pao, Y.H., Takefuji, Y.: Functional-link net computing: theory, system architecture, and functionalities. Computer 25(5), 76–79 (1992)
7. Pao, Y.H.: Adaptive pattern recognition and neural networks (1989)
8. Dehuri, S., Cho, S.-B.: A comprehensive survey on functional link neural networks and an adaptive PSO–BP learning for CFLNN. Neural Computing & Applications 19(2), 187–205 (2010)
9. Patra, J.C., Bornand, C.: Nonlinear dynamic system identification using Legendre neural network. In: The 2010 International Joint Conference on Neural Networks, IJCNN (2010)
10. Patra, J.C., Kot, A.C.: Nonlinear dynamic system identification using Chebyshev functional link artificial neural networks. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 32(4), 505–511 (2002)
11. Abbas, H.M.: System Identification Using Optimally Designed Functional Link Networks via a Fast Orthogonal Search Technique. Journal of Computers 4(2) (2009)
12. Emrani, S., et al.: Individual particle optimized functional link neural network for real time identification of nonlinear dynamic systems. In: 2010 The 5th IEEE Conference on Industrial Electronics and Applications (ICIEA), (2010)
13. Nanda, S.J., et al.: Improved Identification of Nonlinear MIMO Plants using New Hybrid FLANN-AIS Model. In: IEEE International Advance Computing Conference, IACC 2009 (2009)
14. Teeter, J., Mo-Yuen, C.: Application of functional link neural network to HVAC thermal dynamic system identification. IEEE Transactions on Industrial Electronics 45(1), 170–176 (1998)
15. Raghu, P.P., Poongodi, R., Yegnanarayana, B.: A combined neural network approach for texture classification. Neural Networks 8(6), 975–987 (1995)
16. Abu-Mahfouz, I.-A.: A comparative study of three artificial neural networks for the detection and classification of gear faults. International Journal of General Systems 34(3), 261–277 (2005)
17. Liu, L.M., et al.: Image classification in remote sensing using functional link neural networks. In: Proceedings of the IEEE Southwest Symposium on Image Analysis and Interpretation (1994)
18. Dehuri, S., Cho, S.-B.: Evolutionarily optimized features in functional link neural network for classification. Expert Systems with Applications 37(6), 4379–4391 (2010)
19. Klaseen, M., Pao, Y.H.: The functional link net in structural pattern recognition. In: 1990 IEEE Region 10 Conference on Computer and Communication Systems, TENCON 1990 (1990)
20. Park, G.H., Pao, Y.H.: Unconstrained word-based approach for off-line script recognition using density-based random-vector functional-link net. Neurocomputing 31(1-4), 45–65 (2000)
21. Majhi, R., Panda, G., Sahoo, G.: Development and performance evaluation of FLANN based model for forecasting of stock markets. Expert Systems with Applications 36(3, pt. 2), 6800–6808 (2009)
22. Ghazali, R., Hussain, A.J., Liatsis, P.: Dynamic Ridge Polynomial Neural Network: Forecasting the univariate non-stationary and stationary trading signals. Expert Systems with Applications 38(4), 3765–3776 (2011)

23. Misra, B.B., Dehuri, S.: Functional Link Artificial Neural Network for Classification Task in Data Mining. Journal of Computer Science 3(12), 948–955 (2007)
24. Namatame, A., Veda, N.: Pattern classification with Chebyshev neural network. International Jounal of Neural Network 3, 23–31 (1992)
25. Haring, S., Kok, J.: Finding functional links for neural networks by evolutionary computation. In: Van de Merckt, T., et al. (eds.) Proceedings of the Fifth Belgian–Dutch Conference on Machine Learning, BENELEARN 1995, Brussels, Belgium, pp. 71–78 (1995)
26. Dehuri, S., Mishra, B.B., Cho, S.-B.: Genetic Feature Selection for Optimal Functional Link Artificial Neural Network in Classification. In: Fyfe, C., Kim, D., Lee, S.-Y., Yin, H. (eds.) IDEAL 2008. LNCS, vol. 5326, pp. 156–163. Springer, Heidelberg (2008)
27. Sierra, A., Macias, J.A., Corbacho, F.: Evolution of functional link networks. IEEE Transactions on Evolutionary Computation 5(1), 54–65 (2001)
28. Widrow, B., Rumelhart, D.E., Lehr, M.A.: Neural networks: applications in industry, business and science. Commun. ACM 37(3), 93–105 (1994)
29. Pham, D., et al.: The Bees Algorithm, in Technical Note, Manufacturing Engineering Centre, Cardiff University, UK (2005)
30. Karaboga, D.: An Idea Based on Honey Bee Swarm for Numerical Optimization, Erciyes University, Engineering Faculty, Computer Science Department, Kayseri/Turkiye (2005)
31. Karaboga, D., Basturk, B.: On the performance of artificial bee colony (ABC) algorithm. Applied Soft Computing 8, 687–697 (2007)
32. Akay, B., Karaboga, D.: A modified Artificial Bee Colony algorithm for real-parameter optimization. Information Sciences (2010) (in press, corrected proof)
33. Karaboga, D., Ozturk, C.: A novel clustering approach: Artificial Bee Colony (ABC) algorithm. Applied Soft Computing 11(1), 652–657 (2011)
34. Chapter 2. Weather and Climate, `http://www.nasa.gov`
35. Husaini, N.A., Ghazali, R., Mohd Nawi, N., Ismail, L.H.: Jordan pi-sigma neural network for temperature prediction. In: Kim, T.-h., Adeli, H., Robles, R.J., Balitanas, M. (eds.) UCMA 2011, Part II. CCIS, vol. 151, pp. 547–558. Springer, Heidelberg (2011)
36. Mohmad Hassim, Y.M., Ghazali, R.: Using Artificial Bee Colony to Improve Functional Link Neural Network Training. Applied Mechanics and Materials 263, 2102–2108 (2013)