

# Evaluation of Generation System Reliability Using Adaptive Neuro-Fuzzy Inference System (ANFIS) and Artificial Neural Networks (ANNs)

Khanittha Wannakam, Somchat Jiriwibhakorn

**Abstract** – This paper presents an evaluation of the reliability index of power generation systems using the Adaptive Neuro-Fuzzy Inference System (ANFIS) and Artificial Neural Networks (ANNs) to compare the results obtained from the basic method of probability. The reliability index used in this study is the Expected Energy Not Supplied (EENS) index, which is used in planning to increase the installed capacity for the adequate demand for electricity. The ANFIS and ANNs techniques will learn the relationship between the priority level, the installed capacity and the force outage rate (FOR) of the generator, which significantly affect the EENS index. The results indicated that the ANNs techniques have the best predictive performance. The best accuracy of the training data was 1.2488% and the testing data was 2.3963%, calculated using a Mean Absolute Percentage Error (MAPE). Furthermore, the ANNs took more time to learn faster than the ANFIS. **Copyright © 2018 Praise Worthy Prize S.r.l. - All rights reserved.** 

*Keywords:* Adaptive Neuro-Fuzzy Inference System (ANFIS), Artificial Neural Networks (ANNs), Expected Energy Not Supplied (EENS), Generation System, Reliability

# I. Introduction

Currently, the development of industrial equipment or facilities requires electrical energy power sources. An electricity supply is therefore a basic part of an infrastructure and it is important to national development. Continued population growth and ongoing economic growth have increased the annually demands for electricity. Thus, electricity is very important in daily living and is fundamental in driving the economy. If there were insufficient power to supply the demand, this would have a severe effect on the Thai economy. To ensure the capacity of power plants used in production to be large enough to generate a sufficient amount of electricity, it is necessary to plan for reliable electrical systems in accordance with the specified criteria, then they should meet the increasing demand for electricity each year [1]-[27].

The reliability index used for evaluation is the Expected Energy Not Supplied (EENS), which is an index used for decision-making to increase the installed capacity of generators in response to the demand for electricity. In calculation by the probabilistic method, to find the Reliability Index, the Reliability Test System is used to compute the database [1]. The calculation is based on the probability principle of the reliability index calculation. For the ANFIS and ANNs applications [7], [13] the program calculates the correlation of the input and output variables by using a trial and an error method, which allows accurate and fast evaluation of the reliability index.

This was used to make decisions about increasing the installed capacity to respond to the demand for electricity.

Accurate and reliable evaluation are critical to the adequacy of electrical power, as they can determine the timing of future investments and operations in electricity generation. If the evaluation fails, this could affect the reliability and adequacy of the system. This may result in insufficient power to meet the needs of the consumer or in insufficient investment in building power plants.

## **II.** Generation Systems

Most power generation planning is considered only for a single power system, which is used to study the adequacy of the power demand of the system and to determine its reliability. This is due to the failure status of the generator, which may result in the total system capacity being insufficient for the load requirements. In modeling, the specifics of the generator and the model of the demand for electricity [1] were considered.

Generally, the operation of the electrical equipment, such as, generators, is characterized by a period of time between the available and unavailable states.



Fig. 1. Conventional system model

The status of the work is not always due to failure or damage of the equipment, but after repairs have been completed, the operation can continue. The researchers can show the model of the generator using the two-state Markov model [1].



Fig. 2. Two-state Markov model for generating system

The basic generating unit parameter used in the capacity evaluation is the probability of finding the unit on a forced outage of time. This probability is defined in power systems as the unit unavailability. The unit force outage rate (FOR) is the ratio of the time value [1]:

$$Unavailability(FOR) = \frac{\lambda}{\lambda + \mu} = \frac{MTTR}{MTTF + MTTR} \quad (1)$$

Availability = 
$$\frac{\mu}{\mu + \lambda} = \frac{MTTF}{MTTF + MTTR}$$
 (2)

Considering the model of the electrical equipment in the system to be a long-term agent, the Mean time to failure is "Up" and the Mean time to repair is "Down". For this reason, the behavior of the device is periodic.



Fig. 3. Status of equipment in the electrical system

Regarding the long-term capacity planning, it is important to know the estimated amount of the electricity demand and expected peak power. In this case, the researchers used the load duration curve, which changed from an hourly load curve.



Fig. 4. Cumulative load duration curve (CLDC)

The generation and load models were combined to set up the appropriate risk model [1].



Fig. 5. Generating capacity reliability evaluation

## **III.** Reliability Index

Expected Energy Not Supplied (EENS) is an index that indicates the expected energy value of an uninterruptible power supply because the load is greater than the available capacity. The index represents the amount of unrepresented power generated by an inadequate capacity [1]:

$$EENS_0 = t \times \sum_{i=1}^{N} L_i \tag{3}$$

$$EENS_q = \sum_{i=1}^{N} p_k E_k \tag{4}$$



Fig. 6. EENS index calculation

## IV. Adaptive Neuro-Fuzzy Inference System (ANFIS)

The Adaptive Neuro-Fuzzy Inference System uses a hybrid learning algorithm from a two-way learning cycle. In the forward pass, parameters are adjusted using the Least Squares Estimate method while the Backward Pass uses the gradient method, the Gradient Descent for adjusting the parameters of a set [2], [21], [28].

#### IV.1. The Structure of Adaptive Neuro-Fuzzy Inference System (ANFIS)

Layer 1 would transform input (x) into fuzzy values by calculating the value of the functions  $\mu_{A_i}(x_1)$  and  $\mu_{B_i}(x_2)$ , which may be an inverted bell function or another derivative function, such as, a triangular function. Trapezoid functions and the nodes in this layer would be rectangular, which would indicate that the parameters could be adjusted. The parameters in this node would refer to the parameters of the membership function called Premise Parameters.



Fig. 7. The structure of Adaptive Neuro-Fuzzy Inference System [13]

Layer 2 would be responsible for multiplying the signals sent from the first layer together using the T-norm Operator and exporting them to the third layer. The number of nodes in this layer would be equal to the number of fuzzy rules and signals. Export (w) would be the Firing Strength of each rule. The nodes in this class would be in a circular form, which would indicate that the parameters could not be adjusted.

Layer 3 would adjust the Firing Strength to one rule by dividing Firing Strength by Firing Strength. From all rules, the result is called Normalized Firing Strength  $(\overline{w})$ . In this class, a circle would indicate that the parameter could not be adjusted.

Layer 4 would be useful to calculate the result of the fuzzy rule. The nodes in this layer would be rectangular, indicating that the parameters of this node could be adjusted. These parameters are called Consequent Parameters.

Layer 5 would serve as the aggregate of the signals from all four nodes together. Regarding the outgoing data (y), the nodes in this class would be in a circular form, which would indicate that the parameters could not be adjusted.

#### **IV.2.** Membership Function

Selecting a member's function to choose the appropriate data coverage to be received [2] by being able to be overlapped for a smooth operation would have many member values, so the membership function would be changed to suit the purpose:







# V. Artificial Neural Networks (ANNs)

The neural network is a mathematical model that simulates the human brain in learning and memorizing by connecting, processing, and analyzing information. The result, called knowledge, is the outcome of the learning process.

In general, neurons of the ANNs are simulated. When input data are sent, they could be multiplied by the weight, which would represent the importance of each input. The sum of the weighted values resulting from the input data multiplications and the weight values would be analyzed by the transfer function as a result of the output data.



Fig. 8. Neuron network model [7]

Various researchers have proposed a variety of ANNs structures or architecture for use in a variety of applications; such as, feed forward, feedback or recurrent, and competitive models. In order for the network to learn and show the desired behavior, learning would be divided into two types: Supervised Learning and Unsupervised Learning. In this article, the researchers focused on Supervised Learning, as it is used up to 80% of the time.

Supervised neural network modeling is the process of learning. It requires input and output data to be used to learn and memorize relationships [12].

#### V.1. Multi-Layer Feed Forward Neural Networks

1 input neuron model (Fig. 9).



Fig. 9. Neuron with no bias and bias [7]

Multi-input neuron model (Fig. 10).

International Review of Electrical Engineering, Vol. 13, N. 3



Fig. 10. Single neuron with R inputs [7]

#### V.2. Transfer Function [7]

Typical transfer functions are illustrated below:



#### V.3. The structure of Artificial Neural Networks (ANNs)

When taking multiple neurons in parallel, they are made up of layers, and if each layer is serialized, it is a multilayer. When layered together, these are called multilayer feed-forward neural networks. The most popular structure is the multilayer neural network. This is a forward-feeding structure consisting of three layers: the input layer, the hidden layer, and the output layer. The hidden layer may have more than one layer [3]-[6].

Each layer consists of one or more nodes, and data are sent from the input side to the output side without being returned.

The nodes in the same layer are not connected. As shown in Fig. 11, the researchers used the backpropagation technique [14]-[18], which is a way to adjust weight values connected between the nodes, based on the difference between the actual and desired results.



Fig. 11. The structure of Artificial Neural Networks [12]

The neural network consists of an input layer, a hidden layer (layer 1 and 2) and an output layer (layer 3).

Each layer is connected by weight (*w*) and bias (*b*). a1, a2 and a3 are the outputs of the hidden layers 1 and 2 and the output layers. f1, f2 and f3 are the transfer functions of the hidden layers 1 and 2 and the output layer [22]-[23].

The error e(i) when i = 1,2,3, ..., s3 at the output layer is the difference in the output of a3(i) of the neural network and the value of t(i):

$$e(i) = t(i) - a3(i); \quad i = 1, 2, 3, s3$$
 (5)

#### V.4. Algorithm

Therefore, in learning, the training data and algorithm would need to be adjusted for two important parameters in the learning process: weight and bias. Furthermore, multilayer feed-forward neural networks would use the backpropagation algorithm [8]-[10].

The principle of backpropagation would be to adjust the weight and bias according to Equations (6)-(7), so that the error sum of squares (SSE) of Equation (8) would approach zero [12]:

$$w_{new} = w_{old} + \Delta w \tag{6}$$

$$b_{new} = b_{old} + \Delta b \tag{7}$$

$$SSE = \sum_{i=1}^{N} (t_i - a_i)^2$$
(8)

The Levenberg-Marquardt algorithm (LMA) during the weight and bias training was adjusted by Equations (9)-(10):

$$\Delta w = [J^T(w)J(w) + \mu I]^{-1}J^T(w) \cdot e \tag{9}$$

$$\Delta b = [J^{T}(b)J(b) + \mu I]^{-1}J^{T}(b) \cdot e$$
 (10)

During learning,  $\mu$  would be multiplied by a constant; such as, 10 if the new SSE increased and divided by 10 if the new SSE decreased.

If the norm of  $J^{T}(w) \cdot e$  or  $J^{T}(b) \cdot e$  was less than the minimum value set, or was greater than the maximum value set, the program would stop the training immediately:

$$J(w) = \begin{bmatrix} \frac{\partial e(1)}{\partial w_3(1,1)} & \frac{\partial e(1)}{\partial w_3(1,2)} & \cdots & \frac{\partial e(1)}{\partial w_3(1,s2)} \\ \frac{\partial e(2)}{\partial w_3(2,1)} & \frac{\partial e(2)}{\partial w_3(2,2)} & \cdots & \frac{\partial e(2)}{\partial w_3(2,s2)} \\ \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial e(s3)}{\partial w_3(s31)} & \frac{\partial e(s3)}{\partial w_3(s3,2)} & \cdots & \frac{\partial e(s3)}{\partial w_3(s3,s2)} \end{bmatrix}$$
(11)

Copyright © 2018 Praise Worthy Prize S.r.l. - All rights reserved

International Review of Electrical Engineering, Vol. 13, N. 3

$$J(b) = \begin{bmatrix} \frac{\partial e(1)}{\partial b 3(1)} \\ \frac{\partial e(2)}{\partial b 3(2)} \\ \vdots \\ \frac{\partial e(s3)}{\partial b 3(s3)} \end{bmatrix}$$
(12)

#### V.5. Normalization Method

The methodology was used to increase the accuracy of the ANNs output data and to reduce the learning time of the ANNs. The normalization input and output data sets must be made prior to the ANNs learning process, so that the value of the data would be based on the normal distribution. The mean is zero ( $\mu = 0$ ) and the variance is one ( $\sigma = 1$ ).

If each input or output side contains "n", the data set would be  $(X_1, X_2, ..., X_n)$ . Thus, the normalization data sets could be obtained using Equations (13)-(16), respectively [12]:

If 
$$X = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_n \end{bmatrix}$$
 then  $\mu_x = \frac{\sum_{i=1}^n X_i}{n}$  (13)

$$Y = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix} = \begin{bmatrix} X_1 - \mu_x \\ X_2 - \mu_x \\ \vdots \\ X_n - \mu_x \end{bmatrix} \quad ; \mu_y = 0 \quad (14)$$

$$\sigma_{y} = \sqrt{\frac{\sum_{i=1}^{n} (Y_{i})^{2}}{n}}$$
(15)

$$Z = \begin{bmatrix} Z_1 \\ Z_2 \\ \vdots \\ Z_n \end{bmatrix} = Y \cdot \frac{1}{\sigma_y} = \begin{bmatrix} (X_1 - \mu_x) / \sigma_y \\ (X_2 - \mu_x) / \sigma_y \\ \vdots \\ (X_n - \mu_x) / \sigma_y \end{bmatrix} \quad ; \sigma_z = 1 \\ ; \mu_z = 0 \qquad (16)$$

These should then be scaled or bundled into the transfer function using Equations (17)-(19) to enhance the efficiency of the ANNs after the learning process created the a linear distribution model with values between [0.1,0.9] for "n" and there would be a value between [-0.9,0.9] for the "n" set of the output data to be used for learning:

$$aw = \frac{b-a}{mx-mn} \tag{17}$$

$$bw = b - aw \cdot mx \tag{18}$$

$$Scaling = aw \cdot Z_i + bw \qquad ; i = 1, 2, ..., n \qquad (19)$$

#### V.6. Unnormalization Method

If the *O* matrix represents the normalization and scaling of the data set on the output side of each learning curve, then each ANNs output field could be realized by unscaling and unnormalization using Equations (20)-(21), respectively [12].

The Q matrix represents the "n" data set of each ANNs derived data set that would be compared against each target data set. After the learning is completed, the weight and bias values would be stored:

$$[L] = \begin{bmatrix} O_1 / aw - bw / aw \\ O_2 / aw - bw / aw \\ \vdots \\ O_n / aw - bw / aw \end{bmatrix}$$
(20)

$$[Q] = \begin{bmatrix} L_1 \cdot \sigma_y + \mu_x \\ L_2 \cdot \sigma_y + \mu_x \\ \vdots \\ L_n \cdot \sigma_y + \mu_x \end{bmatrix}$$
(21)

## VI. Test

### VI.1. Test procedures using Adaptive Neuro-Fuzzy Inference System(ANFIS) [13]

*Step 1:* Bring the input and output data to normalization from Equations (13)-(19).

Define 3 input variables:

1) Priority level of the generator.

2) Installed capacity of the generator.

3) FOR of the generator.

Define output 1 variable:

1) EENS index of the generator.

*Step 2:* Select data as training data and testing data. The priority level, installed capacity and FOR are input values and the EENS index is the output value.

*Step 3:* Put the prepared data set into the M-file in the MATLAB. The train matrix is the training data and the test matrix is the testing data. Then click on the Run & Save button. The data set is saved to the workspace in the MATLAB and it is prepared for the run.

*Step 4:* Use the "anfisedit" in the command window to run the ANFIS.

*Step 5:* In the Load data field, select the workspace, then choose Load Data and type "train" into the input variable name field to bring the Train data into the ANFIS.

*Step 6:* Select Load Data and type "test" into the input variable name field to bring the testing data into the ANFIS.

Copyright © 2018 Praise Worthy Prize S.r.l. - All rights reserved

International Review of Electrical Engineering, Vol. 13, N. 3



Fig. 12. Graph of training and testing input data

*Step 7:* Select the Generate FIS box. Enter the number of MFs and choose the type of MFs.

*Step 8:* Select the Train FIS box, enter the Epochs, then select Train Now to get the error value of the training.

*Step 9:* Select Testing data and select Test Now to get the error value of the test. Then try to correct it until obtaining the acceptable error value.



Fig. 13. Results of training & testing using ANFIS

*Step 10:* When the acceptable error value is exported, export the file to the ANFIS learning process for the forecasting value by Export to the workspace.

*Step 11:* Find the answer using "evalfis" on the command window. The program displays the output data.

*Step 12:* Unnormalize and unscaling the output values for real values are derived using Equations (20)-(21).

#### VI.2. Test procedures using Artificial Neural Networks (ANNs) [7]

Step 1: Provide training and testing data.
Define 3 input variables:
1) Priority level of the generator.
2) Installed capacity of the generator.
3) FOR of the generator.
Define output 1 variable:
1) EENS index of the generator.



Fig. 14. System structure of ANFIS (MFs 6 3 2)



Fig. 15. Chart of Artificial neural networks (ANNs)

*Step 2:* Bring the training and testing data to normalization from Equations (13)-(19).

*Step 3:* Determine the number of neurons in the hidden layer, set the number of epochs (set to 100-500, but more configurable if the data is very complex) and modify the transfer function.

*Step 4:* Calculate by allowing the ANNs learn from the training data and use the test data in the ANNs learning test.

*Step 5:* The output obtained from the ANNs is compared to the actual output in the form of error values.

*Step 6:* Let the ANNs learn until they receive the smallest error value.

*Step 7:* Unnormalize and unscaling using Equations (20)-(21) to find the optimal result. Perform the procedure shown in Fig. 16.



Fig. 16. Results & error of the ANNs training program

# VII. Performance Measurement of ANFIS and ANNs

The measurement of the accuracy of the ANFIS and ANNs used the Mean Absolute Percentage Error (MAPE) from the ANFIS and ANNs programs, which were compared with the reference output as follows [12]:

$$MAPE = \frac{1}{N} \sum_{i=1}^{N} \left| \frac{Output_{actual} - Output_{predicted}}{Output_{actual}} \right| \times 100\%$$
(22)

## VIII. Results

In Case 1, 32 units of the IEEE Reliability Test System (RTS) were used in the testing in which the installed capacity was 3405 MW and the peak load was 2850 MW [11]. All the learning was done on the MATLAB program using the ANFIS / ANNs Toolbox to find the minimum error value. Case 1 compared the results between the ANFIS and the ANNs using the same set of data. For the RTS systems, there were 32 data sets: 27 training data and five testing data. The input data were the priority level of the generator, installed capacity, and FOR. The output data were the EENS.

Case 1: Learning using the ANFIS technique on the RTS system: When defining a system structure as 7 2 5 using a linear membership function of the type "trimf", the ANFIS learning had the least error value. The MAPE of the training error was 2.2344% and the testing error was 3.9895%, as shown in Table I. For learning using the ANNs technique on the RTS system, when determining the number of neurons to 13 12 and the transfer function of the hidden layer 1, 2 and 3 was "tansig", "purelin" and "purelin", respectively, it was found that the learning of the ANNs would be the smallest error. The MAPE of the training error was 1.2488% and the testing error was 2.3963%, as shown in Table II. When comparing the values between the ANFIS and ANNs, the ANNs showed results closer to the real values than the ANFIS, as displayed in Fig. 17.

 TABLE I

 Test Results Using The Adaptive Neuro-Fuzzy Inference

 System In The RTS Systems

No. of MFs	MFs Type	%M	APE
		Train	Test
725	trimf linear	2.2344	3.9895

TABLE II
TEST RESULTS USING THE ARTIFICIAL NEURAL NETWORKS IN THE
RTS SYSTEMS

No. of	Transfer function			%MAPE	
neurons -	Layer1	Layer2	Layer3	Train	Test
13 12	tansig	purelin	purelin	1.2488	2.3963



Fig. 17. Comparision of the results between the ANFIS and ANNs in Case 1

In Case 2 using Thai thermal power plant systems, 85 units based on PDP 2015 were used in testing in which the installed capacity was 37,612 MW and the peak load was 30,218 MW [19]. All learning was done on the MATLAB program using the ANFIS / ANNs Toolbox to find the minimum error value. Case 2 compared the results between the ANFIS and ANNs using the same set of data. For Thai thermal power plant systems, there were 85 data sets: 70 new training data and 15 new testing data. The input data were the priority level of generator, installed capacity, and FOR. The output data were the EENS.

Case 2: Learning using the ANFIS technique on the RTS system: When defining a system structure as 6 3 6

using a linear membership function of the type "gbellmf", ANFIS learning had the least error value. The MAPE of the training error was 2.1225% and the testing error was 4.0133%, as shown in Table III. For learning using the ANNs technique on the RTS system, when determining the number of neurons to 11 10 and the transfer function of the hidden layer 1, 2 and 3 was "tansig", "logsig" and "purelin", respectively, it was found that the learning of the ANNs would be the smallest error. The MAPE of the training error was 2.4739% and the testing Error was 3.5219%, as shown in Table IV. When comparing the values between the ANFIS and ANNs, the ANNs showed results closer to the real values than the ANFIS, as displayed in Fig. 18.

TABLE III Test Results Using The Adaptive Neuro-Fuzzy Inference System In The THAI Thermal Power Plant Systems

No. of MFs	MFs Type	% N	IAPE
		Train	Test
636	gbellmf linear	2.1225	4.0133

TABLE IV
TEST RESULTS USING THE ARTIFICIAL NEURAL NETWORKS IN THE
THAI THERMAL POWER PLANT SYSTEMS



Fig. 18. Comparision of the results between the ANFIS and ANNs in Case 2

#### IX. Conclusion

The reliability evaluation of power generation using the Adaptive Neuro-Fuzzy Inference System (ANFIS) and Artificial Neural Networks (ANNs) was based on a comparison of the accuracy of the learning of each technique. The test found out that the average error was good and acceptable. However, the ANFIS would take more time to learn than the ANNs. Because of the more variable input and membership functions that would make the structure be more complex, the ANFIS would run slower or stop working, which was a limitation of the ANFIS that could not be solved. The ANNs technique was therefore found to be the most appropriate technique to use in evaluating the reliability of various test systems, as it took time to learn quickly and produced results that were closer to the actual values that were the most accurate. The ANNs could also be improved by adding the number of hidden layers to make learning more effective.

As such, the preparation of the training and test data must be simulated from the program, which would take a long time, so it should be prepared faster. The test data must be different from the training data and must never be used for learning. The input variables used must be significant for the output variables because if they are too large, they will consume more time and memory and would not improve the performance. The preparation of the training data must be distributed consistently and sufficiently in quantity, so that they are of good quality, resulting in better performance and less time. Thus, these techniques could be used to plan long-term capacity expansion.

#### References

- [1] Billinton R., Allan R.N., *Reliability Evaluation of power Systems* (London: Pitman Advance Publishing, 1984, 6-68).
- [2] Jyh Shing, Roger Jang, ANFIS: Adaptive Network based Fuzzy Inference System, *IEEE Transaction on power Systems*, vol. 23, n.3, May/June 1993, pp. 665-685.
- [3] Nima Amjady, Mehdi Ehsan, Evaluation of Power Systems Reliability by an Artificial Neural Network, *IEEE Transaction on Power Systems*, vol. 14, n. 1, Feb 1999, pp. 287-292.
- [4] D.C. Park, M.A. El-Sharkawi, R.J. Marks II, Electric Load Forecasting Using An Artificial Neural Network, *IEEE Transaction on Power Systems*, vol. 6, n. 2, May 1991, pp. 440-449.
- [5] S.Kalyani, K. Shanyi, Study of Neural Network Models for Security Assessment in Power System, *International Journal of Research and Reviews in Applied Sciences*, vol. 1, Issue 2, Nov 2009, pp. 104-117.
- [6] Armando M., Leonidas C., Vladimiro M., Luiz A., Composite Reliability Assessment Based on Monte Carlo Simulation and Artificial Neural Networks, *IEEE Transaction on Power Systems*, vol. 22, n. 3, Aug 2007, pp. 1202-1209.
- H. Demuth, M. Hagan, M. Beale, Neural Network Toolbox<sup>TM</sup> User's Guide (The Math Works, Inc. 1992- 2017).
- [8] N. Amjady, Generation adequacy assessment of power systems by time series and fuzzy neural network, *IEEE Transactions on Power Systems*, Vol. 21, Issue. 3, Aug 2006, pp. 1340-1349.
- [9] Swaroop R., Hussein A., Load Forecasting for Power System Planning and Operation using Artificial Neural Network at Al Batinah Region Oman, *Journal of Engineering Science and Technology*, vol. 7, no. 4, 2012, pp. 498-504.
- [10] P.S. Rajpal, K.S. Shishodia, G.S. Sekhon, An artificial neural network for modeling reliability, availability and maintainability of a repairable system, *Elsevier Science Limited*, 2005, pp. 809-819.
- [11] IEEE RTS Task Force of APM Subcommittee, The IEEE Reliability Test System-1996, *IEEE Transactions on Power Systems*, vol. 14, n. 3, Aug 1999, pp. 1010-1020.
- [12] Somchat Jiriwibhakorn, Application of Neural Networks in Power Systems (Electrical Engineering Faculty: KMITL, 2011).
- [13] Fuzzy logic Toolbox<sup>TM</sup>: Adaptive neuro fuzzy modeling. The Math Works, Received Feb 2017 from https://www.mathworks.com/help/fuzzy/adaptive-neuro-fuzzyinference-systems.html.
- [14] A. Badri, Z. Ameli, A. MototieBirjandi, Application of Aritificial Neural Networks and Fuzzy Logic Methods for Short Term Load Forecasting, *Elsevier Science Limited*, 2012, pp. 1883-1888.
- [15] J. AbdulJaleel, Sreedevi.R.C., Reliability Prediction of 220 kV Kerala Power system using Neural Network, *International Journal of Engineering and Innovative Technology (IJEIT)*, vol. 3, Issue 2, Aug 2013, pp. 141-144.

Copyright © 2018 Praise Worthy Prize S.r.l. - All rights reserved

International Review of Electrical Engineering, Vol. 13, N. 3

- [16] M. Hadow, N.Abd Allah, P.Abdul Karim, Reliability Evaluation of Distribution Power Systems Based on Artificial Neural Network Techniques, *Journal of Electrical and Computer Engineering*, 2012, pp.1-5.
- [17] A. Shrivastava, A. Bhandakkar, Power System Planning and Operation Using Artificial Neural Networks, *International Journal of Engineering and Innovative Technology (IJEIT)*, vol. 4, Issue 1, pp.1-6.
- [18] S. Quaiyum, Y. Lbrahim Khan, S. Rahman, Artificial Neural Network based Short Term Load Forecasting of Power System, *International Journal of Computer Applications*, 2011, pp.1-7.
- [19] Power Development Plan: PDP 2015, Ministry of Energy, Received Apr 2017 from www.eppo.go.th/index.php/th/planpolicy/tieb/pdp.
- [20] K. Xie, H. Zhang, C. Singh, Reliability forecasting models for elevtrical distribution systems considering component failures and planed outage, *Elsevier Science Limited*, 2016, pp. 228-234.
- [21] Y. Yang, Y. Chen, Y. Wang. C. Li, L. Li, Modeling a combined method based on ANFIS and neural network improved by DE algorithm: A case study for short-term electricity demand forecasting, *Elsevier Science Limited*, 2016, pp. 663-675.
- [22] C. N. Lu, H.T. Wu, S. Vemuri, "Neural network based short term load forecasting", *IEEE Transactions on Power Systems*, vol. 8, Issue. 1, Feb 1993, pp. 336-342.
- [23] A. S. Khwaja, X. Zhang, A. Anpalagan, B. Venkatesh, Boosted neural networks for improved short-term electric load forecasting, *Elsevier Science Limited*, 2016, pp. 431-437.
- [24] Galiveeti, H., Goswami, A., Dev Choudhury, N., A Hybrid Method for Distribution Substation Reliability Evaluation, (2017) *International Review of Electrical Engineering (IREE)*, 12 (2), pp. 142-150.
- [25] Rahmat, M., Jovanovic, S., Lo, K., Reliability and Availability Modelling of Uninterruptible Power Supply Systems Using Monte-Carlo Simulation, (2017) *International Journal on Energy Conversion (IRECON)*, 5 (2), pp. 60-66.
- [26] Lin, O., Miyauchi, H., Reliability Forecasting in Distribution System Considering Variable Failure Rate: Combination of Equipment Inspection Method and Weibull Analysis, (2017) *International Review of Electrical Engineering (IREE)*, 12 (1), pp. 67-72.
- [27] Ganesan, E., Dash, S., A New Approach in Modelling and Control of Distributed Energy Resources for Performance Optimisation and Reliability Improvement in a Micro Grid, (2015) International Review on Modelling and Simulations (IREMOS), 8 (1), pp. 26-40.
- [28] Indrawati, A., Girsang, A., Electricity Demand Forecasting Using Adaptive Neuro-Fuzzy Inference System and Particle Swarm Optimization, (2016) *International Review of Automatic Control* (*IREACO*), 9 (6), pp. 397-404.

## **Authors' information**

Department of Electrical Engineering, Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang, Ladkrabang, Bangkok, Thailand.

E-mails: <u>bee\_m\_m@hotmail.com</u> <u>somchat.ji@kmitl.ac.th</u>



Khanittha Wannakam received a B.S. in Electrical Engineering from Burapha University, Thailand, in 2007, an M.S. degree in Electrical Engineering from Chulalongkorn University, Thailand, in 2010. She is currently pursuing a Doctorate degree in the Faculty of Electrical Engineering at King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand.

Her research interests are in power system reliability and neural networks in power systems.



**Somchat Jiriwibhakorn** received B.S. and M.S. degrees in Electrical Engineering from King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand, in 1994 and 1997, respectively, and a Ph.D. in Electrical Engineering from Imperial College, University of London, UK, in 2000. He was a lecturer from 1994 to 2002, an Associate Professor from 2006

to the present with the Faculty of Electrical Engineering, King Mongkut's Institute of Technology Ladkrabang at Bangkok, Thailand. His present research interests are in power system stability, power system planning and forecasting, and applications of neural networks and anfis in power engineering.