International Journal of Power Electronics and Drive System (IJPEDS)
Vol. 11, No. 1, March 2020, pp. 359~366
ISSN: 2088-8694, DOI: 10.11591/ijpeds.v11.i1.pp359-366

An ELM-based single input rule module and its application in power generation

Chong Tak Yaw[1], Shen Young Wong[2], Keem Sian Yap[3]
[1,3] Department of Electronics and Communication Engineering, Universiti Tenaga Nasional, Malaysia.
[2] Department of Electrical and Electronics Engineering, Xiamen University Malaysia, Malaysia

Article Info

Article history:

Received Sep 7, 2019
Revised Nov 12, 2019
Accepted Dec 8, 2019

Keywords:

Data Regression
Extreme Learning Machine (ELM)
NO_x Emission of Power Generation Plant
Single Input Rule Module (SIRM)

ABSTRACT

Extreme Learning Machine (ELM) is widely known as an effective learning algorithm than the conventional learning methods from the point of learning speed as well as generalization. In traditional fuzzy inference method which was the "if-then" rules, all the input and output objects were assigned to antecedent and consequent component respectively. However, a major dilemma was that the fuzzy rules' number kept increasing until the system and arrangement of the rules became complicated. Therefore, the single input rule modules connected type fuzzy inference (SIRM) method where consociated the output of the fuzzy rules modules significantly. In this paper, we put forward a novel single input rule modules based on extreme learning machine (denoted as SIRM-ELM) for solving data regression problems. In this hybrid model, the concept of SIRM is applied as hidden neurons of ELM and each of them represents a single input fuzzy rules. Hence, the number of fuzzy rule and the number of hidden neuron of ELM are the same. The effectiveness of proposed SIRM-ELM model is verified using sigmoid activation functions based on several benchmark datasets and a NOx emission of power generation plant. Experimental results illustrate that our proposed SIRM-ELM model is capable of achieving small root mean square error, i.e., 0.027448 for prediction of NO_x emission.

Corresponding Author:

Shen Yuong Wong
Department of Electrical and Electronics Engineering,
Xiamen University Malaysia,
Jalan Sunsuria, Bandar Sunsuria, 43900 Sepang, Selangor, Malaysia.
Email: shenyuong.wong@xmu.edu.my

1. INTRODUCTION

Lately, Extreme Learning Machine (ELM) has been acknowledged as an effective learning algorithm than the conventional learning methods from the perspective of generalization and learning speed [1-8]. The inspiration of the Extreme Learning Machine (ELM) suggested by Huang et al. comes from biological learning. It is applicable for solving problems pertaining to back-propagation (BP) learning algorithms. It is therefore conjectured that certain parts of the brain signals are made up of random neurons that are independent of their environment [1]. This process is known as ELM or so called Single Layer Feedforward Network (SLFN). Its corresponding general architecture was illustrated in Figure 1. ELM has the capability to make universal approximation with haphazard biases and input weights [9].

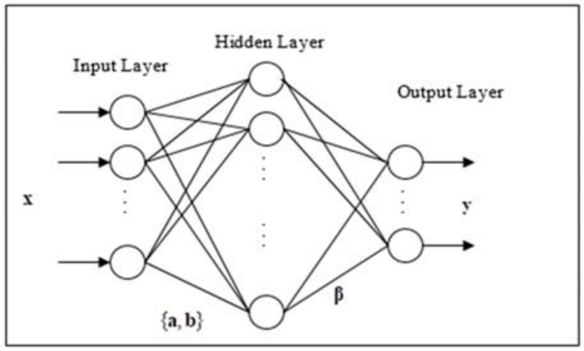

Figure 1. Architecture of ELM

In traditional fuzzy inference method which was the "if-then" rules, all the input and output objects were assigned to antecedent and consequent component respectively. However, a major dilemma was that the fuzzy rules' number kept increasing until the system and arrangement of the rules became complicated [10]. Therefore, the single input rule modules connected type fuzzy inference (SIRM) method where consociated the output of the fuzzy rules modules significantly [11-16].The SIRM method had been applied to control of first as well as second order lag system with dead time [11-12], nonlinear function identification [10], anti-swing control and positioning of overhead traveling crane [13], stabilization control of inverted pendulum systems [14-16], as well as others, of which decent results were acquired [17-22].

Assume that a system consists of n input source and one output source. However, the system can also be extended with plural output sources. This is the basic, with n input source for SIRM:

$$SIRM-i:\left\{R_i^j : if\ x_i = A_i^j \ \ then\ \ \Delta u_i = C_i^j\right\}_{j=1}^{m_i} \tag{1}$$

In (1), each SIRM independently corresponded to n input sources. The SIRM-i where the i refers to i th input source, R_i^j is the j th rule in the SIRM-i, x_i refers to the i th input source variable in the preceding part, and $\triangle u_i$ is the variable in the following part of the SIRM-i. A_i^j and C_i^j are the membership functions of the x_i whereas $\triangle u_i$ is the j th rule in the SIRM-i. Additionally, i = 1, 2,…, n is the index number of the SIRM whereby j = 1, 2,…,m_n is the index number of the rules in the SIRM-i.

This paper proposes an ELM-based model by using ELM hybrid with SIRM (here after denoted as SIRM-ELM). In the SIRM-ELM, there is only a single input that connected to the rules where the rules are the hidden neurons of ELM and each of them represents a single input fuzzy rules. Hence, the number of fuzzy rule and the number of hidden neuron of ELM are equivalent.

The paper is ordered as below. In Section II, the learning algorithms of SIRM-ELM are explained. After that, Section III presents the results of benchmark regression datasets (e.g. Abalone, Balloon, Strike and Space-ga) to test the proposed model's performance. The application of the proposed model is tested and presented in Section IV which is using the NO_x emission in a power generation plant. Lastly, Section V presents a recapitulation of important findings with suggestion for further work.

2. THE ALGORITHMS OF SIRM-ELM

The structure of SIRM-ELM is illustrated in Figure 2. The stepwise training protocols are listed as below. Refer to Figure 2 for the details definition of variables and parameters.

Step 1: Haphazardly set the input weights a_i^j, as well as bias, b_i^j (for i=1, 2..., N where as for j = 1, 2, 3) of hidden neurons. Take into account that a_i^j and b_i^j are parameters of membership function for SIRM, A_i^j. The weights are generated based on $\alpha D - \omega$, where D is uniform distribution function that randomly generate a number between 0 to 1, α and ω are the parameters. By default, $\alpha = 2$, $\omega = 1$. As the result, the a_i^j and b_i^j are in the range of -1 to +1.

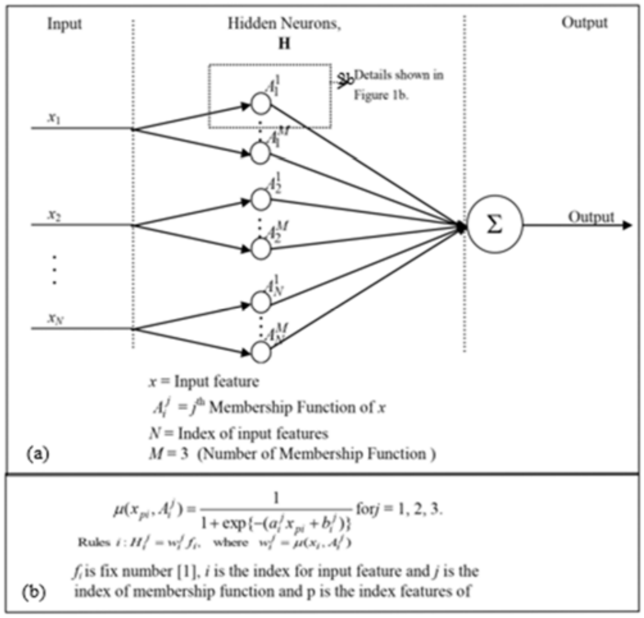

Figure 2. Overview of SIRM-ELM.
(a) General of SIRM-ELM model; (b) General details for each hidden neuron.

Step 2: For the training pair ($\mathbf{x}_{pi}$, t_p) where $\mathbf{x}_{pi}$ is i^{th} feature of p^{th} training pair and t_p is target output (for p = 1, 2, ..., P). Calculate the hidden layer output matrix $\mathbf{H}$ based on membership function $\mu(x_{pi}, A_i^j)$. For simplicity, the membership function can be denoted as μ_{pi}^j

$$\mu(x_{pi}, a_i^j, b_i^j) = \frac{1}{1+\exp\{-(a_i^j x_{pi} + b_i^j)\}} \tag{2}$$

$$\mathbf{H} = \begin{bmatrix} \mu_{11}^1 & \mu_{11}^2 & \mu_{11}^3 & \mu_{12}^1 & \mu_{12}^2 & \cdots & \mu_{1N}^1 & \mu_{1N}^2 & \mu_{1N}^3 \\ \mu_{21}^1 & \mu_{21}^2 & \mu_{21}^3 & \mu_{22}^1 & \mu_{22}^2 & \cdots & \mu_{2N}^1 & \mu_{2N}^2 & \mu_{2N}^3 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \mu_{P1}^1 & \mu_{P1}^2 & \mu_{P1}^3 & \mu_{P2}^1 & \mu_{P2}^2 & \cdots & \mu_{PN}^1 & \mu_{PN}^2 & \mu_{PN}^3 \end{bmatrix}_{P\times 3N} \tag{3}$$

Step 3: The output weights, $\boldsymbol{\beta}$,were computed. Since it is high possibility that $\mathbf{H}$ is a non-symmetry matrix, the inverse matrix cannot be resolved. To circumvent this problem, a moore-penrose pseudo inverse matrix method is utilized, hence work out the output weights of $\boldsymbol{\beta}$ by (4),

$$\boldsymbol{\beta} = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\mathbf{T} \tag{4}$$

where $\mathbf{T}$ is target output matrix, i.e., $\mathbf{T} = [t_1 \quad t_2 \quad \dots \quad t_N]^T$

Step 4: After the output weights of SIRM-ELM were calculated, prediction of a set of new and unlabeled samples $\mathbf{z}$ can be computed, i.e., $\lambda(.)$ is the membership function, $\mathbf{h}$ is the hidden layer whereby $\mathbf{y}$ is the prediction output.

$$\lambda(z_{qi}, a_i^j, b_i^j) = \frac{1}{1+\exp\{-(a_i^j z_{qi} + b_i^j)\}} \tag{5}$$

$$\mathbf{h}=\begin{bmatrix} \lambda_{11}^{1} & \lambda_{11}^{2} & \lambda_{11}^{3} & \lambda_{12}^{1} & \lambda_{12}^{2} & \cdots & \lambda_{1N}^{1} & \lambda_{1N}^{2} & \lambda_{1N}^{3} \\ \lambda_{21}^{1} & \lambda_{21}^{2} & \lambda_{21}^{3} & \lambda_{22}^{1} & \lambda_{22}^{2} & \cdots & \lambda_{2N}^{1} & \lambda_{2N}^{2} & \lambda_{2N}^{3} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \lambda_{Q1}^{1} & \lambda_{Q1}^{2} & \lambda_{Q1}^{3} & \lambda_{Q2}^{1} & \lambda_{Q2}^{2} & \cdots & \lambda_{QN}^{1} & \lambda_{QN}^{2} & \lambda_{QN}^{3} \end{bmatrix}_{Q\times 3N} \tag{6}$$

$$\mathbf{y} = \mathbf{h}\boldsymbol{\beta} \tag{7}$$

where $q = 1, 2, ...Q$ and Q is number of test samples.

Step 5: After compute the output of ELM for testing samples, calculate the root mean squared error (RMSE), i.e.,

$$RMSE_{test} = \sqrt{\frac{\sum_{q=1}^{Q}(y_q - d_q)^2}{Q}} \tag{8}$$

where y_q and d_q were prediction and actual output respective to $\mathbf{z}_q$. Flowcharts were delineated in Figure 3 and Figure 4 to simplify the procedures taken by stepwise training protocols.

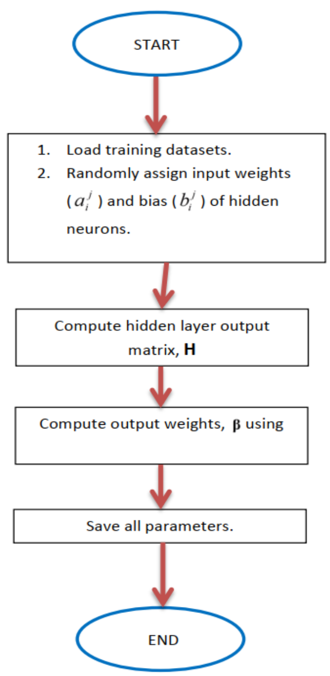

Figure 3. Flowchart that represents the step 1 to step 3.

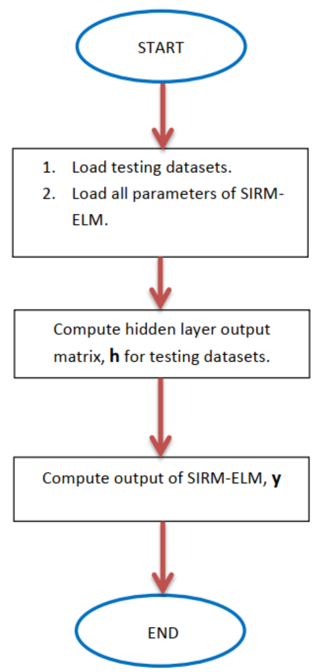

Figure 4. Flowchart that represents the step 4

3. RESULTS AND DISCUSSION

The applicability of the SIRM-ELM model was investigated in this section. Four benchmark regression datasets from the UCI machine repository (e.g. Abalone, Balloon, Strike and Space-ga) were utilized for performance evaluation of SIRM-ELM. Only Addictive Sigmoid hidden neuron (SigAct) was utilized in the analysis. All analysis were run on a personal computer equipped with Intel(R) Core(TM) i7 2.9 GHz CPU and 8 G RAM using MATLAB (ver.2010), as detailed in Table 1. Table 2 listed the datasets specifications used in the experiments.

Table 1. Specification of personal computer and software packages utilized for experiments andcomparison.

Items	Specification
Personal Computer	Asus
Operating Systems	Windows 8.1
CPU	Intel(R) Core(TM) i7 2.5GHz
RAM	8 GB
Software	Matlab 7.11.0.584 (R2010b)
Programming Language	Matlab Language

Table 2. Specification of benchmark regression datasets.

Datasets	# Attributes	# Training Samples	# Testing Samples	# Total Samples
Abalone	8	3000	1177	4177
Balloon	2	1334	667	2001
Strike	6	416	209	625
Space-ga	6	2071	1036	3107

In all experiments, four benchmark regression datasets with training and validation samples were evaluated using the train-validation-test method as suggested by literature [1]. The number of membership function of an input attribute is tested for 1, 2 or 3, (i.e., j = 1, 2, 3) for all the regression datasets. In addition, the RMSE is based on default range for a_i^j and b_i^j for all rules (i.e., i= 1, 2...,3N). Note that in SIRM-ELM, the number of fuzzy rule was equivalent to number of hidden neuron of ELM. For each dataset, the experiments were conducted for 50 times with random a_i^j and b_i^j and the average results are recorded.

The results of proposed SIRM-ELM were also compared to results of other ELM-based methods. As seen from Table 3, the RMSE of SIRM-ELM are better when compare with OS-ELM [21], SVM [21] and ELM [1]. Note that SIRM-ELM perform better than OS-ELM for Abalone dataset as it has only one parameters as compared to OS-ELM that has three parameters.

Table 3. RMSE of SIRM-ELM, ELM [1], SVM [21] and OS-ELM [21]

Algorithm	Abalone RMSE	Balloon RMSE	Strike RMSE	Space-ga RMSE
SIRM-ELM	0.07598	0.04432	0.2656	0.03591
OS-ELM [21]	0.0771	-	-	-
SVM [21]	0.0764	0.059	0.2282	0.0648
ELM [1]	0.0761	0.0553	0.2985	0.0624

4. NO_x EMISSION OF POWER GENERATION PLANT

Nitrogen occurred naturally in the atmosphere as an inactive gas. In addition, our atmosphere contains just about 78% N_2 by volume in the air. The NO_x was referring to nitrogen oxides but mostly include nitrogen monoxide, also identified as nitric oxide, NO as well as nitrogen dioxide, NO_2. There were also others in the family like laughing gas (known as nitrous oxide, N_2O), nitrogen pentoxide (N_2O_5) and nitrogen tetroxide (N_2O_4).

The presence of NO_x in the atmosphere posed direct and indirect effects on human health and ecosystems, i.e. animals and plants, in the environment. NO_x reacted with components such as water, oxygen and other chemicals to form smog and acidic pollutants which leads to the formation of acid rain. In turn, acid rain, together with dry deposition and cloud, may cause damages and deterioration to cars and buildings.

NO_x is mainly released during combustion process of fossil fuels like coal, oil and natural gas. According to European Environment Agency (EEA) technical report (1990 - 2013), 21% of the NO_x gas emissions in European Union were from the energy production and distribution, which was approximately 1,600 kilotonne [23, 24]. However, the growth of power generation industries was expected to be increasing by 18.7 gigawatts (GW) in the coming years, 2016 - 2018, due to price and availability of natural gas. Hence, prediction of NO_x emission is vital to the power generation sector and it shall not be taken lightly.

In case of application, the NO_x emission of an open cycle gas turbine in a power generation plant (located in Port Dickson, Malaysia) has been investigated [25]. The objective was to develop a neural network model for prediction of NO_x emission. There are 150 input attributes taken from the parameters of the power generation plant such as the loading of the gas turbine, temperature, pressure and etc. The targeted output is the quantity of NO_x (in ppm) emission from the gas turbine.

A total of 3,405 data samples have been collected for training and testing of SIRM-ELM. Out of 3,405 data samples, 2,270 were used for training while the remaining 1,135 were used for testing. The number of membership function of an input attribute was tested for 1, 2 or 3, (i.e., j = 1, 2, 3) and the results are shown in Table 4.

Based on the results on the Table 4, the a_i^j and b_i^j ware in default setting (in Step 1). After set the number of membership function of an input attribute as 1, in order to get the lowest root mean squared error (RMSE), the a_i^j and b_i^j need to be tuned in different ranges. The complete tuning results are recorded in Table 5. The best RMSE in Table 5 is 0.028647.

Table 4. Results for NO_x Emission of SIRM-ELM using differences of number of membership function.

# Number of membership function of an input attribute	RMSE
1	0.030358
2	0.056454
3	0.805105

Table 5. Results for NO_x Emission of SIRM-ELM using different ranges of weights.

Range		RMSE
a_i^j	b_i^j	
-1 to +1	-1 to +1	0.030358
-2 to 2	-1 to +1	0.032502
0 to +1	-1 to +1	0.031703
-1 to 0	-1 to +1	0.031173
-1 to +1	0 to +1	0.033823
-1 to +1	-1 to 0	0.033294
-1 to +1	0.5 to +1	0.031032
-1 to +1	0.5	0.028647

In the experiment of using ELM, 2/3 of the data samples were utilized for training while the remaining 1/3 were utilized to verify the most suitable number of neurons of the parent ELM (i.e., L) through a validation process. For sigmoid activation function of ELM, training and validation processes start by setting L = 50 units and then increased by an increment of 50 units. As an example, Table 6 shows the testing processes based on sigmoid activation function. Based on the results of RMSE in Figure 5, the best RMSE is 0.027086. Using the result in Figure 5 to compare with Table 5, the RMSE of ELM is lower than RMSE of SIRM-ELM due to the complexity of hidden neurons in ELM.

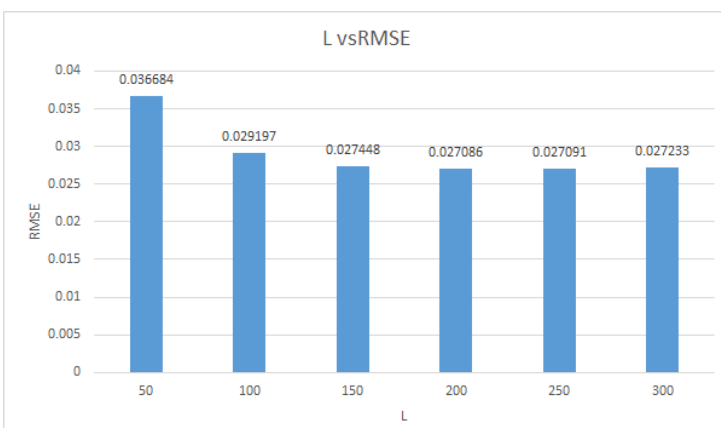

Figure 5. RMSE of NO_x emission for ELM.

4. CONCLUSION

In essence, this paper presented a framework of Extreme Learning Machine with Single Input Rule Module, which was deemed a significant innovation in ELM ideology (here after denoted as SIRM-ELM). Adopting Single Input Rule Module in the ELM hidden layer can be a good alternative to the commonly used activation function, i.e., Sigmoid (SigAct). SIRM-ELM has been tested with sigmoid activation functions utilizing benchmark regression datasets, inclusive of Abalone, Balloon, Strike and Space-ga. The experimental results demonstrated that our proposed model was more superior compared to OS-ELM [21], SVM [21] and ELM [1], as shown in Table 2. As for real world application, the implementation of SIRM-ELM in the prediction of NO_x emitted in power generation plant with low RMSE suggested proposed method is applicable in power generation.

ACKNOWLEDGEMENTS

This work was supported by Universiti Tenaga Nasional (J510050684) and Xiamen University Malaysia (IECE/0001).

REFERENCES

[1] G. B. Huang, Q.Y. Zhu and C. K. Siew, "Extreme learning machine: a new learning scheme of feedforward neural networks," *IEEE International Joint Conference on Neural Networks* (IJCNN2004), vol. 2, Budapest, Hungary, pp. 985-990, Jul 2004.
[2] K. S. Yap and H.J. Yap, "Daily maximum load forecasting of consecutive national holidays using OSELM-based multi-agents system with average strategy," *Neurocomputing*, vol. 81, pp. 108-112, 2012.
[3] G. Huang, G. B. Huang, S. Song and K. You, "Trends in extreme learning machine: A review," *Neural Networks*, vol. 61, pp. 32-48, 2015.
[4] M. V. Heeswijk and Y. Miche, "Binary/ternary extreme learning machine," *Neurocomputing*, vol. 149, pp. 187-197, 2015.
[5] Y. Imamverdiyev and L. Sukhostst, "Anomaly detection in network traffic using extreme learning machine," *IEEE 10th International Conference on Application of Information and Communication Technologies (AICT)*, pp. 14-16, 2016.
[6] G. B. Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme learning machine for regression and multiclass classification", *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 42, no. 2, pp. 513-529, 2012.
[7] G. Huang, Z. Bai, L. Kasun, and C. Vong, "Local receptive based extreme learning machine", *IEEE Computational Intelligence Magazine,* vol. 10, no. 2, 2015.
[8] J. Tang, C. Deng, and G.-B. Huang, "Extreme learning machine for multilayer perceptron," *IEEE transactions on neural networks and learning systems,* vol. 27, no. 4, pp. 809-821, 2016.
[9] M.V. Heeswijk, Y. Miche, E. Oja and A. Lendasse, "GPU-accelerated and parallelized ELM ensembles for large-scale regression," *Neurocomputing,* vol. 74, no. 16, pp. 2430-2437, 2011.
[10] C.D. Li, L.J.L. Gao, J.Q. Yi and G.Q. Zhang, "Analysis and design of functionally weighted single-input-rule-modules connected fuzzy inference systems," *IEEE Trans. on Fuzzy Systems*, vol. 26, no. 1, pp. 56-71, 2018.
[11] N. Yubazaki, J. Yi and K. Hirota, "SIRMs (Single Input Rule Modules) connected fuzzy inference model," *J. Adv. Comput. Intell. Intelligent Inf.*, vol. 1, pp. 23-30, 1997.
[12] J. Yi, N. Yubazaki and K. Hirota, "A proposal of SIRMs dynamically connected fuzzy inference model for plural input fuzzy control," *Fuzzy Sets Syst*., vol. 125, pp. 79-92, 2002.
[13] J. Yi, N. Yubazaki and K. Hirota, "Anti-swing and positioning control of overhead traveling crane," *Inf. Sci*., vol. 155, pp. 19-42, 2002.
[14] J. Yi, N. Yubazaki and K. Hirota, "Stabilization control of series type double inverted pendulum systems using the SIRMs dynamically connected fuzzy inference model," *Artif. Intell. Eng*., vol. 15, pp. 297-308, 2001.
[15] J. Yi, N. Yubazaki and K. Hirota, "Upswing and stabilization control of inverted pendulum system based on the SIRMs dynamically connected fuzzy inference model," *Fuzzy Sets Syst*., vol. 122, pp. 139-152, 2001.
[16] J. Yi, N. Yubazaki and K. Hirota, "A new fuzzy controller for stabilization of parallel-type double inverted pendulum system," *Fuzzy Sets Syst*., vol. 126, pp. 105-119, 2002.
[17] H. Seki and M. Muzumoto, "SIRMs connected fuzzy inference method adopting emphasis and suppression," *Fuzzy Sets and System,* vol. 215, pp. 112-126, 2003.
[18] C. D. Li, L. Wang, G. Q. Zhang, H. D. Wang and F. Shang, "Functional type single input rule modules connected neural fuzzy system for wind speed prediction," *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 4, pp. 751-762, 2017.
[19] H. Miyajima, K. Kishida, N. Shigei and H. Miyajima, "Learning algorithms for fuzzy inference systems composed of double and single input rule modules," *World Academy of Science, Engineering and Technology International Journal of Computer and Information Engineering*, vol. 10, no. 3, pp. 1, 2016.
[20] H. Miyajima, N. Shigei and H. Miyajima, "SIRMs fuzzy inference model with linear transformation of input variables and universal approximation," In: Rojas I., Joya G., Catala A. (eds) *Advances in Computational Intelligence. IWANN 2015. Lecture Notes in Computer Science*, 9094. Springer, Cham, 2015.

[21] T. Yoshimura, "Direct adaptive fuzzy backstepping control for uncertain discrete time nonlinear system using noisy measurements," *International Journal of Systems Science*, vol. 48, no. 4, 2017.
[22] N.Y. Liang and G.B. Huang, "A fast and accurate online sequential learning algorithm for feedforward networks," *IEEE Trans. on Neural Networks*, vol. 17, no. 6, pp. 1411-1423, 2006.
[23] OECD, IEA. "*Energy and air pollution: world energy outlook special report*," 2016.
[24] Asmelash, and Henok Birhanu. "Phasing out fossil fuel subsidies in the G20: Progress, challenges, and ways forward," *Think Piece. Geneva: International Centre for Trade and Sustainable Development (ICTSD)*, 2017.
[25] I.B. Saiful, C.P. Chen and S.K. Tiong, "Prediction of NO_x using support vector machine for gas turbine emission at putrajaya power station," *Journal of Advanced Science and Engineering Research*, vol. 4, no. 1, pp. 37-46, 2014.

BIOGRAPHIES OF AUTHORS

Chong Tak Yaw was born in Kuala Lumpur on February 21, 1984. He received his Bachelors degree from Universiti Tenaga Nasional (UNITEN), Malaysia with Honors in Electrical and Electronics Engineering in 2008. He received his Master degree from Universiti Tenaga Nasional (UNITEN), Malaysia with Honors in Electrical and Electronics Engineering in 2012. Previously he is working as a Project Engineer in the Energy business and Technology Centre (EBTEC) of Universiti Tenaga Nasional (UNITEN) since January 2009 till 2010. In the year of 2011 till 2012, he worked as an engineer in Wintrad Industires (switchboard manufacturer). He just completed his PhD in 2019 in Electrical Engineering. His research interests include neural networks, support vector machines and extreme learning machine.

Shen Yuong Wong received her Bachelor Degree and M.Sc. of Electrical and Electronic Engineering with first class honors from Universiti Tenaga Nasional, Malaysia in 2010, and 2012 respectively. She received her PhD degree in Engineering from the same university in 2015. Her research interests include application of artificial intelligence, pattern recognition, fuzzy logic and Extreme Learning Machines. Currently she is an Assistant Professor at Department of Electrical and Electronics Engineering, Xiamen University Malaysia, Malaysia.

Keem Siah Yap received his B. Eng (Electrical) (Hons.) and M.Sc (Electrical Engineering) degree both from Universiti Teknologi Malaysia in 1998 and 2000 respectively. In year 2010, he received PhD in Electronics Engineering from Universiti Sains Malaysia. His research interests include theory and applications of artificial intelligence. Currently, he is a Professor at College of Engineering, Universiti Tenaga Nasional, Malaysia.