Evolving Radial Basis Function Neural Networks for One-Day-Ahead Hourly Forecasting of PV Power Generation

Chao-Ming Huang^{a*}, Yann-Chang Huang^b, and Kun-Yuan Huang^b

^aDepartment of Electrical Engineering, Kun Shan University, No.195, Kunda Rd., Tainan 710, Taiwan, R.O.C.

^bDepartment of Electrical Engineering, Cheng Shiu University, No.840, Chengcing Rd., Kaohsiung 833, Taiwan, R.O.C.

*Corresponding Author: h7440@ms21.hinet.net

Abstract

This paper proposes a novel method to predict one-day-ahead hourly photovoltaic (PV) power generation. The proposed method comprises three stages: data classification, training and forecasting. In the first stage, a fuzzy k-means algorithm is used to classify the historical data for daily PV power generation into various weather types. In the second stage, five training models are established, according to the verbal weather forecast of the Taiwan Central Weather Bureau (TCWB). Each training model is constructed using a radial basis function neural network (RBFNN), for which the parameters of each RBFNN, including the position of RBF centers, the width of the RBFs and the weights between the hidden and the output layers, are optimized using a harmony search algorithm (HSA). To select an adequate forecasting model from the trained models, fuzzy inference is used in the forecasting stage. The proposed approach is tested on a practical PV power generation system. The results show that the proposed method provides better forecasting results than the existing methods over one-year testing data.

Keywords: PV power forecasting, radial basis function neural network, fuzzy inference, harmony search algorithm.

1. Introduction

To allow efficient planning for grid-connected PV systems, the forecasting of PV power generation is very important. Accurate forecasting of PV power output improves the PV penetration level, increases system reliability and allows for efficient load management strategies, including demand response (DR), the time of use

rates (TOU) and charging schedules for electric vehicle (EV). However, because it is limited by amount of solar irradiation, PV power generation is highly uncertain and difficult to predict. To make effective use of PV power generation, accurate forecasting of PV power output is necessary.

Since the PV power output is affected by varying solar irradiation, many studies focus on predicting solar irradiation, using historical weather information. The forecasted solar irradiances are then converted to PV power outputs, using the manufacture's data for a solar panel. These techniques include time series methods [1], artificial neural network (ANN) methods [2], fuzzy logic methods [3] and wavelet-based methods [4]. Other studies directly forecast PV power output, using historical time-series data and associated weather information. These studies use statistical methods [5] and support vector regression (SVR) techniques [6].

As previously mentioned, most existing studies improve the forecasting accuracy for PV power output. However, a time series method [1] cannot effectively model the high variation in PV power generation, especially for cloudy or rainy days. The ANN method [2] allows good approximation for a wide range of nonlinear functions, but it experiences slow convergence in training and the network structure and parameters must be determined manually. The fuzzy logic method [3] expresses the weather information and the heuristic rules in imprecise linguistic terms, but it is difficult to design an efficient inference engine to draw conclusions from a large volume of rule-based knowledge. The wavelet-based method [4] uses multi-resolution analysis to divide the original data into several levels of subsets. The disadvantages of this method are its complex

structure and the long computational time required. Statistical methods [5] use the influence of several weather parameters on the PV power forecasting. For more accurate forecasting, this method requires accurate detection devices to accurately collect the weather parameters. The SVR method [6] has been successfully applied to data classification and regression prediction. However, the kernel function, which determines the properties of SVC, is often obtained through operator experience.

To allow more accurate forecasting, this paper proposes an intelligent method to predict the one-day-ahead hourly PV power generation. The proposed method comprises data classification, training and forecasting stages. Compared to the existing methods, the proposed approach has the following advantages:

1. Five forecasting models are established to cover diverse PV power outputs, based on different weather conditions. This is more accurate and much more effective than a single forecasting model.
2. HSA is a population-based random search algorithm that provides an efficient scheme to increase the variety of the offspring and has a good probability of converging towards the optimal parameter solution for a RBFNN.
3. Fuzzy inference is used to select an adequate forecasting model from the trained models. This avoids a large forecasting error, if an inaccurate weather forecast is provided by the TCWB.

2. Characteristics of PV Power Generation

PV power generation is sensitive to weather conditions, such as the solar irradiance and temperature, as follows:

$$P_{PV} = n_p V_{pv} (I_{ph} - I_{sat} (\exp(\frac{q}{nkT} \frac{V_{pv}}{n_s}) - 1)) \quad (1)$$

$$I_{ph} = (I_{scr} + \frac{K_r}{1000} (T - T_r)) S_r \quad (2)$$

where P_{PV} is the power output of a PV array, n_p is the number of PV arrays in parallel, n_s is the number of PV arrays in series, V_{pv} is the output voltage of a PV array, I_{ph} is the output current of a PV array, I_{sat} is the dark saturation current, q is the charge on an electron ($1.6 \times 10^{-19} \text{C}$), n is an ideality factor, k is the Boltzmann constant ($1.38 \times 10^{-23} \text{J/}^\circ\text{K}$), T is the absolute temperature ($^\circ\text{K}$), T_r is the reference temperature ($^\circ\text{K}$), I_{scr} is the short-circuit current at both the reference temperature and 1kW/m^2 solar irradiation, K_r is the temperature coefficient of the

short-circuit current and S_r is the solar irradiance (kW/m^2).

As shown in (1) and (2), the PV power generation is affected by solar irradiance and temperature. Different weather types result in a variation in PV power output patterns. Based on the verbal weather descriptions given by the TCWB, the weather types can be classified into sunny, sunny and cloudy, cloudy, cloudy and rainy and rainy. On a sunny day, the power output of PV generation system is high and stable. On the other weather days, the PV power output is low and unstable. In Taiwan, there is solar irradiance from about 6:00 to 19:00 every day.

Since the power output of a PV generation system is intrinsically unstable and has an intermittent nature, providing an accurate forecast of PV power output is important for a renewable energy source. This paper presents a novel method to make more accurately forecast for PV power output. Details of the proposed scheme are given in the next section.

3. The Proposed Approach

3.1 Data classification

The collected historical PV power generation patterns are classified using a fuzzy k-means clustering algorithm. K-means clustering was first proposed by Lloyd in 1982 [7], as a technique for pulse-code modulation. It is a method of vector quantization and is used for cluster analysis in data mining. Given a set of observations ($X_1, X_2, \dots, X_n$), the k-means clustering partitions n observations into k clusters, as follows:

$$\arg \min \sum_{i=1}^k \sum_{j=1}^n w_{ji} \|X_j - R_i\|^2 \quad (3)$$

where X_j is the j th observation, R_i is the i th clustering center, w_{ji} is a synaptic weight and $\|\bullet\|$ is the Euclidean distance. R_i and w_{ji} are respectively expressed as follows:

$$R_i = \frac{\sum_{j=1}^n w_{ji} X_j}{\sum_{j=1}^n w_{ji}} \quad (4)$$

$$w_{ji} = \begin{cases} 1, & \text{if } \|X_j - R_i\| \leq \|X_j - R_m\|, \forall m \neq i \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

As shown in (3), k-means clustering partitions the n observations into k clusters ($k \leq n$), in order to minimize the within-cluster sum of the squares. This algorithm assigns using the least sum of the squares of the Euclidean distance and there is no guarantee that it gives the global optimum.

In contrast to “hard” or “crisp” clusters, fuzzy k-means clustering [8] allows each observation vector, X , to have a degree of membership, as follows:

$$\tilde{w}_{ji} = \frac{1}{\sum_{i=1}^k \left(\frac{\|X_j - R_i\|}{\|X_j - R_m\|} \right)^{\frac{2}{I_m-1}}} \quad (6)$$

where I_m is a weighting index whose value is generally set at 2. Fuzzy k-means clustering has the advantage that it more naturally handles situations in which subclasses are formed by the degree of membership, rather than having to assign X completely to one cluster or the other. In this paper, a fuzzy k-means clustering algorithm is used to extract the features for weather type by classifying the daily PV power output patterns into five different types of weather days. To cope with the variation of PV power output for various seasons, the synaptic weights of fuzzy k-means clustering must be updated according to the recently collected data.

3.2 The creation of training models

As described in the data classification section, a fuzzy k-means clustering algorithm provides information on similar days and their associated weather types. A similar day is defined as having the same weather type. In this stage, the RBFNN evolved using a HSA is used to train the collected input-output data sets. Since the TCWB provides the verbal weather forecasts for the next day, in terms such as the sunny, sunny and cloudy, cloudy, cloudy and rainy and rainy, the training stage is partitioned into five different models.

Figure 1 shows the structure of the training stage. The inputs for each training model contain the actual hourly solar irradiance on the latest similar day, a prediction of the maximum average temperature for the next day and a prediction of probability of precipitation for the next day. The outputs are the hourly PV power output for the next day. The five training models have the same training structure. In this paper, a HSA-based RBFNN is used to train the input/output data sets. The RBFNN and the HSA-based RBFNN model are hereby described.

(a) Radial Basis Function Neural Network (RBFNN) [9]

Fig. 2 shows the structure of a RBFNN. The network consists of an input layer, a hidden layer and an output layer. The training data sets are given as input/output pairs, each consisting of a vector from an input space and a desired network response. Using a stochastic gradient algorithm,

the network adjusts its weights so that the error between the actual and desired response is minimized.

Fig. 1 Structure of training stage.

Fig. 2 Structure of a RBFNN.

As shown in Fig. 2, the output of the RBFNN is calculated as follows:

$$\hat{y}_i = \sum_{k=1}^N w_k^i \Theta_k(\|x - c_k\|_2), \quad (7)$$

where w_k^i is the weight between the k th node in the hidden layer and the i th output node, where $k=1, 2, \dots, N$ and $i=1, 2, \dots, m$, N is the number of hidden nodes and m is the number of output nodes, $x \in \mathbb{R}^{16 \times 1}$ is an input vector, c_k is the k th center node in the hidden layer, $\|\bullet\|_2$ denotes the Euclidean norm and $\Theta_k(\bullet)$ is a non-linear transfer function for the k th center. In this paper, the RBFNN has 16 input nodes and 14 output nodes, as shown in Fig. 2.

As shown in (7), $\Theta_k(\bullet)$ is a non-linear function that transforms the output of the k th center to an adequate value. In this paper, the most widely used Gaussian function is adopted, as follows:

$$\Theta(x) = \exp\left(-\frac{(x-r)^2}{\sigma^2}\right), \quad (8)$$

where σ and r are the parameters that control the “width” and the “position” of the RBF centers, respectively.

The mean-squared error between the actual and the desired network output is used to measure the optimization criterion, as follows:

$$J(\sigma, r, w) = \frac{1}{2} \sum_{i=1}^m (y_i - \hat{y}_i)^2 = \frac{1}{2} \sum_{i=1}^m (y_i - \sum_{k=1}^N w_k^i \Theta_k(\|x - c_k\|_2))^2 \quad (9)$$

where y_i is the i th desired output and $\hat{y}_i$ is the i th actual output.

Equations (8) and (9) show that three sets of parameters govern the optimization performance of the network: the position of the RBF centers, the width of the RBFs and the weights, w_k^i , between the hidden and the output layers. In this paper, a HSA algorithm is used to optimize these three sets of parameters.

(b) The optimization of parameters using a harmony search algorithm

A HSA [10]-[11] operates similarly to the process for musical improvisation, whereby musicians tune the pitches of their instruments to achieve better harmony. Similarly to other optimization algorithms, a HSA uses a population-based random search to achieve a global optimum point and is used in many fields. It has been verified to be more efficient than differential evolution (DE) and particle swarm optimization (PSO) methods [12]. A HSA tunes the position of the RBF centers, the width of the RBFs and the weights between the hidden and the output layers, as follows.

1) Initialization

In this step, an initial harmony matrix (HM) is randomly generated as follows:

$$HM = \begin{bmatrix} u_1^1 & u_2^1 & u_3^1 \\ u_1^2 & u_2^2 & u_3^2 \\ \vdots & \vdots & \vdots \\ u_1^{H-1} & u_2^{H-1} & u_3^{H-1} \\ u_1^H & u_2^H & u_3^H \end{bmatrix}, \quad (10)$$

where u_1^j is the position of the j th RBF center, and $j=1, 2, \dots, H$, H is the size of the harmony matrix, u_2^j is the width of the j th RBF, $u_3^j (= w_k^j, i=1, 2, \dots, m$ and $k=1, 2, \dots, N)$ are the weights of the j th vector between the hidden and the output layers.

As shown in (10), each row ($[u_1^j, u_2^j, u_3^j]$) represents a vector solution that is composed of a set of decision variables. In this paper, u_3^j is randomly generated in the range 0 to 1, u_1^j and u_2^j are generated as follows:

$$u_h^j = u_{\min}^j + rand \times (u_{\max}^j - u_{\min}^j), \quad (11)$$

where $h=1, 2$ and $j=1, 2, \dots, H$, $u_{\min}^j$ and $u_{\max}^j$ are the lower and upper bounds of u^j and $rand$ is a uniform distribution value that ranges between 0 and 1. Equation (11) shows that there are $2+N \times 14$ variables in a vector solution, where N is the number of center nodes and “14” is the number of output nodes.

2) Calculation of the objective function

For each row vector of HM , the objective value is evaluated using (9). The solution with the lowest objective value in HM is regarded as the optimal solution for the RBFNN.

3) The generation of a new harmony matrix

Using the memory consideration, the pitch adjustment and random selection processes, a new harmony matrix $U'=[u_1', u_2', u_3']$ is generated. The decision variables in U' are updated by the difference ($u^1 - u^H$) in the memory consideration. The harmony memory considering rate ($HMCR$) is used to improve the diversity of the solution vector, as follows:

$$u_i' = \begin{cases} u_i^1 \in [u_i^1, u_i^2, \dots, u_i^H], & \text{if } rand \leq HMCR \\ u_i' \in U_i, & \text{otherwise} \end{cases} \quad (12)$$

where $HMCR \in [0,1]$ is set by operators and $U_i \in [u_{i,\min}, u_{i,\max}]$ is the set of the possible range of values for each decision variable. If $HMCR$ is set at 0.9, there is a 90% probability that the HSA chooses decision values from historically stored values and a 10% probability that it chooses decision values from the entire range.

The pitch adjustment that is used to improve the vector solution is checked when a new harmony matrix is obtained using memory consideration, as follows:

$$u_i' = \begin{cases} u_i' \pm rand \times BW & \text{if } rand \leq PAR \\ u_i' & \text{otherwise} \end{cases} \quad (13)$$

where $PAR \in [0,1]$ is the pitch adjusting rate and BW is the bandwidth of the variations.

4) Parameter adjustments

In the parameter adjustment process, PAR and BW are tuned to improve the performance of the HSA, as follows:

$$PAR(iter) = PAR_{\min} + \frac{PAR_{\max} - PAR_{\min}}{NI} \times iter \quad (14)$$

$$BW(iter) = BW_{\max} \exp\left(\frac{\ln(BW_{\min}/BW_{\max})}{NI} \times iter\right) \quad (15)$$

where $PAR_{\min}$ and $PAR_{\max}$ are the lower and upper bounds of PAR , $BW_{\min}$ and $BW_{\max}$ are the lower and upper limits of BW , $iter$ is the number of iteration and NI is the number of improvisations so far in the optimization process. The improvisations allow the best objective value of the offspring harmony matrix to be better than that of the parent harmony matrix.

5) The selection of the best vector solution

After adjusting the parameters, the size of HM is maintained at $2H$. H sets of the better row vector in the harmony matrix are then selected. Processes 3 and 4 are repeated, until the maximum number of iterations is reached. The best vector solution in HM is regarded as the optimal solution for the RBFNN.

(c) Model selection for forecasting using fuzzy inference

Fuzzy inference uses *If-Then* statements to invoke each fuzzy rule, where *If* is the premise and *Then* is the consequence. The basic form of fuzzy inference is as follows:

$$\begin{aligned} R^i: & \text{If } x_1 \text{ is } A_{i1} \text{ and } x_2 \text{ is } A_{i2} \text{ and } \dots \text{ and } x_n \text{ is } A_{in}, \\ & \text{Then } y_i = f_i(\bullet) \end{aligned} \quad (16)$$

where R^i is the i th fuzzy rule, $x_1, \dots, x_n$ are the premise (or input) variables, $A_{i1}, \dots, A_{in}$ are the linguistic values related to the inputs for the i th rule and $f_i(\bullet)$ is the consequent (or output) variable of the i th rule. In general, $f_i(\bullet)$ is a nonlinear transfer function of the premise variable.

In this paper, fuzzy inference is used to select an adequate model for accurate forecasting. The input variables of fuzzy inference comprise the maximum PV power output on the latest similar day, the prediction of the maximum average temperature for the next day by the TCWB and the prediction of the probability of precipitation for the next day by the TCWB. The output variables are the models of the different weather days. Figures 3-5 show the fuzzy input variables and their associated membership functions which are tuned by trial-and-error experiments to achieve the best performance. Figure 6 shows the output variables for different weather types.

As previously described, each fuzzy input variable is partitioned into three regions. Therefore, there are 27 ($3 \times 3 \times 3$) fuzzy rules in the knowledge base. The results for the rules that fire are de-fuzzified to a crisp output value in the de-fuzzification stage. In this paper, the popular centroid method [13] is used to integrate the fuzzy output.

Fig. 3 The input variable of maximum PV power output.

Fig. 4 The input variable of maximum average temperature.

Fig. 5 The input variable of probability of precipitation.

Fig. 6 The output variables of different weather types.

4. Numerical Results

4.1 Data collection

The proposed method is used to forecast the one-day-ahead hourly PV power output for a 5 kWp system. To verify the superiority of the proposed approach, the traditional RBFNN and ANN methods are also tested using the same database. The practical PV power generation data

and the irradiance values used were from 2013 May 1st to 2014 April 30th. The data was sampled every 5 minutes. The hourly PV power output and the irradiance data are then obtained by averaging the data collected over one hour.

To verify the proposed approach, the data for the last five days of each month was used for testing and that for the other days are for training. For comparison purposes, the traditional RBFNN and ANN also use the same input/output structure, while the number of intermediate layers remained to be determined independently.

4.2 Evaluation criteria

To evaluate the performance of each method, the criteria for the mean relative error (MRE) and root mean square error (RMSE) are used, as follows [14]:

$$MRE = \frac{1}{N_f} \sum_{i=1}^{N_f} \frac{|P_{fore} - P_{true}|}{P_{total}} \times 100\% \quad (18)$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N_f} (P_{fore} - P_{true})^2}{N_f}} \quad (19)$$

where P_{fore} is the forecasting value, P_{true} is the real value, P_{total} is the capacity of the PV system, and N_f is the number of forecasting points.

4.3 Forecasting results

Figure 7 shows the forecasting results for Aug. 27 to Aug. 31 of 2013. The weather on Aug. 29 was rainy, so there was a very low PV power output. The MRE value is 0.3688% for the proposed method and the RBFNN and ANN give values of 0.4362% and 0.6496%, respectively. For the RMSE value, the proposed method gives a value of 468.00 (W), the RBFNN gives a value of 524.18 (W) and the ANN gives a value of 733.98 (W). The proposed method gives a better forecast for PV power output. Figure 8 shows the forecasting results for Nov. 26 to Nov. 30 of 2013. Nov. 26 and Nov. 27 were rainy and the other days were sunny. The RBFNN and the ANN methods give a worse forecast for Nov. 26 and Nov. 27. The proposed method produces a better forecast than the other methods, except for Nov. 28.

Figure 9 shows the forecasting results for Feb. 24 to Feb. 28 of 2014. These days were sunny and the PV power outputs were stable. However, the RBFNN and ANN methods produce lower forecasts from Feb. 24 to Feb. 27. Fig. 10 shows the forecasting results for Apr. 26 to Apr. 30 of 2014. During these days the weather was unstable. Although the proposed method gives a better forecast than

the other methods, the forecasting errors are higher than those for the other months.

Table 1 summarizes the forecasting results for each method. The average MRE, using the one-year testing data, is 0.3148% for the proposed method and the RBFNN and ANN give values of 0.4802% and 0.5968%, respectively. For the average RMSE, the proposed method gives a value of 370.36 (W), the RBFNN gives a value of 526.69 (W) and the ANN gives a value of 673.50 (W). The results show that the proposed method gives a better forecasting result than the existing methods.

Fig. 7 Forecasting results for Aug. 27 to Aug. 31 of 2013.

Fig. 8 Forecasting results for Nov. 26 to Nov. 30 of 2013.

Fig. 9 Forecasting results for Feb. 24 to Feb. 28 of 2014.

Fig. 10 Forecasting results for Apr. 26 to Apr. 30 of 2014.

Table 1 Summary of forecasting results for each method.

Month	MRE(%)			RMSE(W)		
	ANN	RBFNN	Pro.	ANN	RBFNN	Pro.
May,2013	0.4674	0.3586	0.1350	607.96	439.71	158.59
June,2013	0.4673	0.2591	0.1388	629.37	342.55	165.97
July,2013	0.6200	0.3943	0.2057	649.68	398.40	195.81
Aug.,2013	0.6496	0.4362	0.3688	733.98	524.18	468.00
Sep.,2013	1.1546	0.6472	0.5035	1198.7	655.51	611.45
Oct.,2013	0.4189	0.1784	0.1777	570.41	225.84	216.72
Nov.,2013	0.4231	0.5617	0.3698	346.57	425.22	337.99
Dec.,2013	0.6263	0.7361	0.4546	464.35	625.45	452.05
Jan.,2014	0.5100	0.4262	0.2903	786.35	694.03	495.81
Feb.,2014	0.5780	0.5821	0.2248	849.26	804.66	365.16
Mar.,2014	0.4186	0.4068	0.3791	569.34	565.86	541.43
Apr.,2014	0.8276	0.7756	0.5290	676.06	618.89	435.3
Max.	1.1546	0.7756	0.5290	1198.7	804.66	611.4
Min.	0.4186	0.1784	0.1350	346.57	225.84	158.5
Ave.	0.5968	0.4802	0.3148	673.50	526.69	370.36

5. Conclusions

This paper presents a novel approach for the one-day-ahead hourly forecasting of PV power output. The proposed approach comprises data classification, training and forecasting stages. The data classification stage uses a fuzzy k-means clustering algorithm to classify the daily PV power output patterns into different weather types. In the training stage, five training models, representing different weather types, are created using a HSA-based RBFNN. The forecasting stage uses fuzzy inference to invoke an adequate model for accurate forecasting, according to the weather predictions of the TCWB. For one complete whole year data, the proposed approach achieves better forecasting accuracy than the RBFNN and ANN methods, in terms of MRE and RMSE criteria. Although only five models are considered to establish the forecasting model, extension of the proposed approach to consideration of

more models is feasible to increase the forecasting accuracy.

Acknowledgment

Financial supports from the Ministry of Science and Technology, Taiwan, R.O.C. under the Grant No. 103-2221-E-168-015 and 101-2632-E-230-001-MY3 are acknowledged.

References

- (1) L. Martín, L. F. Zarzalejo, J. Polo, A. Navarro, R. Marchante, and M. Cony, "Prediction of global solar irradiance based on time series analysis: application to solar thermal power plants energy production planning," *Solar Energy*, 2010, vol. 84, pp. 1772–1781.
- (2) G. Capizzi, C. Napoli, and F. Bonanno, "Innovative second-generation wavelets construction with recurrent neural networks for solar radiation forecasting," *IEEE Trans. Neural Network and Learning Systems*, 2012, vol. 23, no. 11, pp. 1805–1815.
- (3) K. Tanaka, K. Uchida, K. Ogimi, T. Goya, A. Yona, T. Senjyu, T. Funabashi, and C. H. Kim, "Optimal operation by controllable loads based on smart grid topology considering insolation forecasted error," *IEEE Trans. Smart Grid*, 2011, vol. 2, no. 3, pp. 438–444.
- (4) P. Zhang, H. Takano, and J. Murata, "Daily solar radiation prediction based on wavelet analysis," *SICE Annual Conference*, Waseda University, Tokyo, Japan, Sep. 2011, pp. 13–18.
- (5) M. G. De Giorgi, P. M. Congedo, and M. Malvoni, "Photovoltaic power forecasting using statistical methods: impact of weather data," *IET Sci. Meas. Technol.*, 2014, vol. 8, no. 3, pp. 90–97.
- (6) J. Shi, W. J. Lee, Y. Liu, Y. Yang, and P. Wang, "Forecasting power output of photovoltaic systems based on weather classification and support vector machines," *IEEE Trans. Industry Applications*, 2012, vol. 48, no.3, pp. 1064–1069.
- (7) S. P. Lloyd, "Least squares quantization in PCM," *IEEE Trans. Inform. Theory*, 1982, vol. 28, no.2, pp.129–137.
- (8) Z. Huang, "A fuzzy k-modes algorithm for clustering categorical data," *IEEE Trans. Fuzzy Systems*, 1999, vol. 7, no. 4, pp. 446–452.
- (9) F. M. Ham and I. Kostanic, "Principal of Neurocomputing for Science and Engineering," 2000,

New York: McGraw-Hill.

- (10) Z. W. Geem, J. H. Kim, and G. V. Loganathan, "A new heuristic optimization algorithm: harmony search," *Simulation*, 2001, vol. 76, no.2, pp. 60-68.
- (11) M. Mahdavi, M. Fesanghary, and E. Damangir, "An improved harmony search algorithm for solving optimization problems," *Appl. Math. Comput.*, 2007, vol. 188, no. 2, pp. 1567-1579.
- (12) C. M. Huang and Y. C. Huang, "Hybrid optimisation method for optimal power flow using flexible AC transmission system devices," *IEE Proc. Gener. Transm. Distrib.*, 2014, doi: 10.1049/iet-gtd.2014.0096.
- (13) T. J. Ross, "*Fuzzy Logic with Engineering Applications*," 1995, McGraw-Hill, U.S.
- (14) C. S. Ioakimidis, H. Eliasstam, and P. Rycerski, "Solar power forecasting of a residential location as part of a smart grid structure," *Energytech*, 2012, pp. 1-6.