2014 International Joint Conference on Neural Networks (IJCNN)
July 6-11, 2014, Beijing, China

Wind Power Forecasting - An Application of Machine learning in Renewable Energy

Gul Muhammad Khan, Jawad Ali and Sahibzada Ali Mahmud
Center for Intelligent Systems and Networks Research,
Department of Electrical Engineering, University of Engineering and Technology
Peshawar, Pakistan 25000
Email:{gk502, jawad.ali, sahibzada.mahmud}@nwfpuet.edu.pk

***Abstract*— The advancement in renewable energy sector being the focus of research these days, a novel neuro evolutionary technique is proposed for modeling wind power forecasters. The paper uses the robust technique of Cartesian Genetic Programming to evolve ANN for development of forecasting models. These Models predicts power generation of a wind based power plant from a single hour up to a year - taking a big lead over other proposed models by reducing its MAPE to as low as 1.049% for a single day hourly prediction. Results when compared with other models in the literature demonstrated that the proposed models are among the best estimators of wind based power generation plants proposed to date.**

I. Introduction

THE environmental hazards like pollution and global warming have compelled the power generation sectors to come up with different green energy production techniques. The depletion of natural resources like petroleum, coal and natural gas which are the main sources of thermal power generation has also forced the power generation industry to work with renewable power generation sources. Wind power production sector is one of them. According to the European Wind Energy Association (EWEA) report, the European Union (EU) installed 9.616GW wind based power plants during 2011. In 2012, EU further installed wind based power plants having capacity of 11.66GW thus achieved the total capacity of 105.6GW by the end of 2012 [1]. This meets 7% of the Europe's electricity demand [2]. Considering these facts, one must adopt techniques that are feasible and provide optimum forecasting results regarding wind power generation. Estimation of future production of wind power is one of them. Techniques used for forecasting wind power generation includes stochastic, probabilistic and machine learning techniques. The application of machine learning techniques is on a leading edge due to its self-learning nature and adaptability. The forecast, made by machine learning techniques, covers a time span of as short as 10min and can go up to years as proposed in the paper. The paper circles around the theme of short term load forecasting that is the basic ingredient in the application of Real Time Load Forecasting (RTLF).

Section II gives a review of wind power forecast and various techniques that are used for modeling forecasters for wind power. It also explains concepts regarding Genetic Programming, CGP, Neuro-evolution and Artificial Neural Networks. Section III describes CGPANN in brief while Section IV covers experimental setup. Section V contains analysis and validation results whereas the whole work is summarized in Section VI.

The authors would like to express their gratitude for the generous funding and resources provided by the National ICT R&D Fund Pakistan.

II. Literature Review

A. Wind Power Forecasting

The energy production forecast of a wind based power plant is associated with the wind speed, seasonal changes and wind power plant characteristics. Research has explored different time horizons for Wind Power Forecasting using various techniques. Adaptive Neuro-Fuzzy Interface Systems (ANFIS) is used for short term Wind Power Forecasting in [3] considering 150 turbines, each having the capacity of 750KW and producing 112.5MW power instantly. Different models for Wind Power Forecasting is highlighted in [4] including GH Forecaster, Alea Wind, SOWIE, WPMS and WEPROG. First and Second order Markov Chain Models are employed for Very Short-term probabilistic Wind Power Forecasting having time intervals of 10min in [5]. The same method can be applied for different time spans but the over all technique is not generic i.e. one model can't be employed directly for its another time series prediction application. In [6] different error calculations methods are revised that are being used in the field of Wind Power Forecasting supervised by statistical models. 10min and hourly Wind Power Forecasting is done in [7] using 10min persistence, 10min averaging, hourly persistence and direct ANN method. The minimum NRMSE in the prediction Models is 9.52% for 10min averaging and 10.45% for Hourly direct ANN. Radial Basis Function (RBF) Network is used for 1 hour interval prediction of Wind Power and wind Speed Forecasting in [8]. Coefficient of determination, that is a statistical method used to find the accuracy, is used to find the exactness of the model during the evaluation. Further, the minimum Mean Absolute Error (MAE) is found to be 0.69m/s and 9.66W for wind speed and wind energy forecasting respectively. Enhanced Particle Swarm Optimization and Modified Hybrid Neural Network are used for Wind Power Prediction by the proposed New Forecaster Engine in [9] estimating power production by the Model having minimum Mean Absolute Percentage Error of 2.12% and RMSE of 4.18% for a month. [10], [11] has employed Support Vector Machine (SVM) and Wavelet SVM for hourly wind power forecast giving a Mean Absolute Percentage Accuracy of 98.92% and 96.2% respectively. Wind power forecasting uses RBF Neural Network in [12] for predicting wind power from 1 hour up to 60hrs. The effectiveness of this RBF Neural Network is satisfactory for short time series prediction. The model proposed in [13] is based on Artificial Neural Network that predicts short-term wind power showing RMSE below 20%. Time series analysis of the historical data of power production is made using Autoregressive integrated moving average (ARIMA) and Neural Networks [14] for Wind Speed Forecasting. The paper uses the data that is used in [15] for 1 hour ahead Wind Power Forecasting for two different farms. The best model that is proposed in [15] is having an RMSE of 5.8%.

B. Cartesian Genetic Programming

Cartesian Genetic Programming (CGP) is an effective genetic programming method that is developed by J. F. Miller and Thomson [16], [17]. CGP utilizes a two dimensional programming architecture that is incorporated by nodes or genes. Its architecture is a two dimensional directed graph inspired from traditional FPGA that evolves digital circuits [18]. The genetic behavior of learning that leads the specimen to evolve, is achieved through feed forward mechanism. Node, the basic constituent of CGP, consists of logical functions such as NOR, NAND, NOT or an arithmetic function. Evolutionary strategy of $1 + \lambda$ is used for producing offspring for the next generation, where λ is the number of offspring. Mutation is used as a parameter for producing offspring.

In CGP, a genotype is represented by a string of integers with the corresponding phenotype a two dimensional nodal network. The genotype is evolved by changing the connectivity and functions of nodes in the network, thus obtaining a range of topologies.

C. Neuro-Evolution

The term Neuro-Evolution (NE) is concerned with the evolution of various characteristics of an ANN. ANN is the behavioral interpretation of natural nervous system thus ANNs evolve in the same way as living being's neural system. Basic parameters of an ANN includes node functions, number of inputs, input weights, node connections and network topology. Varying these network parameters, counts to Neuro-Evolution in ANN while fixing these ANN attributes stops the evolutionary process. The evolution of a genotype results in obtaining the desired phenotype behavior. The evolution of a genotype can either be due to variation in a single network attribute or it can be accomplished by changing multiple network parameters. For example only varying connection weight in a genotype can restrict the network performance. On the other hand, varying multiple network attributes like node functions, number of inputs, input weights, node connections and network topology enhances the network performance without any restriction.

Various evolutionary techniques have been discussed in [19] including TWEANN. An evolutionary design systems for ANNs-EPNets has been developed in [20]. NEAT-evolved ANN are applied in [21] for object class recognition while bio-signal processing is dealt in [22] using CGPANN. Each of the mentioned work in [21] and [22] explains the dynamic change of network topologies for different environments.

III. Cartesian Genetic Programming Evolved Artificial Neural Network (CGPANN)

ANN is always credited for its dynamic characteristics that includes self-modifying and adoptable network architecture. The variable space of ANN changes dynamically with the environment thus it leads to efficient results when it is evolved using CGP which is a two dimensional grid based architecture [19].

A feed forward CGPANN node consists of input connections, connection weight and node function. The Node is considered as input node if its input is only from outside the network. An intermediate node takes its input from the preceding node(s) as well as input to the system, whereas the system output can be from system inputs or any node in the system. A typical CGPANN Node is shown in Fig. 1 The

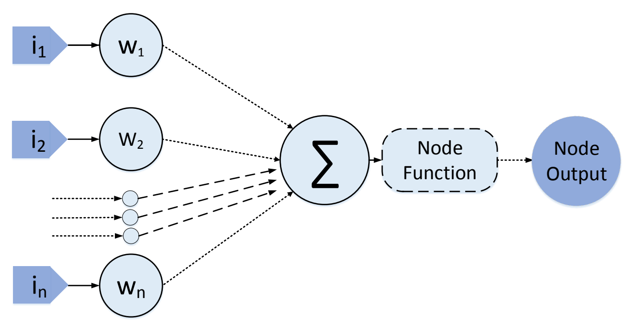

Fig. 1. Structure of a Typical CGPANN Node

selection of the inputs is from the input array I, such that

$$I = \{i_1, i_2, i_3, ...i_n\} \tag{1}$$

using Psuedo-Random Generator (PRG). The weight Matrix W, whereas

$$W = \{W_1, W_2, ...W_n\} \tag{2}$$

consists of randomly generated values between -1 and +1.

A summing junction in ANN can be represented by

$$y' = \sum_{i=1}^{N} x_i \tag{3}$$

where x_i is the input to the junction. If the same input is scaled with a randomly assigned weight w_i then

$$y' = \sum_{i=1}^{N} x_i w_i \tag{4}$$

Let for N inputs to a node, we have y_j output such that

$$y_j = f^j(y'_j) = f^j\left(\sum_{i=1}^{N} x_i w_i\right) \tag{5}$$

Here f is an activation function, particular to node j. If total number of nodes in the network are N_T then j is defined by

$$\{j|j\epsilon N \wedge 1 \leq j \leq N_T\} \tag{6}$$

Let I be the input set to a unique genotype network G_k, consisting unique entries i such that, for all inputs

$$i\epsilon R \wedge 0 \leq i \leq 1 \tag{7}$$

in

$$I = \{i_1, i_2, i_3, ...i_n\}$$

so, network G_k is the a set of random selection from inputs $[I]$, outputs of nodes y_j, for a single output O_p such that

$$O_p = \frac{1}{n}\sum_{i=1}^{N}\Bigg(f\bigg(\sum(y_jW_j + y_{j-1}W_{j-1} + ... + y_1W_1 + IW_k)\bigg)\Bigg) \tag{8}$$

here, W_j, W_{j-1}, ... W_1 are Random subsets of W such that W_k is a subset W_j and

$$W_j = \{w_k|w_k\epsilon R \wedge -1 \leq W_k \leq 1\} \tag{9}$$

Now

$$G_k = \{I, y_j, y_{j-1}, ..., y_1, O_p\} \tag{10}$$

$$\begin{aligned} y_j &= \Bigg(f\bigg(\sum(y_{j-1}W_{j-1} + y_{j-2}W_{j-2} + ... + y_1W_1 + IW)\bigg)\Bigg) \\ y_{j-1} &= \Bigg(f\bigg(\sum(y_{j-2}W_{j-2} + y_{j-3}W_{j-3} + ... + y_1W_1 + IW)\bigg)\Bigg) \\ &\vdots \\ y_2 &= \Bigg(f\bigg(\sum(y_1W_1 + IW)\bigg)\Bigg) \\ y_1 &= \Bigg(f\bigg(\sum IW\bigg)\Bigg) \end{aligned} \tag{11}$$

Let G_k and G_l be the two successive genotypes. G_l is produced from G_k by mutating $\mu\%$ weights of connections in $G - k$, nodal connectivity, functions defining each node or the combination of these.
Let the total entries in G_k, including weights, connections and functions be N_{cwf}
then

$$N'_{cwf} = \mu \times N_{cwf} \tag{12}$$

N'_{cwf} represents genotype entries that are to be randomized to get G_l.
Let r be a unique entry to the set ζ that contains value to be changed to get mutated G_k or G_l defined by r.

$$\{r|r\epsilon\zeta\forall(W, I, y_j)\} \tag{13}$$

each value in r is defined by a Pseudo Random Generator (PRG) that takes N'_{cwf} values from available N_{cwf} entries using the relation

$$\Big\{r_i|r_i\epsilon\{1, 2, 3, ..., N_{cwf}\} \wedge r_i \subset \{1, 2, 3, ..., N_{cwf}\}\Big\} \quad i = 1, 2, 3, ...N'_{cwf} \tag{14}$$

where r_i can its value from $\{1, 2, 3, ...N_{cwf}\}$ so each entry in ζ is

$$\Gamma(i) = \Big\{r_i|r_i\epsilon\{1, 2, 3, ...N_{cwf}\} \wedge \{1 \leq r_i \leq N_{cwf}\}\Big\} \tag{15}$$

The value that is replaced in G_k using $\Gamma(r)$ for N'_{cwf} values uses Theorem 1, proven by the author, to assign values randomly.

Theorem 1 (Pseudo-Random-Number-Generator): A generator $\mu : \{0,1\}^a \rightarrow (\{0,1\}^b)^\phi$ is pseudo random in characteristics within a space μ with a size ρ as the block size and ϵ as a parameter if for each Finite Machine State R having size 2^n over the character $\{0,1\}^b$, we have

$$|Probability_y[y \mapsto R] - Probability_x[\mu(x) \mapsto R]| \leq \epsilon \tag{16}$$

where y is a uniform random choice from $(\{0,1\}^b)$ and x is a random choice from $\{0,1\}^a$.

Feed forward CGP evolved ANN is shown in Fig. 2. The Network takes inputs i_1, i_2, i_3 upto input i_n, defined by the algorithm. The input layer comprised of Nodes $1, 2, 3$ upto n. It is directly taking the Network inputs. The outputs from the input layer is fed into intermediate nodes that are also called processor nodes. Intermediate nodes either perform Arithmetic or logical operations on the feed (data from input layer) or passes it to the output layer nodes depending upon the nature of output. Junk nodes, that contributes nothing to the network performance by not taking part in data processing are also shown in Fig. 2.

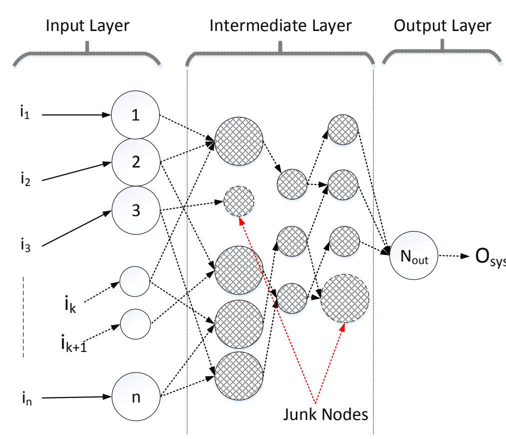

Fig. 2. Feed forward CGP evolved Artificial Neural Network

Log-sigmoid function is used as the activation function, given by Eq. 17

$$f(x) = \frac{1}{1 + e^{-x}} \tag{17}$$

Here x is input to the activation function $f(x)$. x is further defined by the Eq. 18

$$O(g, c, j) = \sum_{i=1}^{N} w(i, c).I(g, c, i) \tag{18}$$

Where N is the number of inputs to the given node j for n = {1, 2, 3,.., N} numbered inputs.

The total network inputs are K in set $k = \{1, 2, 3, ..N, ...K\}$. g defines a specific genotype in a population and c is the respected node that is under operation.

$w(i, c)$ is the pseudo-random weight assigned to each input $I(g, c, i)$ as given by Eq. 2.

Also $I(g, c, i)$ is defined by Eq. 19 as a randomly selected input to the node j.

$$\begin{aligned} I(g, c, j) =& PRG([I(g, c, 1), I(g, c, 2), ..., I(g, c, N) \\ & ..., I(g, c, k)] : [O(g, c, 1), O(g, c, 2), \\ & ....O(g, c, j - 1)]) \end{aligned} \tag{19}$$

CGPANN uses 1 + λ evolutionary strategy where as λ represents the number of offspring having their computational characteristics slightly different from its parent and produced using mutation. In proposed model, λ is set at 9. Amongst the 1 + λ number of ANNs, the fittest network is treated as parent in the next generation and the same criteria is applied to produce its offspring by taking into account the network parameters and node functions. The historical data of hourly produced wind power plant situated in Galicia [23], Spain, is used for the validation of the proposed model. The same data source is used in [15] for estimation purposes. A one year hourly spaced dataset, starting from August 1st, 2005 up to July 31st, 2006, is used to train the network models. The networks take a single day, 15 days and 30 days instances of hourly spaced time series as its input and predicts the power production of the site for the next instance. Sliding window mechanism (Fig. 3) is then used for further prediction. The mutation rate (μ_r) is kept at 10% due to its better outcomes as proposed in [24].

The time series data is used for two purposes. In the beginning, a population of defined number of neural networks is randomly generated. In our case it is 10. A fixed number of time series entries is given to the ANN and the output(s) are compared with the actual data of the time series. This is called training session. Here, Mean Absolute Error (MAPE) for outputs of ANN are compared and the fittest ANN is used for further mutation and generation as discussed in section III. The MAPE is given by Eq. 20.

$$MAPE = \frac{1}{N} \sum_{i=1}^{N} \frac{|P_{if} - P_{iA}|}{P_{iA}} \times 100\% \tag{20}$$

Where P_{if} is the forecasted wind power, P_{iA} is Actual load experienced at that same instant and N is the total duration for which forecasting is made. The FITNESS is given by the Eq. 21.

$$FITNESS = 100\% - MAPE \tag{21}$$

When the phenotype of the desired fitness is obtained, it is provided with the dataset from the time series for its evaluation and thus prediction.

Fig. 3 graphically explains the sliding window mechanism for prediction of the 6th element in the given time series. It takes 5 inputs i.e. I_1, I_2, I_3, I_4 and I_5 in the first computation for predicting P_1 as window size in this case is 5. The window slides to the next level having its inputs range from I_2 to I_6. The CGPANN thus predicts P_2 in the second level

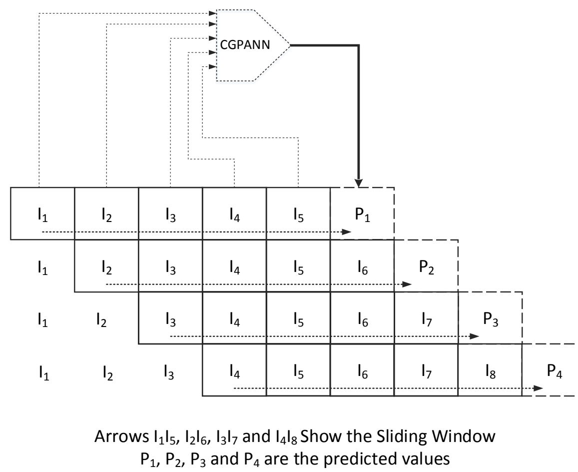

Fig. 3. Sliding Window Mechanism - Predicting Single output by taking 5 inputs

of the sliding window. The dotted arrows I_1P_1, I_2P_2, I_3P_3 and I_4P_4 represent the total span of the window for a single prediction.

IV. Performance and Evaluation

The proposed CGPANN, forecaster model has been evaluated using hourly spaced power production data of the Sotavento, Galicia wind farm for the years 2012. MAPE and NRMSE has been used as an evaluation for the proposed forecaster. The networks take 24 entries, 360 entries and 720 entries for the prediction of next single hour. Here, the number of nodes on the network are varied between 50 and 500 with a step size of 50. Fig. 4 pictures the relation between input arguments and output of the network. The network size has been reduced from 150 starting nodes (initialized manually) to 32 nodal phenotype. The connectivity of each node is in forward direction starting from the least integer and a non conventional increment. Each connection has a randomly defined weight and is performing regular summation as indicated in Fig. 1.

Sliding window mechanism is used for further predictions spanning up to a year. The results of the training session for each node and for each network is tabulated in Table IV.

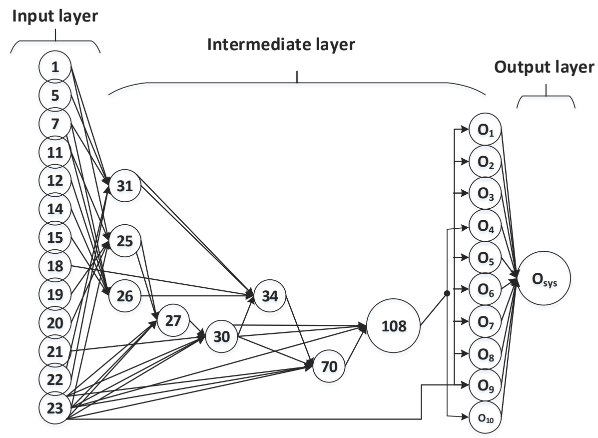

Fig. 4. Phenotype evolved from 150 starting nodes in ANN using CGP

Table IV evidences the fact that each network gives different results while varying the number of nodes of the network improving the MAPE values for the year 2006.

In the evaluation, an independent dataset is given to the network for its validation. In our case this is the hourly wind power production data for year 2012. MAPE is then calculated for each hour, day, week, month and year for the estimated values.

Table IV contains the yearly MAPE for year 2012. It is evident from the results in Table IV that results vary dynamically while varying the nodes' count and the number of inputs to the proposed CGPANN. When the number of inputs is large, fewer nodes can give accurate results as in the case of 30days input CGPANN Model. It goes up to 5.79% in this case.

The yearly MAPE value for 24hrs input network is 4.718% which is far less than the MAPE results of the network with 30days inputs. This clarifies that when time span is increased in terms of input variables, several other random environmental, technical and seasonal issues effects the forecasting of a forecaster model.

Monthly MAPE values for 24hrs input CGPANN model are given in Table IV. The MAPE in the month of March reaches up to 3.089% which is 23% more consistent if it is compared with the yearly results of the same data input model but with 300 nodes' network .

Minimum MAPE values for a single week are tabulated in Table IV for number of nodes ranging between 50 and 500 with an increment of 50 nodes for each of the Networks. The best results are found on the 3rd week for the CGPANN model that takes single day, hourly spaced generated wind power data input for predicting the next hour. The weekly MAPE is calculated to be **1.58%** for the estimations of this 150 node single day input CGPANN model. Here, the improvement in MAPE value is about **60%** while shortening the span of the prediction models by decrementing the free run of the sliding window from 1 year to a single week.

The last column in Table IV lists the best estimated week of the year 2012. The results of single day input model are far better than that of 15 days input and 30 days input CGPANN model.

The MAPE values of each day for the weeks tabulated in Table IV are given in Table IV. Table IV further explains the fact that the CGPANN model gives better results for short term load forecasting irrespective of the total number of inputs to the Network models.

Figure 5 visually summarize the perfection of the proposed models. Hourly spaced normalized Kilo Watts predicted by the best model are plotted against the actual normalized KW for the year 2012 and for a single month. The normalized forecast wind power values are elevated by +0.3 for its better visualization. Both of the graphs start from 25th hour as the first 24 hours are used as input to the Model. This hourly data goes up to 8736 instances.

Table IV has compared the proposed model with the existing models, declaring the propose model to be the best amongst these proposed models for hourly prediction of Wind power for single day.

The NRMSE in Table IV is calculated using Eq. 22 as

$$NRMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\frac{P_{ia}-P_{if}}{P_{ia}})^2} \tag{22}$$

where P_{ia} is the observed power and P_{if} is the forecasted power at instance i. Also N is the hour(s) count for which the NRMSE is being calculated. In case of daily averaging, its value is 24.

V. CONCLUSION

Based on CGP evolved ANN, three different forecasting models have been proposed in the work. Each model is forecasting generating wind power for the next 1 hour. Estimation of the generating wind power has been made while a MAPE value of 4.71% has been achieved for a full one year. This number indicates the accuracy of the proposed Model. Weekly and monthly MAPE results also prove the fact that environmental and seasonal changes effect the proposed model to the least limits due to its self-adaptability and fast learning characteristics. The Accuracy of the model goes up to **98.951%** for single day prediction, evidencing the perfection of the proposed CGPANN Models for short term forecasting. Though the ruling contribution of the research work is to the power generation and power regulatory bodies but this proposed solution can be further enhanced by considering parameters such as wind-speed and its direction at the site, instantaneous humidity, atmospheric pressure and air temperature. Therefore a large area in the field of CGPANN for its application in wind power forecasting is still waiting for its exploration.

REFERENCES

[1] EWEA. European wind power statistics. [Online]. Available: http://www.ewea.org/statistics/european-statistics/

[2] T. E. W. E. Association, "Wind in power: 2012 european statistics," no. 1–30, 2012.

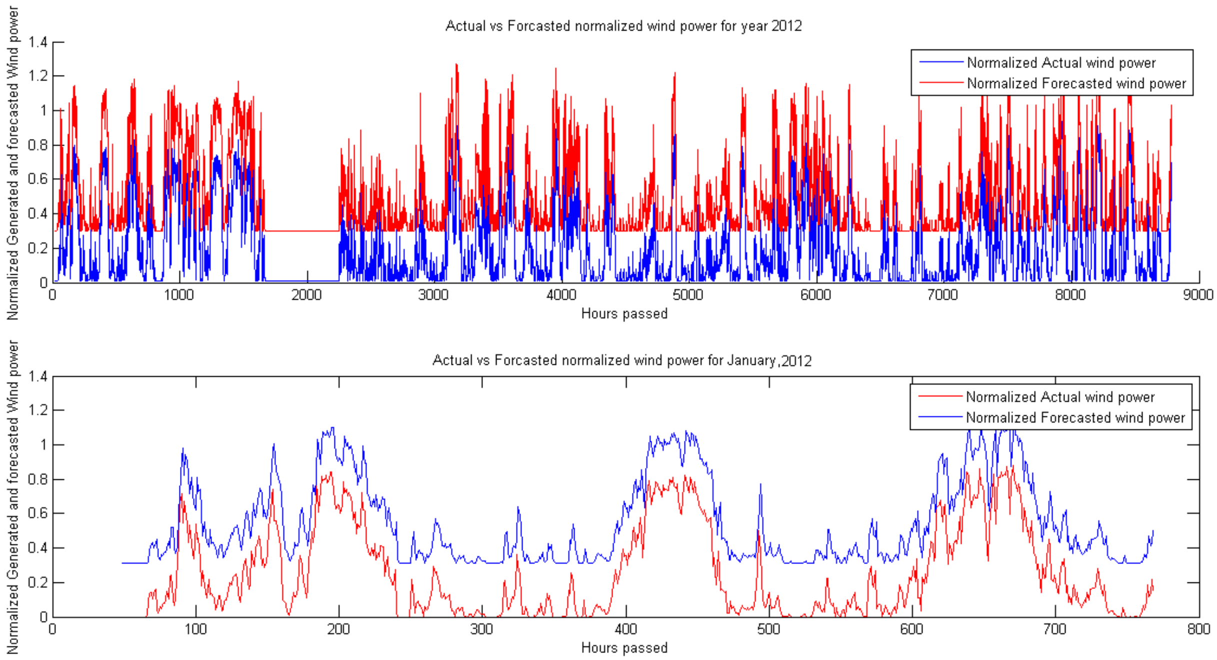

Fig. 5. Normalized Forecasted wind power verses Actual Generated Wind Power

[3] M. N. P. Johnson and K. M. Muttaqi, "Short term wind power forecasting using adaptive neuro-fuzzy inference systems," *Australasian Universities Power Engineering Conference, 2007. AUPEC*, pp. 1–6, 2007.

[4] P. G. L. A. M. Foley and E. J. McKeogh, "'wind power forecasting and prediction models," *IEEE, 9th International Conference on Environment and Electrical Engineering - Prague*, vol. 1, pp. 61–64, 2010.

[5] A. T. A. Carpinone, R. Langella and M. Giorgia, "'very short-term probabilistic wind power forecasting based on markov chain models," *IEEE 11th Conference on Probabilistic Methods Applied to Power Systems (PMAPS)*, pp. 107–112, 2010.

[6] Y. H. Jin Zhong and F. F. Wu, "Wind power forecasting and integration to power grids," *International Conference on Green Circuits and Systems (ICGCS)*, pp. 555–560, 2-10.

[7] J. L. L. D. Lijie Wang, A. Gerber and X. Liao, "Wind power forecasting for reduction of system reserve," *45th International Universities Power Engineering Conference (UPEC)*, pp. 1–5, 2010.

[8] Y. Y. Hong and C.-P. Wu, "'hour-ahead wind power and speed forecasting using market basket analysis and radial basis function network," *Proceeding of 2010 International Conference on Power System Technology (POWERCON)*, pp. 1–6, 2010.

[9] F. K. N. Amjady and H. Zareipour, "Wind power prediction by a new forecaster engine composed of modified hybrid neural network and enhanced particle swarm optimization," *IEEE Transactions on Sustainable Energy*, vol. 2, no. 3, pp. 265–276, 2011.

[10] J. Zeng and W. Qiao, "Support vector machine-based short-term wind power forecasting," *Proceedings of the IEEE (PES) Power System Conference and Exposition, Phoenix, Arizona*, 2011.

[11] ——, "Short-term wind power prediction using a wavelet support vector machines," *IEEE Transactions on sustainable energy*, vol. 3, no. 2, pp. 255–264, 2012.

[12] G. Sideratos and N. D. Hatziargyriou, "Probabilistic wind power forecasting using radial basis function neural network," *IEEE Transactions on Power Systems*, vol. 27, no. 4, 2012.

[13] A. S. M. Jabbari G., S. H. Gilani and H. Afrakhteh, "A new method for short-term wind power forecasting," *Proceedings of 17th Conference on Electrical Power Distribution Networks (EPDC)*, pp. 1–6, 2-12.

[14] J. R. J. C. Palomares-Salas, J. G. Ramiro and J. Melgar, "Arima vs. neural networks for wind speed forecasting," *IEEE International Conference on Computational Intelligence for Measuring Systems and Applications*, pp. 129–133, 2009.

[15] P. P. M. Poncela and J. R. Peran, "Automatic tuning of kalman filters by maximum likelihood methods for wind energy forecasting," *Applied Energy*, vol. 108, pp. 349–362, 2013.

[16] J. F. Miller and P. Thomson, "Cartesian genetic programming," *In Proceeding of the 3rd European Conference on Genetic Programming*, vol. 18, no. 2, pp. 121–132, 2000.

[17] J. A. Rothermich and J. F. Miller, "'studying the emergence of multicellularity with cartesian genetic programming in artificial life," *Proceedings of U.K. Workshop on Computational Intelligence*, 2002.

[18] B. G. G. Vamvakas and S. Perantonis, "Handwritten character recognition through two-stage foreground subsampling," *Pattern Recogn*, vol. 43, pp. 2807–2816, 2010.

[19] X. Yao, "Evolving artificial neural networks," *In Proceedings of the IEEE*, vol. 87, no. 9, pp. 1423–1447, 1999.

[20] X. Yao and M. M. Islam, "Evolving artificial neural network ensembles," *Computational Intelligence Magazine, IEEE*, vol. 3, no. 1, pp. 31–42, 2008.

[21] A. Hasanat, "Object class recognition using neat-evolved artificial neural network," *Fifth International Conference on Computer Graphics, Imaging and Visualization, CGIV '08*, pp. 271–275, 2008.

[22] A. M. Ahmad and G. M. Khan, "'bio-signal processing using cartesian genetic programming evolved artificial neural network (cgpann)," *10th International Conference on Frontiers of Information Technology (FIT)*, pp. 261–268, 2012.

[23] G. Sotavento. Sotavento wind farm data statistics. [Online]. Available: http://www.sotaventogalicia.com

[24] M. H. E. Barakat, M. Qayyum and S. A. Al-Rashed, "Short-term peak demand forecasting in fast developing utility with inherent dynamic load characteristics," *IEEE Transactions on Power Systems*, vol. 5, no. 1, pp. 813–824, 1990.

Nodes	24hrs input for predicting one year	15days input for predicting one year	30days input for predicting one year
50	5.197348326	**5.285745194**	6.499721648
100	5.009361269	5.885202194	**6.055184734**
150	4.901174734	5.915135162	6.414513325
200	**4.92469284**	5.752828466	6.323593239
250	5.008637518	5.352886353	6.350555627
300	5.196817585	5.309733938	6.438980128
350	4.924292382	6.32252627	6.760614222
400	5.55743624	5.363175172	6.576495275
450	5.176022548	5.953459225	6.273806938
500	4.926968658	6.1806760967	7.101056701

TABLE I

MAPE FOR TRAINING SESSION FOR PREDICTING OF A YEAR DATA USING HISTORICAL DATA OF 24HRS, 360HRS AND 720HRS FOR PREDICTING NEXT HOUR

Nodes	24hrs input for predicting one year	15days input for predicting one year	30days input for predicting one year
50	5.03070%	5.17784%	6.26146%
100	4.85882%	5.78293%	**5.79552%**
150	**4.71874%**	5.74162%	6.38897%
200	4.73554%	5.62914%	6.03870%
250	4.81306%	5.21547%	6.25025%
300	5.10188%	**5.15298%**	6.23876%
350	4.73802%	6.26090%	6.47594%
400	5.38251%	5.24453%	6.39620%
450	4.96082%	5.80705%	6.17640%
500	4.74677%	6.04693%	7.04533%

TABLE II

MAPE FOR EVALUATION SESSION FOR PREDICTING OF A YEAR DATA USING HISTORICAL DATA OF 24HRS, 360HRS AND 720HRS FOR PREDICTING NEXT HOUR

MONTHS	50	100	150	200	250	300	350	400	450	500
JAN	3.87	3.69	3.591	3.631	3.6758	3.8583	3.634	4.035	3.85	3.6
FEB	6.28	5.979	5.921	5.937	6.065	6.084	5.94	6.68	6.26	5.952
MAR	6.06	5.816	5.781	5.82	5.9	5.90	5.817	6.369	6.22	5.802
APR	5.793	5.736	5.543	5.549	5.63	6.053	5.538	6.449	5.723	5.582
MAY	5.497	5.261	5.23	5.261	5.381	5.341	5.266	5.762	5.58	5.25
JUN	5.179	5.026	4.927	4.9	5.03	5.217	4.95	5.67	5.22	4.93
JUL	3.366	3.251	**3.089**	3.15	3.159	3.460	3.161	3.47	3.33	3.11
AUG	4.119	4.021	3.89	3.94	4.022	4.177	3.947	4.39	4.221	3.91
SEP	5.473	5.29	5.181	5.19	5.3	5.46	5.211	5.822	5.467	5.211
OCT	4.61	4.364	4.26	4.3	4.3	4.5	4.3	4.95	4.51	4.28
NOV	6.0	5.67	5.56	5.57	5.65	5.86	5.5	6.58	5.76	5.6
DEC	6.1	5.993	5.844	5.79	5.92	6.34	5.76	6.52	5.97	5.8

TABLE III

MONTHLY MAPE RESULTS FOR A SINGLE DAY INPUT CGPANN MODEL FOR PREDICTING NEXT HOUR

NODES	50	100	150	200	250	300	350	400	450	500	Week
Single day input	1.705	1.868	**1.58**	1.68	1.62	2.27	1.681	1.91	1.81	1.60	3rd
15 days input	2.10	2.65	2.75	1.86	2.17	**1.67**	2.55	1.93	2.82	2.81	3rd
30 days input	3.85	**2.92**	3.87	2.99	3.35	4.26	3.95	3.67	3.28	3.31	29th

TABLE IV

WEEKLY MINIMUM MAPE VALUES FOR 50 UP TO 500 NODES WITH 50 NODES SEPARATION

Network Model	**Single day input**	**15 days input**	**30 days input**
No. Nodes	150	300	100
Week	3rd	3rd	29th
Monday	1.73195144	1.537707725	3.165312782
Tuesday	1.448382014	3.747227909	4.684221208
Wednesday	3.439170294	1.311564903	4.298508772
Thursday	1.213078905	3.835364514	2.941825104
Friday	**1.049062798**	1.10593271	2.116909619
Saturday	1.097208015	1.166413005	1.515822673
Sunday	1.083210534	1.14719582	1.695776838

TABLE V

MAPE RESULTS FOR SINGLE DAY IN THE BEST WEEK OF THE PREDICTED WIND POWER FORECASTED YEAR

Model Reference	**MAPE/NRMSE**
Enhanced Particle Swarm Optimization (EPSO)[9]	7.52%(NRMSE)
Hourly Persistence[7]	10.71%(NRMSE)
Direct ANN[7]	10.45%(NRMSE)
Auto tuning Kalman Filter[15]	5.8%(NRMSE)
Wavelet Support Vector Machine (WSVM)[11]	3.7404%(MAPE)
CGPANN	**1.0497**%(MAPE), **1.49**%(NRMSE)

TABLE VI

COMPARISON OF HOURLY MAPE VALUES OF DIFFERENT MODELS WITH 24HRS INPUT - SINGLE HOUR OUTPUT CGPANN MODEL (AVERAGED DAILY)