A New algorithm to find number of hidden neurons in Radial Basis Function Networks for Wind Speed Prediction in Renewable Energy Systems

K.Gnana Sheela*, S.N.Deepa**

**Ann University, Regional Centre, Coimbatore, India*
(Tel: 918870776053; e-mail: sheelabijins@gmail.com).
*** Ann University, Regional Centre, Coimbatore, India,*
(e-mail:deepapsg@gmail.com)

Abstract: This paper proposes a new algorithm to find hidden neurons in Radial Basis Function Networks for wind speed prediction in renewable energy systems. The proper selection of hidden neuron is important in the design of neural network architectures. The random selection of hidden neuron may cause over fitting and under fitting problem in the network. To find the number of hidden neurons, 101 various criteria are examined based on the error values of mean squared error, mean absolute percentage error and mean absolute error. The minimal error values are considered as the best solution to find hidden neurons in Radial Basis Function network. The proposed new algorithm is tested on real time wind data. The significance of number of hidden neurons is analyzed using these criteria. The criteria are satisfied with convergence theorem. The experimental results show that as compared to alternative methods proposed algorithm performs better in terms of testing errors. The new algorithm is effective, accurate with minimal error than other approaches**.** Simulations infer that with minimum error the proposed algorithm can be used for wind speed prediction in renewable energy systems.

Keywords: Mean Square Error, Neural networks, Radial basis function networks, Renewable energy systems, Wind Speeds**.**

1. INTRODUCTION

A Neural Network (NN) is a computing methodology which resembles a human biological network. It has been an explosion of interest over the last few decades and has been successfully applied in many fields such as prediction, pattern recognition, image processing and classification etc. The main features of Neural network are nonlinearity, ability to generalize, ability to learn, input and output mapping and fault tolerance. One of the major challenges in the design of neural network is the proper selection hidden neuron with minimal errors. The excessive hidden neurons will cause over fitting i.e. the neural network have over estimate the complexity of the target problem (Jinchuan *et al.*, 2008). It greatly degrades generalization capability, to lead with significant deviation in prediction. In this sense, determining the proper number of hidden neurons to prevent over fitting is critical in prediction problem using neural networks. The random selection of neural network model parameters causes either over fitting or under fitting problem. These drawbacks are rectified by using the proposed algorithm. The result with minimum estimated error is determined as optimum for the fixation of hidden neurons. The process of defining proper neural network architecture is designed using the parameters includes number of hidden neurons, algorithm, activation function and stopping criteria. The fixing of the hidden neuron is an important for the designing of Neural Network model.

The wind energy is one of the renewable energy sources with lowest cost of electricity production. Due to the fluctuation of wind, prediction result may change rapidly. Thus increases the importance of accurate wind speed prediction. The proposed algorithm is implemented for the accurate wind speed prediction. The importance of wind speed prediction is to protect security of wind power integration. In this paper determining of hidden neurons is discussed for radial basis function networks (RBFN) as applied for wind speed prediction in renewable energy systems. The quality of prediction made by the network is measured in terms of generalization error.

Several researchers tried and proposed many approach for fixing hidden neuron in feed forward neural network. In (Jin Yan Li *et al.*, 1995) proposed to estimate number of hidden neurons in prediction of time series. In (S.Tamura *et al.*, 1997) is presented another method based on Akaike information criteria. In 1998, Osamu Fujitha (Osamu Fujitha, 1998) developed statistical estimation of number of hidden neuron by feedforward neural network. This approach adds hidden units one by one. In 2003, Zhaozhi Zhange (Zhaozhi Zhange *et al.*, 2003) implemented an algorithm is as set covering algorithm for the selection of hidden neuron. In (Shuxiang *et al.*, 2008) is presented novel approach for fixation of hidden neuron in data mining application.

Radial Basis Function Network is one of the learning neural networks. Hidden neuron can influence error on the neuron to which their output is connected. Thus various criteria were proposed for fixing hidden neuron by researchers during the last couple of decades. Most of researchers have fixed number of hidden neuron based on trial rule (Yuan Lan *et al.*, 2010). In this paper, new algorithm is proposed and is applied for RBFN for wind speed prediction in renewable energy systems. The proposed various criteria tested using convergence theorem which converges infinite into finite sequences. The new algorithm is analyzed with minimal error criteria. The error criteria are mean squared error (MSE), mean absolute percentage error (MAPE) and mean absolute error (MAE). This proposed RBFN model is stable and has fast convergence. The main objective is to minimize error, improve accuracy and stability of radial basis function network.

2. PROBLEM DEFINITION

Wind speed prediction problem have been considered and number of hidden neuron have been analyzed using RBF neural network. To find the number of hidden neurons to solve specific task has been important problem. The problem is analyzed with different criteria as mentioned. With few hidden neurons, the network may not be powerful enough to meet the desired requirement including capacity, error precision and so on. With large number of hidden neurons, the training time and testing time may be long. So find the number of a hidden neuron is important for given problem. An important but difficult problem is to determine the optimal number of parameters. In other words, it needs to measure the discrepancy between neural network and an actual system. In order to tackle this most of the researcher does have mainly focused on improving the performance. There is no way to find hidden neurons in RBFN without try and test during training and computing the generalization error. The hidden output connection weights becomes small as number of hidden neuron become large, and also that the trade off in stability between input hidden and hidden output connection exists. A trade off is formed that if the number of hidden neuron (N_h) become too large, the output neuron becomes unstable, and if the number of hidden neuron becomes too small, the hidden neuron becomes unstable again. The problems of hidden neurons are not fixed. The key of wind speed prediction is rational selection of forecasting model and effective optimization of model performance. The perfect design of neural network architecture is important for the challenge of better accuracy of predictive models. Accurate and reliability of wind speed will be an effective tool for optimizing operating cost and improving reliability of renewable energy systems.

In order to find optimal neural network architecture, different error criteria are used. The optimal number of hidden neurons in network based on the following error criteria.

$$\text{MSE}= \sum_{i=1}^{N}\frac{(Y_i'-Y_i)^2}{N} \tag{1}$$

$$\text{MAPE}=\frac{100}{N}\sum_{i=1}^{N}\left|\frac{(Y_i'-Y_i)}{\overline{Y_i}}\right|, \tag{2}$$

$$\text{MAE}=\frac{1}{N}\sum_{i=1}^{N}(Y_i'-Y_i), \tag{3}$$

where Y_i is predicted output, Y_i' is actual output $\overline{Y_i}$ is average actual output and N is number of samples.

The perfect design of radial basis function network needs proper selection of hidden neuron for the challenge of better accuracy of neural network architecture. There is no generally accepted theory to determine how many hidden neurons are needed to approximate any given function in single hidden layer can be useful in certain architectures. If it has few numbers of hidden neurons, there might have a large training error due to under fitting. If it has more number of hidden neurons, there might have a large training error due to over fitting. An exceeding number of hidden neuron made on the network deepens the local minima problem. These drawbacks are rectified by using proposed new algorithm. The proposed new criteria show the stable performance on training despite of large number of hidden neurons. The objective is to find hidden neurons in designing the RBFN and minimizing the error for wind speed prediction in renewable energy systems.

3. OVERVIEW OF RADIAL BASIS FUNCTION

Radial Basis Function Network is a feed forward neural network (Sivanandam *et al.*, 2008). It has been used for approximating function and recognizing applications. RBFN consist of input, hidden and output layer. The hidden layer is a layer of radial basis function units. Each hidden neurons employs radial basis function to produce localized output with respect to input. The outputs are combination of weighted inputs that are mapped by Gaussian function that is symmetric. The interconnection between input and hidden layer form hypothetical connection and between the hidden and output layers form weighted connections. Each hidden layer unit represents a single radial basis function, with associated center position and width. Each neuron on the hidden layer employs a radial basis function as a nonlinear transfer function to operate on the input data. The most often used radial basis function is a Gaussian function that is characterized by a center and width. RBF functions by measuring the Euclidean distance between input vector and the radial basis function center. The Gaussian RBF may be tuned by adjusting spread. It is less susceptible to problem with non stationary input because of the behaviour of RBF hidden units. The Gaussian function curve which has a peak at zero distance and it decreases as the distance from the centre increases. The advantages of radial basis function networks are more compact, less training time, while eliminating local minima phenomena. The selection of the centers for the gaussian function is important for nonlinear approximation. The weights between the hidden and output layer can then be updated by using the gradient descent rule.

4. PROPOSED WORK TO FIND NUMBER OF HIDDEN NEURONS IN RBFN UNITS

The various heuristics exist in the literature amalgamating knowledge gained from previous experiments on where a near optimal topology might exists .The objective is to devise a criteria that find the number of hidden neurons as a function of input neurons(n) and to develop the model for wind speed prediction in renewable energy systems.

4.1 Proposed Architecture

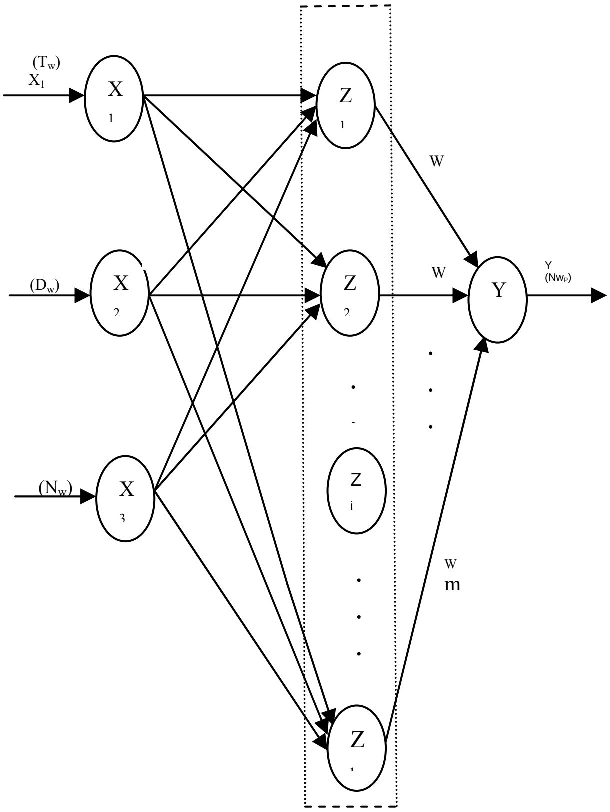

Fig. 1. Architecture of the proposed model.

From Fig.1, it can be observed that, the layers make independent computation on data that it receives and passes the results to another layer and finally, determine the output of the network. Each unit makes its computation based upon the weighted sum of its inputs. The hidden layer has gaussian function and output layer has linear function. The number of hidden neuron used in hidden layer is selected based on new criteria. The proposed model is used for estimation and prediction. The key of the proposed criteria is to select the number of neurons in hidden layer.
From Fig. 1,

Input vector, $X = [T_w, D_w, N_w]$

Output vector, $Y = [N_{wp}]$

Gaussian function is defined as $f(Y_{in}) = e^{-Y_{in}^{\;2}}$ (4)

where Y_{in} is the net input.

Weight vector of hidden to output vector, $W = [W_1, W_2, ... W_n]$

Output of network= $\sum_{k=1}^{h} f(\|X - C_k\|).W_{ik}$ (5)

where k =1 to h, 'h' is the number of hidden neurons. X is the input and $\|X - C_k\|$ is the Euclidean distance between C_k and X. f is the nonlinear function, w_{ik} is the weight between hidden and output layer.

4.2 Proposed Algorithm

Generally, Neural Networks involves the process of training, testing and developing model at end stage for the past years in wind farms. The perfect design of neural network model is important for challenge of better accuracy of model. The higher valued collected data tend to suppress the influence of smaller variable during training. To overcome this problem, the normalization technique is used. The perfect design of neural network model based on the selection criteria is applied on convergence theorem.

The steps involved in the proposed algorithm are as given below.

Step 1: Data collection

The real time data is collected from Suzlon Energy Ltd., India Wind farm for a period from April 2011 to December 2012.The inputs are temperature, wind vane direction from true north and past wind speed in anemometer. The height of wind tower is 65 m. The number of samples were taken to develop a proposed model is 10000.

Table 1. Input Parameters of the proposed model

S.No	Input Parameters	Units	Range of the Parameters
1	Temperature	Degree. C	24-36
2	Wind direction	Degree	1-308
3	Wind speed	m/s	1-16

The parameters are that are considered as input to the neural network model are shown in Table 1. The sample inputs are as shown in Table 2.

Table 2. Sample Inputs

Temperature (Degree Celsius)	Wind Vane direction from true north (degree)	Wind Speed (m/sec)
26.4	285.5	8.9
26.4	286.9	7.6
25.9	285.5	8.6
25.9	284.1	8.9
31.9	302.7	3
25.9	285.5	8.1
25.8	282.7	7.9
33.8	307.4	6.7
25.8	281.2	7.9
25.9	282.7	7.9

Step 2: Data Normalization

The Normalization of data is essential as the variable for different units. Therefore, data are scaled within the range 0 to 1.It scales number of data to improve accuracy of subsequent numeric computation and obtain better output of model. The advantage is preserving exactly all relationships in the data and it does not introduce bias. Calculate normalized value of each input variable using Min Max normalization technique is as follows.

Normalized

$$\text{input}= X_i' = \left(\frac{X_i - X_{\min}}{X_{\max} - X_{\min}}\right)(X'_{\max} - X'_{\min}) + X'_{\min} \quad (6)$$

where X_i, $X_{\min}$, $X_{\max}$ be the actual input data, minimum and maximum input data.

$X'_{\min}$, $X'_{\max}$ be the minimum and maximum target value.

Step 3: Design Network

Set up parameter includes learning rate, epoch and dimensions and so forth. The training can be learned from the past data after normalization. The dimensions like number of input, hidden and output neuron are to be designed. The number of hidden layer is one. The number of hidden neurons is to be fixed on proposed criteria. The net input is weighted sum of inputs. The activation function is applied over the net input to calculate the output of RBFN. The parameters are used for design of RBFN is shown in Table 3.

Table 3. Parameter selection of RBF NN

Spread=3
output=1
No. of hidden layer=1
Inputs=3
Epochs=2000
Spread=3
output=1

Step 4: Selection of proposed criteria

For the proposed model, 101 various criteria were examined to estimate training process and different errors in network. The input neuron is taken into account for all criteria. It is tested on convergence theorem. Convergence is infinite into finite sequence. All chosen criteria satisfied the convergence theorem. Initially, apply the chosen criteria to the RBF network for the development of proposed model. Then train the neural network and compute different errors. The result with the minimum error value is determined as the selection of neurons in hidden layer in the RBFN model.

Step 5: Compute prediction errors using MSE, MAPE, MAE criteria

Calculate error values of each criteria using the equation (1)-(3).Selecting best criteria based on lowest estimation of error values. This is used to determine number of hidden neurons for a given problem.

Step 6: Training and evaluate performance of network

The adequate selection of input for neural network training is highly influential to the success of training. The error minimization process is repeated until acceptable criteria for convergence is reached. Poor convergence reflects inadequate number of hidden neurons in neural network. The collected data is divided into training and testing of network. The training data was used to develop models of wind speed prediction, while testing data was used to validate performance of models from training data. 7000 data is used for training and 3000 data is used for testing data. The training can be learned from past data after normalization. Apply testing data to evaluate the performance of the trained network. The performance of network is evaluated by error values. The selection of proposed criteria is based on the lowest error. The network checks whether performance is accepted. Otherwise goes to next criteria then train and test performance of network. The error values are calculated for each criterion. The result with minimum error is determined as the best for selection of neurons in hidden layer of RBFN. To ensure the effectiveness of proposed approach, actual and predicted wind speed is obtained.

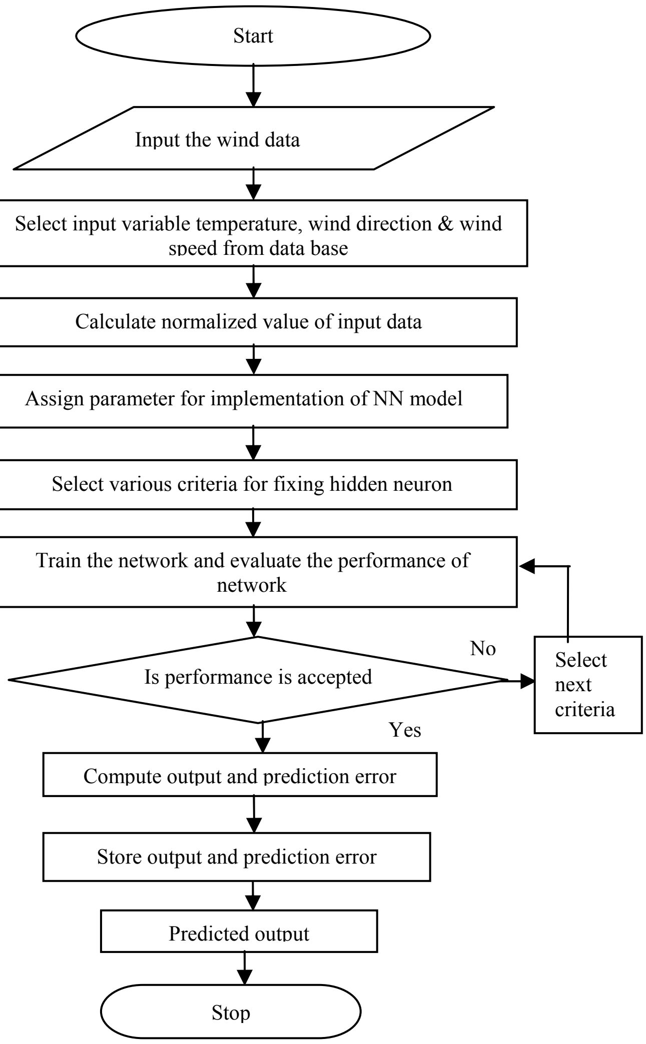

Fig. 2. Flowchart of Proposed approach

Table 4. Statistical Analysis of various criteria in RBF Network.

Considered criteria for fixing number of hidden neuron	no. of hidden neuron	MSE	MAPE	MAE	Considered criteria for fixing number of hidden neuron	no. of hidden neuron	MSE	MAPE	MAE
$2^n+2/n-2$	10	0.0037	0.5448	0.0295	$4n/n-1$	6	0.0231	1.8132	0.098
$5n+5/n-2$	20	1.37E-04	0.1042	0.0056	$6n/n-2$	18	0.0014	0.3354	0.0181
$8n+1/n-2$	25	4.52E-05	0.0638	0.0035	$4^n+2/n-2$	66	3.75E-06	0.0034	1.82E-04
$8n^2/n^2-7$	36	6.58E-06	0.011	5.92E-04	$5n^2+4/n^2-8$	49	4.90E-07	0.005	2.68E-04
$7n+2/n-2$	23	8.09E-05	0.0879	0.0048	$4^n+1/n-2$	65	3.82E-06	0.0034	1.84E-04
$5(n^2+1)+3/n^2-8$	53	3.97E-07	0.0044	2.40E-04	$7(n^2+4)+1/n^2-8$	92	5.43E-08	5.36E-04	2.90E-05
$2n/n+1$	2	0.0586	2.5344	0.2053	$4^n+12/n-2$	76	4.40E-06	8.14E-06	4.40E-05
$4^n+3/n-2$	67	3.73E-06	0.0029	1.55E-04	$4^n-7/n-2$	57	3.33E-07	0.0045	2.40E-04
$6^n+5/n$	74	1.15E-06	8.96E-04	4.84E-05	$8n+5/n-2$	29	1.50E-05	0.0211	0.0011
$7(n^2+4)+3/n^2-8$	94	2.69E-08	4.73E-04	2.56E-05	$4^n-6/n-2$	58	2.37E-07	0.0039	2.13E-04
$7(n^2+5)-3/n^2-8$	95	2.69E-08	4.73E-04	2.56E-05	$3n/n-1$	5	0.0227	1.981	0.1071
$4n^2+1/n^2-8$	37	3.21E-06	0.0095	5.16E-04	$4^n/n+1$	16	0.002	0.3784	0.025
$2^n+3/n-2$	11	0.0037	0.5448	0.0295	$5^n +3 / n$	43	8.53E-06	0.0075	4.06E-04
$4^n/n$	21	1.15E-04	0.102	0.0055	$2^n/n-2$	8	0.0222	1.706	0.0922
$8n+2/n-2$	26	4.60E-05	0.0637	0.0034	$3n+8/n-2$	17	0.0014	0.3322	0.018
$n/n+1$	1	8.4672	33.293	2.6973	$4^n/n-2$	64	1.25E-06	0.004	2.19E-04
$3^n-3/n-2$	24	8.12E-05	0.0845	0.0046	$4^n+15/n-2$	79	4.82E-06	4.99E-06	2.70E-05
$6^n/n+1$	54	3.85E-07	0.0044	2.37E-04	$6(n^2+4)+3/n^2-8$	81	2.70E-08	4.20E-04	2.70E-05
$4^n+4/n-2$	68	3.52E-06	0.0025	1.37E-04	$5n^2/n^2-8$	45	3.23E-06	0.0075	4.05E-04
$7(n^2+4)+2/n^2-8$	93	2.69E-08	4.73E-04	2.56E-05	$3^n +1/n-2$	28	4.48E-05	0.0638	0.0034
$6^n+2/n$	73	1.15E-06	8.96E-04	4.84E-05	$4n^2+3/n^2-8$	39	4.07E-06	0.0088	4.73E-04
$7^n -3/n+1$	85	3.18E-08	3.88E-04	2.10E-05	$4^n /n-1$	32	8.57E-06	0.0171	9.26E-04
$4^n+13/n-2$	77	4.40E-06	8.14E-06	4.40E-05	$2^n/n-1$	4	0.027	2.209	0.1194
$7(n^2+5)-1/n^2-8$	97	2.62E-08	4.71E-04	2.55E-05	$4^n+5/n-2$	69	3.81E-06	0.0019	1.05E-04
$8^n-10/n+2$	100	2.62E-08	4.71E-04	2.55E-05	$2^n+1/n-1$	9	0.0142	1.012	0.082
$5n+7/n-2$	22	1.14E-04	0.1012	0.0055	$5^n/n$	42	8.22E-06	0.0076	4.13E-04
$4n/n-2$	12	0.0028	0.4799	0.0259	$4^n+11 /n-2$	75	4.40E-06	8.14E-06	4.40E-05
$8^n-15 /n+2$	99	2.62E-08	4.71E-04	2.55E-05	$5(n^2+1)+2/n^2-8$	52	6.78E-07	0.0048	2.60E-04
$6^n+2 /n+1$	55	2.88E-07	0.0041	2.20E-04	$8^n-5/n+2$	101	2.62E-08	4.71E-04	2.55E-05
$4^n/n+2$	13	0.0022	0.4596	0.0248	$6(n^2+5)-1/n^2-8$	83	3.07E-08	3.76E-04	2.03E-05
$4n^2+2/n^2-8$	38	4.19E-06	0.0089	4.80E-04	$4^n+6/n-2$	70	3.93E-06	0.0018	9.94E-05
$6^n/n$	72	9.78E-06	9.10E-04	4.92E-05	$4^n-5/n-2$	59	1.04E-07	0.0044	2.36E-04
$7^n/n+1$	86	7.70E-08	5.74E-04	3.10E-05	$2n/n-1$	3	0.0212	1.7685	0.0956
$7(n^2+5)-2/n^2-8$	96	2.69E-08	4.73E-04	2.56E-05	$3^n/n-2$	27	4.69E-05	0.06464	0.0035
$7^n +3/n+1$	87	7.85E-08	5.56E-04	3.00E-05	$4^n+1/n-1$	33	8.56E-06	0.0171	9.23E-04
$7^n -6/n+1$	84	3.18E-08	3.88E-04	2.10E-05	$5n/n-2$	15	0.002	0.5036	0.0272
$5n^2+2/n^2-8$	47	4.08E-06	0.0072	3.90E-04	$4n^2+4/n^2-8$	40	4.24E-06	0.0095	5.12E-04
$3^n +4 /n-2$	31	8.86E-05	0.0178	9.60E-04	$5(n^2+1)/n^2-8$	50	4.96E-07	0.0049	2.62E-04
$4.5n/n-1$	7	0.0231	1.8132	0.098	$4^n-3/n-2$	61	1.26E-07	0.0043	2.31E-04
$7^n +7 /n+1$	88	6.25E-08	5.70E-04	3.10E-05	$8^n+20 /n+2$	98	2.62E-08	4.71E-04	2.55E-05
$5n^2+3/n^2-8$	48	3.19E-06	0.0068	3.68E-04	$6^n +6 /n+1$	56	2.93E-07	0.0044	2.39E-04
$7^n +11 /n+1$	89	7.33E-08	5.60E-04	3.00E-05	$4^n-4/n-2$	60	1.14E-07	0.0043	2.33E-04
$5n+4/n-2$	19	3.91E-04	0.1419	0.0077	$4^n+7/n-2$	71	7.52E-06	9.60E-04	5.19E-05
$7(n^2+4)/n^2-8$	91	5.94E-08	5.52E-04	2.98E-05	$3^n/n-1$	14	0.0018	0.4513	0.0244
$4^n +13 /n-2$	78	6.43E-06	8.50E-06	4.60E-05	$4^n+3 /n-1$	34	8.40E-06	0.0112	6.05E-04
$7^n +15/n+1$	90	8.34E-08	5.34E-04	2.89E-05	$4n^2+5/n^2-8$	41	4.34E-06	0.0094	5.08E-04
$4^n +18/n-1$	**82**	**2.53E-08**	**3.65E-04**	**1.97E-05**	$4^n-2/n-2$	62	1.26E-07	0.0043	2.31E-04
$5n^2+1/n^2-8$	46	2.19E-06	0.0071	3.83E-04	$5(n^2+1)+1/n^2-8$	51	6.78E-07	0.0048	2.60E-04
$3^n +3 /n-2$	30	1.60E-05	0.0187	0.001	$4^n-1/n-2$	63	1.33E-07	0.0043	2.31E-04
$4^n+5 /n-1$	35	8.52E-06	0.0112	6.06E-04	$5(n^2+7)/n^2-8$	80	2.70E-08	4.20E-04	2.70E-05
$5^n +6 / n$	44	1.34E-06	0.0075	4.03E-04					

The considered 101various criteria for fixing the number of hidden neuron with errors are established in Table.4 .The selected criteria for NN model is $4^n +18/n-1$ it has been observed that error values are less compared to other criteria. So this proposed criterion is very effective for wind speed prediction in renewable energy systems.

4.3 Proof for the chosen proposed criteria

Based on the discussion on convergence theorem in the Appendix, the proof for the selection criteria is established henceforth.

Definition1 (convergence): A sequence is said to be convergent if it has a finite limit.

Theorem 1 (Convergence theorem): Convergence theorem states that if the infinite sequence has limit sequence. All sequence has finite value.

A convergent sequence Z_1, Z_2 ....is one that has a limit C written as

$$\lim_{n\to\infty} Z_n = C$$

By definition of limit this means that for every $\varepsilon > 0$, it can find N such that

$$\left|Z_n - C\right| < \varepsilon \text{ for all } n>N$$

A divergent sequence is one that does not converge.

The Lemma 1.1 is an estimate of sequence which is essential for proofing convergence theorem

Lemma1.1

Suppose a sequence $a_n = \dfrac{4^n + 18}{n-1}$, *is converged and* $a_n \geq 0$

It has limit l . *If there exists constant* $\varepsilon > 0$ *such that*

$$\left|a_n - l\right| < \varepsilon$$

then $$\lim_{n\to\infty} a_n = l$$

Proof: The proof based on Lemma1.1

According to theorem, parameter converges to finite value

$$a_n = \frac{4^n + 18}{n-1},$$

$$\lim_{n\to\infty} a_n = \lim_{n\to\alpha} \frac{4^{\infty+}18}{\infty - 1} = finite \ ,$$

Here it is finite limit of sequence as $n \to \infty$

If sequence has limit then it is convergent sequence.

5. DISCUSSION AND RESULTS

Several researchers proposed many approaches to find number of hidden neurons in neural network. The approaches mainly classifies into constructive and pruning approach. The constructive approach, it start with undersized network and then add additional hidden neuron (Vera Kurkova *et al.*, 1997 and Xiaoqin Zeng *et al.*, 2006). The pruning approach, it starts with oversized network and then prunes the less relevant neuron and weights to find the smallest size. The problems of proper number of hidden neuron for a particular problem are to be fixed. The existing methods to determine number of hidden neuron is trial and error rule. This starts with undersized number of hidden neurons (N_h) and adds neurons to number of hidden neuron. The disadvantage is the time consuming and there is no guarantee of fixing the hidden neuron.

The salient points of the proposed approach are discussed here. The result with minimum error is determined as best solution for fixing hidden neurons in neural networks. Simulation results are showing that predicted wind speed is in good agreement with the experimental measured values. Initially real time data are divided into training and testing set. The training set performs in neural network learning and testing set performs to estimating the error. The testing performance stops improving as number of hidden neuron continue to increase; training has begun to fit the noise in the training data, and over fitting occurs. From the results, it is observed that the proposed work given better results than the other approaches. In this paper, proposed criteria are considered for designing a three layer neural networks. It is known that certain approaches produce large size network than necessary and others are expensive. The fixing of number of hidden layer neurons is important in the implementation of neural network. The analysis of wind speed prediction is carried out by the proposed new criteria**.**Table.5 shows that proposed model gives better value for errors in comparison with other existing models.

Table 5. Performance Analysis of various approaches.

S. No.	Various methods	Number of hidden neuron	MSE	MAPE	MAE
1	Jin-Yan Li et al.	$N_h = [(\sqrt{1+8n}) - 1/2]$	8.86E-05	0.0178	9.60E-04
2	S.Tamura	$N_h = N-1$	0.0586	2.5344	0.2053
3	Osamu Fujitha	$N_h = K \log (\|Pc Z\| / C)/\log s$	0.027	2.209	0.1194
4	Zhaozhi Zhang et al.	$N_h = 2^n/n-1$	0.0586	2.5344	0.2053
5	Shuxiang Xu et al.	$N_h = C_f (N/d \log N)$ ½	0.0231	1.8132	0.098
6	Proposed approach	**$N_h = 4^n+18/n-1$**	**2.53E-08**	**3.65E-04**	**1.97E-05**

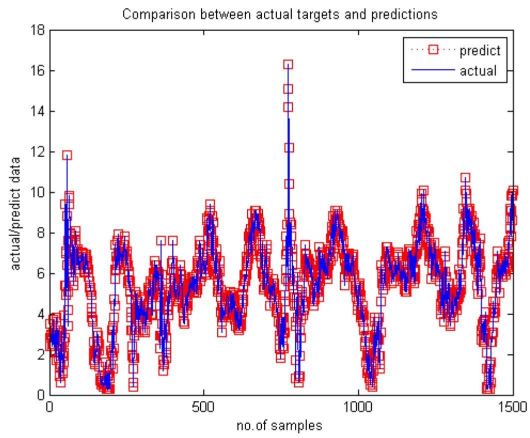

Fig. 3. Actual/ Predicted output waveform

From Fig.3, it's observed the actual and predicted wind speed based on proposed model. The advantages of proposed approach are minimal error, effective and ease of implementation for wind speed prediction in renewable energy systems. The proposed algorithm was simulated and obtained a minimal MSE of 2.5e-08, MAPE of 3.65e-04 and MAE of 1.97e-05.

6. CONCLUSION

A new algorithm has been proposed in this paper to find number of hidden neurons in radial basis function network for wind speed prediction in renewable energy systems. The novelty of new algorithm is that it can find number of hidden neuron of radial basis function network with minimal errors. To obtain optimal number of hidden neuron, different error criteria are used. The improvements are highest accuracy, and minimal error. It is beneficial for wind speed prediction with proper selection of hidden neuron in the network. Simulation results show that proposed algorithm converges to better solution much faster than earlier approaches. Future studies are expected to improved neural network architecture with better performance.

ACKNOWLEDGMENT

The authors are thankful to Suzlon Energy Ltd for providing real time data of wind farm, in order to carry out the research.

REFERENCES

Ashraf U. Haque,Paras Mandal,and Mary E. Kaye . (2012) A new strategy for predicting short term wind speed using soft computing model. *Renewable & Sustainable Energy Reviews*,16,4563-4573.

A.Sfetso. (2000) A comparison of various forecasting techniques applied to mean hourly wind speed time series. *Renewable Energy*, 2123-2125.

B.S.Grewal. (2007) *Higher Engineering Mathematics*, Khanna Publisher, 40th edition, 399-407.

Bumghi Choi, Ju-Hong Lee, and Deok - Hwan Kim. (2008) Solving local minima problem with large number of hidden nodes on two layered feed forward artificial neural networks. *Neurocomputing*, 71, 3640– 3643.

Gaurang Panchal, Amit Ganatra, Y.P Kosta and Devyani Panchal. (2011) Behaviour Analysis of Multilayer Perceptrons with Multiple Hidden Neurons and Hidden Layers. *International Journal of Computer Theory and Engineering,* 3, 332-337.

Jinchuan Ke, Xinzhe Liu. (2008) Empirical Analysis of Optimal Hidden Neurons in Neural Network Modelling for Stock Prediction. *IEEE Pacific Asia Workshop on Computational Intelligence and Industrial Application*, 2, 828-832.

Jin Yan Li, T. W. S Chow and Ying Lin Yu. (1995) The Estimation Theory and Optimization Algorithm for the Number of Hidden Units in the Higher-order Feed forward Neural Network. *Neural Networks*, 3, 1228-1233.

Kanad Keeni, Kenji Nakayama and Hiroshi Shimodaira. (1999) Estimation of Initial Weights and Hidden Units for Fast Learning of Multi layer Neural Networks for Pattern Classification, Department of Information Systems & Quantitative Sciences, Japan.

K. Z. Mao and Guang Bin Huang. (2005) Neuron Selection for RBF Neural Network Classifier Based on Data Structure Preserving Criterion, *IEEE Transactions on Neural Networks,* 16, 6,1531-1540.

Masafumi Hagiwara. (1996) A Simple and effective method for removal of hidden units and weights, *Neuro computing*, 6, 207-218.

Min Han, and JiaYin. (2008) The hidden neurons selection of the wavelet networks using Support vector machines and ridge regression. *Neurocomputing*, 72,471–479

Min Han,Hui Peng,and Liyuan Chen. (2012) A global local optimization approach to parameter estimation of RBF type models.*Information Sciences*,197,144-160.

Md.Monirul Islam, and K. Murase. (2011) A new algorithm to design compact two hidden layer artificial neural network. *Neural Networks*, 14, 1265-1278.

Osamu Fujita. (1998) Statistical estimation of the number of hidden units for feed forward neural networks. *Neural Networks*, 11, 851-859.

Saman Razavi,Bryan A Tolson. (2011) A new formulation for feedforward neural network. *IEEE trans on NN*, 22, 10,1588-1598

Shinichi Tamura and Masahiko Tateishi. (1997) Capabilities of a Four Layered Feed forward Neural Network: Four Layers Versus Three. *IEEE Transactions on Neural Networks*. 8, 251-255.

Shuxiang Xu and Ling Chen.(2008) A Novel Approach for Determining the Optimal Number of Hidden Layer Neurons for FNN's and Its Application in Data Mining, *5th International Conference on Information Technology and Applications*,683-686.

S.N. Sivanandam, S.Sumathi, S.N.Deepa. (2008)*Introduction to Neural Networks using Matlab 6.0*, Tata McGraw Hill, First Edition.

Vera kurkova, Paul C. Kainen and Vladikkreinovich. (1997) Estimates of the Number of Hidden Units and Variation with respect to half spaces. *Neural Networks*, 10, 1061-1068.

Xiaoqin Zeng, and Daniel S. Yeung. (2006) Hidden neuron pruning of multilayer perceptrons using a quantified sensitivity measure. *Neurocomputing,* 825–837.

Yinyin Liu, Janusz A. Starzyk, and Zhen Zhu. (2007) Optimizing number of hidden neurons in neural networks. *IEEE proceedings*, 121-126.

Yuan Lan, Yeng Chai Soh, and Guang Bin Huang. (2010) Constructive hidden nodes selection of extreme learning machine for regression. *Neurocomputing*, 3191-3199.

Yu Jiang, Zhe Song,and Andrew Kusiak. (2013)Very short term wind speed forecasting with Bayesian structural break model.*Renewable Energy*,50.637-647.

Zhaozhi Zhang, Xiaomin Ma, and Yixian Yang. (2003)Bounds on the number of hidden neurons in three-layer binary neural networks. *Neural Networks*, 16, 995–1002.

Zhenhai Guo,Weigang Zhao, Haiyan Lu, and Jianzhou Wang. (2012) Multi-step forecasting for wind speed using a modified EMD based artificial neural network model. *Renewable Energy*,7,241–249.

APPENDIX

Consider various criteria 'n' is number of input neurons. All criteria are satisfied with Convergence theorem. Some illustrations are shown below. The convergence sequence have finite limit. Other sequence is called divergent sequence (Grewal, 2007). The characteristics of Convergence theorem is

- A convergent sequence has a limit
- Every convergent sequence is bounded
- Every bounded point has a limit point
- A necessary condition for the convergence sequence is that it is bounded and has limit
- An oscillatory sequence is not convergent i.e. divergent sequence

A network is stable means no change occurs in the state of the network regardless of the operation. An important property of the neural network model is that it always converges to a stable state. The convergence is important in optimization problem in real time, since it prevents a network from the risk of getting stuck at some local minima. Due to presence of discontinuities in model, the convergence of sequence in finite has been established in convergence theorem. The properties of convergence are used in the design of real time neural optimization solvers.

Discuss convergence of the following sequence

(1) Consider the sequence, $a_n = \frac{2^n + 2}{n-2}$,

Apply convergence theorem,

$$\lim_{n\to\infty} a_n = l$$

where l is the limit of sequence.

$$\lim_{n\to\infty} \frac{2^n + 2}{n-2} = \frac{2^\infty + 2}{\infty - 2} = finite$$

Therefore, the terms of sequence are bounded and the sequence has a limit value. Hence the sequence is convergent.

(2) Consider the sequence, $a_n = \frac{5n+5}{n-2}$,

$$\lim_{n\to\infty} a_n = l$$

where l is the limit of sequence.

Apply convergence theorem,

$$\lim_{n\to\infty} \frac{5n+5}{n-2} = \frac{5n(1+\frac{1}{n})}{n(1-\frac{2}{n})} = 5, \quad \text{finite value}$$

Therefore, the terms of sequence are bounded and the sequence has a limit value. Hence the sequence is convergent.

(3) Consider the sequence, $a_n = \frac{4n}{n-1}$,

Apply convergence theorem,

$$\lim_{n\to\infty} \frac{4n}{n-1} = \lim_{n\to\infty} \frac{4n}{n(1-\frac{1}{n})} = 4, \quad \text{finite value}$$

Therefore, the terms of sequence are bounded and the sequence has a limit value. Hence the sequence is convergent.

(4) Consider the sequence, $a_n = \frac{2n}{n-1}$,

Apply convergence theorem,

$$\lim_{n\to\infty} \frac{2n}{n-1} = \lim_{n\to\infty} \frac{2n}{n(1-\frac{1}{n})} = 2, \quad \text{finite value}$$

Therefore, the terms of sequence are bounded and the sequence has a limit value. Hence the sequence is convergent.