Journal of Engineering Science and Technology
Vol. 17, No. 1 (2022) 0361 - 0378
© School of Engineering, Taylor's University

RADIAL BASIS FUNCTION NEURAL NETWORKS-BASED SHORT TERM ELECTRIC POWER LOAD FORECASTING FOR SUPER HIGH VOLTAGE POWER GRID

MOHAMMED OMAR ALI[1], SADIQ J.ABOU-LOUKH[2], AYAD Q. AL-DUJAILI[3], AHMED ALKHAYYAT[4], AHMED IBRAHEEM ABDULKAREEM[5], IBRAHEEM KASIM IBRAHEEM[6,*], AMJAD J. HUMAIDI[5], ARIF A. AL-QASSAR[5], AHMAD TAHER AZAR[7],

[1]Department of Electrical Power Techniques Engineering, Al-Hussain University College, Karbala, Iraq
[2]Department of Computer Engineering Techniques, Ibn Khaldun University College, Baghdad 10001, Iraq
[3]Department of control and automation techniques engineering, electrical engineering technical college, Middle technical University, Baghdad 10001, Iraq
[4]College of technical engineering, The Islamic University, Najaf, Iraq
[5]Control and Systems Engineering Department, University of Technology, Baghdad 10001, Iraq
[6]Department of Computer Engineering Techniques, Al-Rasheed University College, Baghdad 10001, Iraq
Department of Electrical Engineering, College of Engineering, University of Baghdad, Baghdad 10001, Iraq
[7]Faculty of Computers and Artificial Intelligence, Benha University, Benha 13518,Egypt
College of Computer and Information Sciences, Prince Sultan University, Riyadh 11586, Saudi Arabia
*Corresponding Author: ibraheemki@alrasheedcol.edu.iq

Abstract

Load forecasting plays an essential role both in developed and developing countries for policymakers and related organizations. It helps an electrical utility to make important decisions including decisions on purchasing and generating electrical power, load switching, and infrastructure development. In recent years Artificial Neural Networks (ANNs) have been applied for short-term power load forecasting (STPLF). This work presents a study of STPLF for the Iraqi national grid by means of Radial Basis Function NN(RBFNN) and Multi-Layer Perceptron NN (MLPNN) model. Inputs to the ANN are past loads and the output of the ANN is the load forecast for given days. Historical load data obtained from the Control and Operation Office at the Iraqi ministry of electricity has been split into two main parts, where 50% of the data are used for the training and the other 50% has been devoted to test the trained network. Simulations have been accomplished in MATLAB environment, where the data have been preprocessed and rearranged. Lastly, the simulation results proved that the predicted load values are following closely the actual load.

Keywords: Artificial intelligence, Energy consumption, Load demand, Load forecasting, Load prediction, Neural networks, Radial basis function.

1. Introduction

Power load forecasting (PLF) precisely assumes a vital part for electrical operations in an ambitious climate made by the electric business deregulation. PLF assists an electrical industry by settling on significant resolutions on producing, exchanging, and buying electrical energy, load swapping, and substructure improvement. Besides, PLF is vital for power providers, monetary establishments, and others associated with electrical power productions, transmission, delivery, and marketing [1, 2]. Additionally, PLF is assuming a critical part in diminishing the production cost, also it is important for the power system reliability.

Since in energy systems it is necessary to schedule the power generation of the next day, day-ahead short-term power load forecasting (STPLF) is an important everyday task for “power dispatch”. Its precision influences monetary activity and system reliability significantly. Underestimation of STPLF prompts inadequate reserve assimilation planning and, thus, expands the working expense by utilizing costly peaking units. Furthermore, the overestimation of STPLF prompts the superfluously enormous reserve assimilation, which is correspondingly associated with high working expenses. It is assessed that in the British energy grid, each 1% increment in the prediction error is related to an increment in working expenses of 10 million pounds every year [3].

The STPLF forecaster calculates the estimated load for each hour of the day, the daily peak load, or the daily or weekly energy generation. STLF is important to electrical suppliers because they can use the forecasted load to control the number of generators in operation, to startup new units when the forecasted load is high. The research methods of STPLF can be classified into two classifications: or artificial intelligence methods and statistical parametric methods. In statistical techniques, after training the historical data, conditions can be acquired depicting the connection among the load and its relative factors, while artificial intelligence strategies attempt to mimic the way of thinking and reasoning of the human beings, to obtain information from the past-experience and predict future load [3].

As said previously, the study methods of STPLF can be partitioned into two classes: *statistical* and *artificial intelligence methods*. The statistical methods incorporate state-space [4], general exponential smoothing [5], stochastic time series [6], multiple linear regression [7]. Lately, support vector regression [8, 9], which is an encouraging statistical learning technique, has likewise been implemented to STPLF and has proven great outcomes.

Evolutionary algorithm [10], fuzzy interface [11], artificial neural network (ANN) [12], and Expert systems [13], belong to the computational intelligence category. Moreover, [14] applied Wavelet Neural Network, Particle Swarm Optimization, and Ensemble Empirical Mode Decomposition to short-term load forecasting. The authors of [15] presented an ensemble ANN predictive model to improve STPLF. Different from existing studies, a bagged-boosted ANN has been trained by combining both bagging and boosting techniques. An SVM model, hybrid of mode decomposition, and PSO, was present for predicting short-term Electrical Energy Demand of the market of Australian electricity [16].

In [17], a forecasting model for the next-day Albania electricity demand was presented based on Fuzzy Logic. Authors in [18] suggested a model for forecasting electricity demand in South African based on an adaptive neuro-fuzzy inference system (ANFIS). In South African, the short-term demand for electricity during the peaking period (i.e., from 6:00 to 8:00 pm) has been forecasted based on partially linear additive quantile regression models [19].

The work in [20] forecasted the Spanish demand for electricity using Autoregressive integrated moving average model. A less computational time achievement is obtained in this study with an enhancement in the forecasting of short-term of electricity demand. A dynamic mode decomposition was proposed in [21] to create an STPLF model, the suggested model proved better accuracy and stability as compared with other forecasting methods. Load forecasting using the time series model for the forecast of the load is adopted in [22]. Table 1 summarizes the aforementioned studies in comparison with the main objectives of the current work.

Table 1. Summary of the literature survey.

Reference	Method used
[4]	state-space
[5]	general exponential smoothing
[6]	stochastic time-series
[7]	multiple linear regression
[8, 9]	support vector regression
[10]	Evolutionary algorithm
[11]	fuzzy interface
[12]	artificial neural network
[13]	Expert systems
[14]	Wavelet Neural Network, Particle Swarm Optimization, and Ensemble Empirical Mode Decomposition
[15]	ensemble ANN predictive model
[16]	SVM model
[17]	Fuzzy Logic
[18]	adaptive neuro-fuzzy inference system (ANFIS)
[19]	linear additive quantile regression models
[20]	Autoregressive integrated moving average model
[21]	dynamic mode decomposition
Proposed work	RBFNN

Short-term power load forecasting will be investigated in detail in this paper considering the Iraqi high voltage power grid as a case study. The data for Iraq high voltage power grid that will be used in the developed STPLF algorithms are obtained from Al-Ameen Control Center. This research work involves:

i. Study and analysis of the short-term power load forecasting problem on Iraqi high voltage power grid.

ii. Modeling STPLF using Radial basis function artificial neural networks (RBFNNs).

iii. Developing algorithms for short-term power load forecasting problem using RBFNNs technique which would produce a load forecasting that is as close as possible to the actual one.

iv. Applying the developed STPLF algorithms on the Iraqi High voltage Power grid and comparing the forecasted results with the actual ones.

This paper is structured as follows. Section 2 introduces the requirements of the STPLF systems and the challenges that face them. The structure of the RBFNN, its operation, and parameters training are presented in Section 3. The proposed STPLF using RBFNN is demonstrated in Section 4. Three case studies have been investigated to simulate the proposed RBFNN-based STPLF procedure in Section 5. Finally, the paper is concluded in Section 6.

2. Requirements and Challenges of a good STPLF System

The greater part of the load or demand supervision programs utilized by electrical utilities involve STPLF units. Each utility expects to have a dependable STPLF program for the economic operations of the energy system. The robustness and reliability of the system principally rely upon the precision of the load forecasting. There are other significant necessities for a decent STPLF framework. These necessities consider the accompanying: automatic forecasting report generation, automatic bad data detection, evaluation of the obtained forecast, automatic performance, timely forecast, friendly interface, automatic data access, accuracy, fast speed. On the other hand, several difficulties exist in short-term power load forecasting, these include the precise hypothesis of the input-output relationship, generalization of experts' experience, the forecasting of anomalous days, inaccurate or incomplete forecasted weather data, less generalization ability caused by overfitting. Moreover, Difficulties getting accurate data on consumption behavior due to changes in factors such as pricing and the corresponding demand based on such a price change. The utility may suffer losses if they do not understand and decide on an acceptable margin of error in short term load forecasting. The load forecasting has both commercial and technical implications and if not done properly, it may lead to bad planning and inefficient operation of the electrical power systems.

3. Radial Basis Function Artificial Neural Networks

A new technique is suggested in this study for addressing the STPLF issue utilizing RBFNNs. The additional new training data is treated by the RBFNN without requiring retraining, which is one of the merits of RBFNNs. The output linear layer and a hidden layer of the RBFNNs have the capability of adapting their connection weights efficiently without getting stuck into a local minimum. A less computational time is needed for the training of the RBFNNs since just the second layer's weights have to be adapted based on the error signal. The momentum and adaptive learning rates can be used to accelerate the training of RBFNNs.

3.1. Radial basis function network structure and operation

The RBFNN model includes three layers; the output, hidden, and input layers. The nodes inside each layer are completely connected to the past layer, as shown in Fig. 1. Each node in the input layer is assigned an input variable, while the connection weights between the input and hidden layers are all set to unity. The transfer functions of the hidden layer nodes are the radial basis functions, hence the name, RBFNNs. They are similar to the sigmoid activation functions that are used in the multi-layer perceptron NNs. They are represented by the bell-shaped curve in the hidden nodes shown in Fig. 1. While the output layer gives a linear combination of output weights. RBFNN is a useful tool for the analysis of the relationships between a major sequence and the other comparative sequences in a given set. In comparison with MLPNN, RBFNN has better approximation properties and faster training velocity, in addition to solving the local minima problem [21].

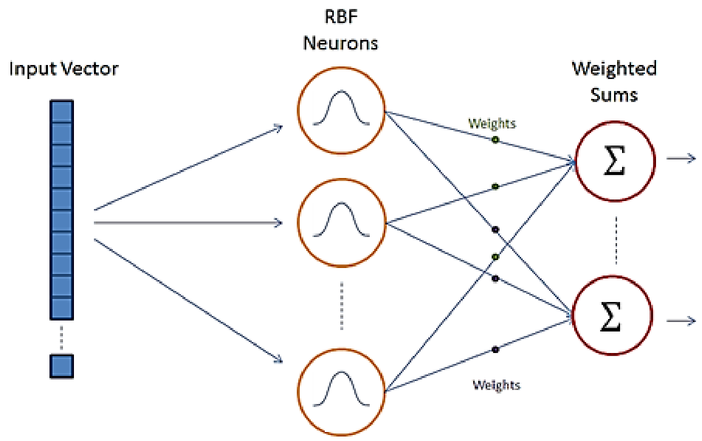

Fig. 1. Schematic representation of an RBFNN.

The two parameters that describe each activation function in the hidden layer of the RBNFF are known as the width (normalization parameter) and the center. Similar to another feedforward NNs, a given RBFNN has to adapt its parameters during the training process, if the hidden layer nodes span the training data input space, that leads to optimum performance, where too overlapping or too sparse functions may cause performance degradation of the load forecasting. During the testing or simulation mode, input vectors are imposed at the input layer of the RBFNN and the output vectors are calculated based on this. The most widely used form of RBF is the Gaussian kernel function given by

$$f_i(x) = exp\left(\frac{-\|x-c_i\|}{2\sigma_i^2}\right) \tag{1}$$

where c_i and σ_i are, respectively, the center and the width of the Gaussian potential function of the ith neuron in the hidden layer, x is the input pattern. The connection between the hidden and output units is the linear weighted sums. The output O_{kp} for the pth incoming input pattern is expressed using of the kth output node,

$$O_{kp} = \sum_{j=1}^{hln} \omega_{kj}\, f_j(x_p) + \omega_{ko} \tag{2}$$

where ω_{kj} is the weight between kth output node and jth RBFNN unit, hln is the number of hidden layer neurons, ω_{ko} is the bias term at the kth output node. In this paper, ω_{ko} is taken as zero.

3.2. Calculations of the RBFNN center (c_i) and the width (σ_i)

The centers (c_i) of radial basis functions for each hidden node define input vectors causing maximal activation of these units. The widths (σ_i) of the radial basis functions of each hidden node determine the radii of the areas of the input space around the centers where activations of these nodes are significant. In the first step, the center vectors (c_i) of the RBFNN in the hidden layer are chosen. This step can be performed in several ways; centers can be randomly sampled from some set of examples, or they can be determined using k-means clustering. Note that this step is unsupervised. So, assign randomly some of the training inputs (x_{1i} , x_{2i} , . . . , x_{ji}) to each hidden node centers [23], i.e.,

$$c_{ji} = x_{ji} \quad (3)$$

where $j = 1, \ldots, hln$, $i = 1, \ldots, iln$, and iln is the number of input patterns. Each training pattern will be assigned to the "nearest cluster". This can be realized by finding the Euclidean distance between the training patterns and cluster centers [24]. The Euclidean distance between x_{pi} and weight vectors (c_{ji}) of each unit is found by,

$$d_j = \sqrt{\sum_{i=1}^{iln} (x_{pi} - c_{ji})^2} \quad j=1,\ldots, hln \quad (4)$$

where x_{pi} is the *ith* variable of the *p*th input pattern, and c_{ji} is the center from neuron *j* to input *i*. When all the training patterns are assigned, the new cluster centers will be determined by averaging the values of each cluster center. This process will be repeated until the cluster center values are not changing. The width of each hidden neuron in the RBFNN is calculated based on the following [22],

$$\sigma_j = \sqrt{\frac{1}{hln} \sum_{k=1}^{hln} \sum_{i=1}^{iln} (c_{ji} - c_{ki})^2} \quad (5)$$

where c_{ji} and c_{ki} are the *i*th entries of the centers of *j*th and *k*th hidden nodes.

3.3. Computation of the weights between hidden neurons and output nodes

Once the centers and the widths of the radial basis functions are obtained, the next stage of the training begins. For this, we can use supervised learning-based techniques such as the least-squares method or the gradient method to update the weights between the hidden layer and the output layer. The training process is presented in the following steps. Firstly, impose an input vector *x* from the training set, then, calculate the outputs of hidden layer neurons $f_i(x)$. Furthermore, Compute the network output vector O_k, compare it with target vector *T*. Adjust each weight in [*w*] in a direction to reduce the difference. For this, the following is used and it is expressed as,

$$\omega_{kj}(n+1) = \omega_{kj}(n) + \beta(t_k - y_k) f_j(x) \quad (6)$$

where ω_{kj} is the weight between the *k*th output layer neuron and the *j*th hidden layer unit, y_k is the *k*th output of the output layer neuron, t_k is the targeted output for *k*th the output layer neuron, and β is an adapting rate parameter. Repeat this process for each input pattern and the entire procedure is repeated until an acceptable error is obtained.

4. Short-Term Power Load Forecasting Using RBFNN

The following steps describe the short-term power load forecasting (STPLF) problem:

(1) *Data collection*. Data collection is a very important process for accurate load forecasting. The data that can be collected includes the daily load records, including 8760 hourly load data in each year.

(2) *Data Pre-processing*. The collected data are normalized to the range [0, 1]. The actual load has been normalized as follows,

$$L_s = \frac{L}{L_{max}} \tag{7}$$

where L_s is the normalized or scaled load, L is the actual load in MW, L_{max} is the maximum load in MW.

(3) *RBFNN construction*. The RBFNN is created using (***Newrb***) function in MATLAB; the network has one output layer, one input layer, and one hidden layer. The functions **radbas**" and "**purelin**", are used for hidden and output layers of the RBFNN respectively, where radbas is a MATLAB function that simulates the Gaussian bell-shaped function in the hidden layer of the RBFNN, while purelin is a MATLAB function that represents a linear combination at the output layer of the RBFNN. The number of neurons in output and input layers was carefully associated with the sample, in accordance with the historical data, however, the number of hidden layer neurons can be calculated from the experimental formula,

$$hln = \sqrt{(iln + oln)} + a \tag{8}$$

where iln is the number of the input layer neurons, *oln* is the number of in the output layer neurons, hln is the number of the hidden layer neurons, a is a constant 1< a <10.

(4) *RBFNN Initialization*. Let $[\omega]$ be the connecting weights between the output and hidden layers. Set the values of the weights to small random values in the range [− 1, 1] [23],

$$[\omega]^o = [\text{random weights}]; [\Delta\omega]^o = [0] \tag{9}$$

where $\Delta\omega$ is the incremental change in the weights.

(5) *RBFNN centers and width calculations*. Use K-mean clustering algorithm to calculate the centers of the hidden layer neurons of the RBFNN. Moreover, calculate the width of each neuron in the hidden layer using formula (5).

(6) *RBFNN Training*. The RBFNN is trained with previous data according to (6). In this work, the training data were from the period (2010) to (2011). The training is stopped until a minimum error has been achieved or the maximum number of iterations has been reached.

(7) *RBFNN simulation*. The network is tested or simulated at the end of the training. This procedure is realized by calling the "**sim**" function in MATLAB.

(8) *RBFNN Post-processing*. De-normalizing the output from the tested network to compare it with the actual data.

(9) *RBFNN Validation*. The last stage is achieved by invoking a performance function to compute and save the performance error statistics, e.g., *MSE* as given below

$$MSE = \frac{1}{N}\sum_{i=1}^{N}[L_a(n) - L_p(n)]^2 \tag{10}$$

where $L_a(n)$ is the value of the actual load, $L_p(n)$ is the predicted load value, and *N* is the number of data points. A flowchart of the entire procedure is shown in Fig. 2.

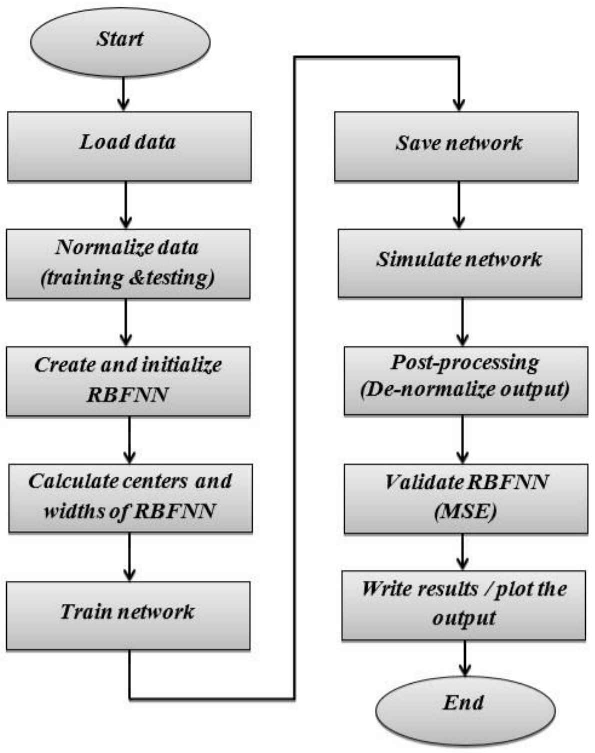

Fig. 2. Flowchart of RBFNN implementation and optimization for STPLF.

5. Numerical Simulations and Results

In this section, a model of RBFNN is introduced as a case study to predict the weekend output load for specific months in 2012 for certain governorates (Baghdad, Basra, and Mosul) in Iraq. The suggested configuration for estimating load demands comprised of four inputs for training and one output. Figure 3 shows the proposed structure of RBFNN models to forecast weekends. For the three cities (Baghdad, Basra, and Mosul) taken as a case study, all the data for the energy network have been gathered from the Iraqi Operation and Control Office for 3 years from 2010 to 2012.

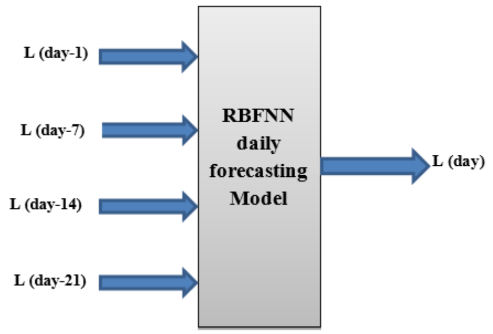

Fig. 3. The suggested RBFNN STPLF model to predict a weekend load.

5.1. Forecasting a weekend of Friday for January in 2012 for Baghdad Governorate.

The data used for the training and testing of the RBFNN are listed in Table 2; they represent the daily power consumption for Baghdad city. The data used in the training belong to the period 7-27 Jan 2011, while the data used in the testing belong to the period 6-26 Jan 2012. The target vector of the training set was the daily power consumption for the day Fri 28, Jan 2011. Table 3 presents the forecasted values for a weekend of Friday in January 2012 for Baghdad city using both Multi-layer perceptron NN (MLPNN) and RBFNN and actual Load values and are plotted against each other in Fig. 4. It can be seen from Table 3 that the forecasted load values produced by the RBFNN are very closer to the actual load values than those predicted by the MLPNN. Moreover, Fig. 4 depicts the forecasted load for Baghdad city using both RBFNN and MLPNN and proves that the load curves predicted by the RBFNN are closely following the actual load curve with a minimum MSE of 1.42×10^{-4} while the MSE of the MLPNN was 0.0035.

Table 2. The testing and training data for predicting a weekend of Friday in January 2012 for Baghdad city.

	Training data (MW)				Training Target	Testing data (MW)				Actual o/p (MW)
Date	**Fri, 07 Jan, 11**	**Fri, 14 Jan, 11**	**Fri, 21 Jan, 11**	**Thu, 27 Jan, 11**	**Fri, 28 Jan, 11**	**Fri, 06 Jan, 12**	**Fri, 13 Jan, 12**	**Fri, 20 Jan, 12**	**Thu, 26 Jan, 12**	**Fri, 27 Jan, 12**
1	2058	2284	2057	2088	1956	2314	2529	2891	2493	2384
2	2060	2289	2068	2090	1987	2256	2466	2820	2431	2324
3	2019	2245	2016	2051	1944	2227	2435	2785	2400	2294
4	1927	2165	1939	1964	1869	2256	2466	2820	2431	2324
5	2048	2292	2090	2084	1992	2314	2529	2891	2493	2384
6	2290	2545	2346	2322	2222	2457	2686	3067	2647	2532
7	2367	2625	2398	2406	2302	2429	2655	3032	2616	2503
8	2435	2673	2405	2538	2343	2544	2780	3173	2740	2622
9	2436	2669	2388	2446	2322	2601	2843	3243	2802	2681
10	2416	2667	2428	2447	2326	2615	2859	3261	2817	2696
11	2435	2665	2442	2455	2317	2615	2859	3261	2817	2696
12	2508	2765	2514	2538	2396	2687	2938	3349	2894	2770
13	2472	2725	2543	2518	2356	2673	2922	3331	2879	2755
14	2431	2677	2490	2480	2361	2673	2922	3331	2879	2755
15	2546	2672	2482	2478	2346	2673	2922	3331	2879	2755
16	2614	2709	2535	2519	2409	2630	2875	3278	2832	2711
17	2702	2789	2571	2585	2511	2716	2969	3384	2925	2800
18	2834	2952	2789	2743	2712	2774	3032	3455	2987	2860
19	2624	2747	2567	2577	2511	2716	2969	3384	2925	2800
20	2550	2671	2489	2458	2444	2687	2938	3349	2894	2770
21	2519	2634	2429	2409	2393	2630	2875	3278	2832	2711
22	2277	2511	2291	2289	2274	2572	2812	3208	2771	2651
23	2199	2429	2231	2208	2189	2508	2749	3137	2709	2592
24	2158	2388	2241	2168	2155	2472	2681	3043	2643	2554

Table 3. Predicted and Actual load using MLPNN and RBFNN models for Friday weekend in January 2012 for Baghdad Governorate.

No. of hours	Forecasted output using MLPNN(MW)	Forecasted output using RBFNN(MW)	Actual output (MW)
1	2240.5	2420.1	2384
2	2215.4	2359.5	2324
3	2199	2329.3	2294
4	2215.4	2359.5	2324
5	2240.5	2420.1	2384
6	2310.4	2572.2	2532
7	2291.8	2541.8	2503
8	2389.6	2663.6	2622
9	2458	2724.3	2681
10	2475.5	2739.5	2696
11	2475.5	2739.5	2696
12	2552.9	2815.2	2770
13	2539.4	2800.1	2755
14	2539.4	2800.1	2755
15	2539.4	2800.1	2755
16	2492.6	2754.7	2711
17	2576.3	2845.3	2800
18	2609.3	2905.4	2860
19	2576.3	2845.3	2800
20	2552.9	2815.2	2770
21	2492.6	2754.7	2711
22	2423	2694	2651
23	2358.4	2631.1	2592
24	2312.5	2567.9	2554
MSE	**0.0035**	**1.42E-04**	

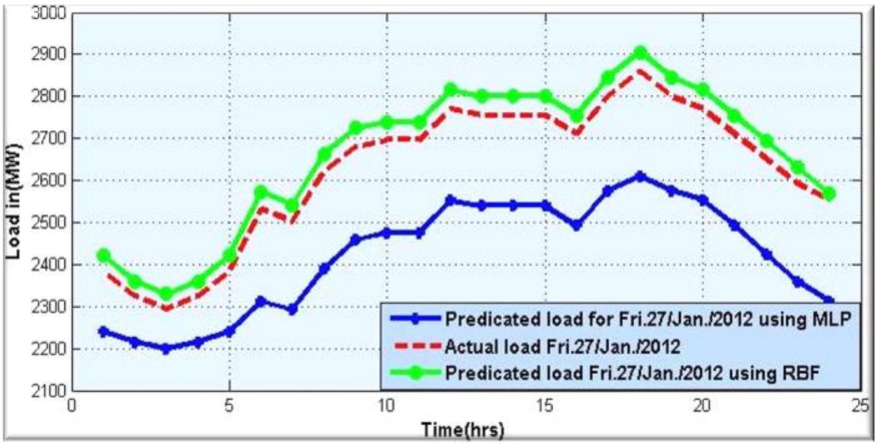

Fig. 4. Forecasted and Actual load using MLPNN and RBFNN of Friday weekend in January 2012 for Baghdad Governorate.

5.2. Forecasting a weekend of Friday for January in 2012 for Basra Governorate

The data used for the training and testing of the RBFNN are listed in Table 4, they represent the daily power consumption for Basra city. The data used in the training belong to the period 7-27 Jan 2011, while the data used in the testing belong to the period 6-26 Jan 2012. The target vector of the training set was the daily power consumption for the day Fri 28, Jan 2011.

Table 5 presents the forecasted values for a weekend of Friday on 27 Jan 2012 for Basra city using both multi-layer perceptron NN (MLPNN) and RBFNN and actual Load values and are plotted against each other in Fig. 5. It can be seen from Table 5 that the forecasted load values produced by the RBFNN are very closer to the actual load values than those predicted by the MLPNN.

Moreover, Fig. 5 depicts the forecasted load for Baghdad city using both RBFNN and MLPNN and proves that the load curve predicted by the RBFNN is closely following the actual load curve with a minimum MSE of 1.05×10^{-4} while the MSE of the MLPNN is 0.0021.

Table 4. The testing and training data for predicting Friday weekend in January 2012 for Basra Governorate.

	Training data (MW)				Training Target	Testing data (MW)				Actual o/p (MW)
Date	**Fri, 07 Jan, 11**	**Fri, 14 Jan, 11**	**Fri, 21 Jan, 11**	**Thu, 27 Jan, 11**	**Fri, 28 Jan, 11**	**Fri, 06 Jan, 12**	**Fri, 13 Jan, 12**	**Fri, 20 Jan, 12**	**Thu, 26 Jan, 12**	**Fri, 27 Jan, 12**
1	1006	1029	1014	991	965	786	859	964	844	813
2	1006	1029	1014	991	965	767	838	940	823	793
3	994	1016	1001	979	953	757	828	928	813	783
4	970	991	977	955	930	767	838	940	823	793
5	1006	1029	1014	991	965	786	859	964	844	813
6	1079	1103	1087	1063	1034	834	912	1022	895	863
7	1103	1128	1111	1087	1058	824	901	1011	885	853
8	1115	1140	1123	1098	1069	863	943	1058	926	893
9	1115	1140	1123	1098	1069	882	964	1081	947	912
10	1115	1140	1123	1098	1069	886	969	1087	952	917
11	1115	1140	1123	1098	1069	886	969	1087	952	917
12	1139	1165	1148	1122	1093	910	996	1116	978	942
13	1127	1152	1136	1110	1081	906	990	1110	973	937
14	1115	1140	1123	1098	1069	906	990	1110	973	937
15	1115	1140	1123	1098	1069	906	990	1110	973	937
16	1127	1152	1136	1110	1081	891	975	1093	957	922
17	1151	1177	1160	1134	1104	920	1006	1128	988	952
18	1212	1239	1221	1194	1162	939	1027	1152	1009	972
19	1151	1177	1160	1134	1104	920	1006	1128	988	952
20	1127	1152	1136	1110	1081	910	996	1116	978	942
21	1115	1140	1123	1098	1069	891	975	1093	957	922
22	1079	1103	1087	1063	1034	872	954	1069	937	902
23	1055	1078	1062	1039	1011	853	933	1046	916	883
24	1042	1066	1050	1027	1000	839	917	1028	901	868

Table 5. Forecasted and actual load using MLPNN and RBFNN models for Friday weekend in January 2012 for Basra Governorate.

No. of hours	forecasted output for MLPNN (MW)	forecasted output for RBFNN (MW)	Actual output (MW)
1	873.5	824.7574	813
2	857.3	804.6587	793
3	849.5	794.6104	783
4	857.3	804.6587	793
5	873.5	824.7574	813
6	917.4	875.0155	863
7	908.2	864.9628	853
8	946.3	905.1769	893
9	966.7	925.2866	912
10	971.9	930.3143	917
11	971.9	930.3143	917
12	998.7	955.4539	942
13	993.3	950.4258	937
14	993.3	950.4258	937
15	993.3	950.4258	937
16	977.2	935.3421	922
17	1009.8	965.5103	952
18	1032.4	985.6237	972
19	1009.8	965.5103	952
20	998.7	955.4539	942
21	977.2	935.3421	922
22	956.4	915.2315	902
23	936.5	895.1226	883
24	922.1	880.0421	868
MSE	**0.002135479**	**1.05E-04**	

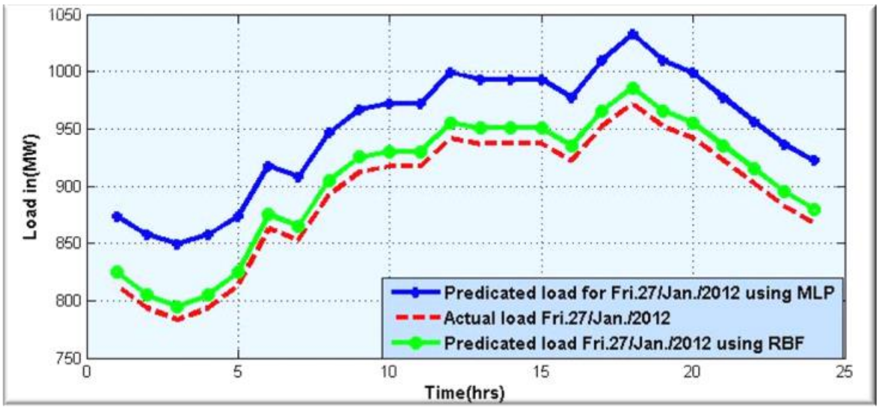

Fig. 5. forecasted and actual load using MLPNN and RBFNN of Friday weekend in January 2012 for Basra Governorate.

5.3. Forecasting a weekend of Friday for January in 2012 for Mosul Governorate.

The data used for the training and testing of the RBFNN are listed in Table 6, they represent the daily power consumption for Mosul city. The data used in the training belong to the period 7-27 Jan 2011, while the data used in the testing belong to the period 6-26 Jan 2012. The target vector of the training set was the daily power consumption for the day Fri 28, Jan 2011.

Table 7 presents the forecasted values for a weekend of Friday on 27 Jan 2012 for Mosul city using both multi-layer perceptron NN (MLPNN) and RBFNN and actual Load values and are plotted against each other in Fig. 6. It can be seen from Table 7 that the forecasted load values produced by the RBFNN are very closer to the actual load values than those predicted by the MLPNN.

Moreover, Fig. 6 depicts the forecasted load for Mosul city using both RBFNN and MLPNN and proves that the load curve predicted by the RBFNN is closely following the actual load curve with a minimum MSE of 1.19×10^{-4} while the MSE of the MLPNN is 8.55×10^{-4}.

Table 6. The training and testing data for predicting Friday weekend in January 2012 for Mosul Governorate.

	Input data for training (MW)				Target for training	Input data for testing (MW)				Actual o/p (MW)
Date	**Fri, 07 Jan, 11**	**Fri, 14 Jan, 11**	**Fri, 21 Jan, 11**	**Thu, 27 Jan, 11**	**Fri, 28 Jan, 11**	**Fri, 06 Jan, 12**	**Fri, 13 Jan, 12**	**Fri, 20 Jan, 12**	**Thu, 26 Jan, 12**	**Fri, 27 Jan, 12**
1	915	936	922	902	878	846	925	1037	909	876
2	915	936	922	902	878	825	903	1012	886	854
3	904	925	911	891	867	815	891	999	875	843
4	882	902	889	869	846	825	903	1012	886	854
5	915	936	922	902	878	846	925	1037	909	876
6	981	1003	989	967	941	898	982	1101	964	929
7	1004	1026	1011	989	962	887	970	1088	953	918
8	1015	1037	1022	999	973	929	1015	1139	997	961
9	1015	1037	1022	999	973	949	1038	1164	1019	982
10	1015	1037	1022	999	973	954	1044	1170	1025	988
11	1015	1037	1022	999	973	954	1044	1170	1025	988
12	1037	1060	1044	1021	994	980	1072	1202	1053	1014
13	1026	1049	1033	1010	983	975	1066	1196	1047	1009
14	1015	1037	1022	999	973	975	1066	1196	1047	1009
15	1015	1037	1022	999	973	975	1066	1196	1047	1009
16	1026	1049	1033	1010	983	960	1049	1177	1031	993
17	1048	1071	1055	1032	1005	991	1083	1214	1064	1025
18	1103	1127	1111	1086	1058	1011	1106	1240	1086	1046
19	1048	1071	1055	1032	1005	991	1083	1214	1064	1025
20	1026	1049	1033	1010	983	980	1072	1202	1053	1014
21	1015	1037	1022	999	973	960	1049	1177	1031	993
22	981	1003	989	967	941	939	1027	1151	1008	972
23	959	981	967	945	920	918	1004	1126	986	950
24	948	970	955	934	909	903	987	1107	970	934

Table 7. Forecasted and actual load using MLPNN and RBFNN models for Friday weekend in January 2012 for Mosul Governorate.

No. of hours	forecasted output for MLPNN (MW)	forecasted output for RBFNN (MW)	Actual output (MW)
1	924	887.8	876
2	906.9	866.2	854
3	898.3	855.5	843
4	906.9	866.2	854
5	924	887.8	876
6	967.6	941.9	929
7	958.6	931.1	918
8	995.6	974.4	961
9	1014.9	996.1	982
10	1019.6	1001.5	988
11	1019.6	1001.5	988
12	1040.8	1028.5	1014
13	1036.9	1023.1	1009
14	1036.9	1023.1	1009
15	1036.9	1023.1	1009
16	1024.2	1006.9	993
17	1048.1	1039.3	1025
18	1060.7	1060.8	1046
19	1048.1	1039.3	1025
20	1040.8	1028.5	1014
21	1024.2	1006.9	993
22	1005.3	985.2	972
23	986	963.6	950
24	972.1	947.3	934
MSE	**8.55E-04**	**1.19E-04**	

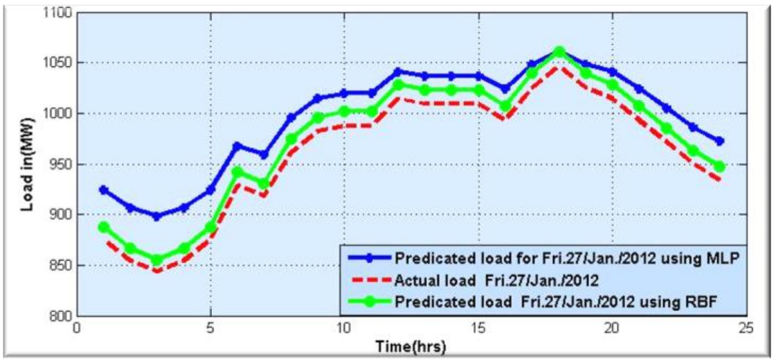

Fig. 6. forecasted and Actual load using MLPNN and RBFNN of Friday weekend in January 2012 for Mosul Governorate.

5.4. Discussion

This study presents a simulation of the STPLF problem for the Iraqi national grid using two neural networks, i.e., RBFNN and MLPNN. Training stops when any of these conditions occur:

- The maximum number of epochs (repetitions) is reached.
- The maximum amount of time has been exceeded.

- Performance has been minimized to the goal.

The momentum term and learning rate have a very substantial effect on learning convergence. Faster convergence can be obtained with a large value of learning rate, but it results in network oscillations. While a slower convergence is attained with a small value of learning rate with a more stable process. Similarly, for higher values of momentum coefficient, the connection weights are rapidly updated in the correct direction and improve the convergence.

The selection of the number of hidden layer neurons is very important and troublesome. If the number of the hidden layer neuron is fewer, the neural networks cannot receive all necessary information of the modeling system and have less tolerance on faults, so that it gives a wrong output. On the contrary, the neural networks may cause a phenomenon called "overfitting". The reason that RBFNN outperforms the conventional MLPNN is due to the fact that RBFNN has the advantage of handling the augmented new training data without requiring retraining. The linear output layer and a hidden layer of RBFNNs have the capability of adapting the connection weights efficiently without getting stuck in the local minimum.

The detailed network parameters are presented along with the training and testing patterns. For each case under study, the actual and forecasted load demand values are presented which shows that the forecasted values are following the actual load values. Finally, has many applications including energy purchasing, generation and control, load switching, and infrastructure development [25-38].

6. Conclusion

This study offers the application of RBFNN for STPLF for the Iraqi national grid for three different cities (Baghdad, Basra, and Mosul). In this work, the modeling and design of NN architecture for STPLF purposes have been examined and successfully implemented.

The obtained results demonstrated the effectuality of the proposed RBFNN technique. The RBFNN was subjected to several training sessions with a varying number of training cycles. After each experimentation, the RBFNN was verified for its capability to correctly forecast the tested data.

From the implementations of the proposed RBFNN model, it can be concluded that among other methods of STPLF, the RBFNN has been established as an encouraging tool in energy system load forecasting problem solutions.

The forecasting reliabilities of the RBFNN were evaluated by computing the mean square error (MSE) between the exact and predicted values.

Moreover, the RBFNN model with the developed structure can perform accurate load forecasting with minimum MSE, and this neural network could be an important tool for short-term power load forecasting.

Finally, the results strongly indicated that the RBFNN model performs better than the MLPNN model. The RBFNN model can also compute reliability measures which is an added advantage of the RBFNN model. Therefore, RBFNN is more suitable for the applications in the design of load forecasting instruments.

Future work includes adding weather condition and changes in temperature, humidity, and other factors that influence consumption.

Nomenclatures

c_i	Center of the Gaussian potential function
d_j	Euclidean distance between x_{pi} and weight vectors (c_{ji})
hln	Number of hidden layer neurons
iln	Number of input patterns
O_{kp}	*k*th output node
t_k	Targeted output for *k*th the output layer neuron
y_k	*k*th output of the output layer neuron

Greek Symbols

β	Adapting rate parameter
σ_i	Width of the Gaussian potential function
ω_{kj}	Weight between the *k*th output layer neuron and the *j*th hidden layer unit
ω_{kj}	Weight between *k*th output node and *j*th RBFNN unit
ω_{ko}	Bias term at the *k*th output node

Abbreviations

ANNs	Artificial Neural Networks
MLPNN	Multi-Layer Perceptron NN
PLF	Power load forecasting
RBFNN	Radial Basis Function NN
STPLF	short-term power load forecasting

References

1. Lee, K.Y.; Cha; Y.T.; and Park, J.H. (1992). Short-term load forecasting using an artificial neural network. *IEEE Transactions on Power Systems*, 7(1), 124-132.
2. Chow, J.H. (2005). *Applied mathematics for restructured electric power systems: Optimization, control, and computational intelligence*. New York: Springer Verlag, Ch. 12.
3. Gross, G.; and Galiana, F.D. (1987). Short-term load forecasting. *Proceedings of the IEEE*, 75(12),1558-1573.
4. Nagbe, K.; Cugliari, J.; Jacques, J. (2018). Short-term electricity demand forecasting using a functional state space model. *Energies*, 11(5):1120.
5. Christiaanse, W.R. (1971). Short term load forecasting using general exponential smoothing. *IEEE Transactions on Power Apparatus and Systems*, PAS-90(2), 900-911.
6. Agyeman, K.A.; Kim, G.; Jo, H.; Park, S.; and Han, S. (2020). An ensemble stochastic forecasting framework for variable distributed demand loads. *Energies*, 13(10): 2658.
7. Dhaval, B.; and Deshpand, A. (2020). Short-term load forecasting with using multiple linear regression. I*nternational Journal of Electrical and Computer Engineering* (*IJECE*), 10(4), 3911-3917.
8. Chen, Y.; Xu, P.; Chu, Y.; Li, W.; Wu, Y.; Ni, L.; Bao, Y.; and Wang, K. (2017). Short-term electrical load forecasting using the support vector regression (SVR) model to calculate the demand response baseline for office buildings. *Applied Energy*, 195, 659-670.

9. Zhang, Z.; Hong, and W.-C. (2019). Electric load forecasting by complete ensemble empirical mode decomposition adaptive noise and support vector regression with quantum-based dragonfly algorithm. *Nonlinear Dynamics* 98(2), 1107-1136.
10. Hu, Y.; Qu, B.; Wang, J.; Liang, J.; Wang, Y.; Yu, K.; Li, Y.; and Qiao, K. (2021). Short-term load forecasting using multimodal evolutionary algorithm and random vector functional link network based ensemble learning. *Applied Energy*, 285: 116415.
11. Anoop, K.J.; and Kanchana, K. (2017). Short term load forecasting using fuzzy logic control. 2017 *IEEE International Conference on Power*, *Control*, *Signals and Instrumentation Engineering* (*ICPCSI*), Chennai, India, 2983-2986.
12. Ibraheem, I.K.; and Ali, M.O. (2014). Short term electric load forecasting based on artificial neural networks for weekends of Baghdad power grid. *International Journal of Computer Applications*, 89(3), 30-37.
13. Geysen, D.; Somer, O.D.; Johansson, C.; Brage, J.; and Vanhoudt, D. (2018). Operational thermal load forecasting in district heating networks using machine learning and expert advice. *Energy and Buildings*,162, 144-153.
14. López, C.; Zhong, W.; Zheng, M. (2017). Short-term electric load forecasting based on wavelet neural network, particle swarm optimization and ensemble empirical mode decomposition. *Energy Procedia*, 105, 3677-3682.
15. Khwaja, A.S.; Anpalagan, A.; Naeem, M.; and Venkatesh, B. (2020). Joint bagged-boosted artificial neural networks: using ensemble machine learning to improve short-term electricity load forecasting. *Electric Power Systems Research*, 179: 106080.
16. Hong, W.-C.; and Fan, G.-F. (2019). Hybrid empirical mode decomposition with support vector regression model for short term load forecasting. *Energies*, 12(6): 1093.
17. Konica, J.A.; and Hanelli, L. (2016). Forecasting next-day the electricity demand based on fuzzy logic method case for Albania. *Journal of Multidisciplinary Engineering Science and Technology* (*JMEST*), 3(12), 6172-6180.
18. Motepe, S.; Hassan, A.N.; and Stopforth, R. (2018). South African distribution networks load forecasting using ANFIS. In Proceedings of 2018 IEEE International Conference on Power Electronic Drives Energy System (PEDES), Chennai, India, 1-6.
19. Lebotsa, M.E.; Sigauke, C.; Bere, A.; Fildes, R.; and Boylan, J.E. (2018). Short term electricity demand forecasting using partially linear additive quantile regression with an application to the unit commitment problem. *Applied Energy*, 222, 104-118.
20. Caro, E.; Juan, J.; and Cara, J. (2020). Periodically correlated models for short-term electricity load forecasting. *Applied Mathematics and Computation*, 364: 124642.
21. Chang, W.-Y. (2013). Estimation of the state of charge for a LFP battery using a hybrid method that combines a RBF neural network, an OLS algorithm and AGA. *International Journal of Electrical Power and Energy Systems*, 53, 603-611.
22. Lekshmi, M.; Subramanya, K.N.A. (2019). Short-term load forecasting of 400kv grid substation using r-tool and study of influence of ambient temperature on the forecasted load. *Proceedings of* 2019 *Second International Conference on Advanced Computational and Communication Paradigms* (*ICACCP*), 1-5.
23. Khwaja, A.S.; Zhang, X.; Anpalagan, A.; and Venkatesh, B. (2017). Boosted neural networks for improved short-term electric load forecasting. *Electric Power Systems Research*,143, 431-437.

24. Salkuti, S.R. (2018). Short-term electrical load forecasting using radial basis function neural networks considering weather factors. *Electrical Engineering*, 100, 1985-1995.
25. Ibraheem I.K. (2012). On the frequency domain solution of the speed governor design of non-minimum phase hydro power plant. *The Mediterranean Journal of Measurement and Control,* 8(3), 422-429.
26. Mohammed, I.A.; Maher, R.A., Ibraheem, I.K. (2011). Robust controller design for load frequency control in power systems using state-space approach. *Journal of Engineering*, 17(2), 265-278.
27. Firas, M.; Tuaimah, I,K.; Ibraheem, I.K. (2010). Robust H_∞ controller design for hydro turbines governor. *Proceedings of the* 2*nd* *Regional Conference for Engineering Sciences*, College of Engineering, Al-Nahrain University, Baghdad, Iraq, 1-9.
28. Maher, R.A.; Mohammed, I.A.; Ibraheem, I.K. (2014). Polynomial based H∞ robust governor for load frequency control in steam turbine power systems. *International Journal of Electrical Power & Energy Systems*, 57, 311-317.
29. Al-Qassar, A.A.; Abdulkareem, A.I.; Humaidi, A.J.; Ibraheem, I.K.; Azar, A.T.; Hameed, A.H.(2021). Grey-wolf optimization better enhances the dynamic performance of roll motion for tail-sitter VTOL aircraft guided and controlled by STSMC. *Journal of Engineering Science and Technology* (*JESTEC*), 16(3), 1932-1950.
30. Al-Qassar, A.A.; Al-Dujaili, A.Q.; Hasan, A.F.; Humaidi, A.J.; Ibraheem, I.K.; Azar, A.T. (2021). Stabilization of single-axis propeller-powered system for aircraft applications based on optimal adaptive control design. *Journal of Engineering Science and Technology* (*JESTEC*), 16(3), 1851-1869.
31. Ajel, A.R.; Humaidi, A.J.; Ibraheem, I.K.; Azar, A.T.(2021). Robust Model Reference Adaptive Control for Tail-Sitter VTOL Aircraft, *Actuators*, 10(7), 162.
32. Ibraheem, I.K. (2011). A digital-based optimal AVR design of synchronous generator exciter using LQR technique. Al-Khwarizmi Engineering Journal, 7(1), 82-94.
33. Ibraheem, I.K.(2014). Damping low frequency oscillations in power system using quadratic gaussian technique based control system design. *International Journal of Computer Applications*, 92(11), 18-23.
34. Maher, R.A., Mohammed, I.S., Ibraheem, I.K.(2012). State-space based H∞ robust controller design for boiler-turbine system. *Arabian Journal for Science and Engineering*, 37(6), 1767-1776.
35. Ibraheem, I.K.(2009). Robust governor design for hydro machines using H∞ loop-shaping techniques. *Proceedings of the* 6*th* *Engineering Conference*, Baghdad, Iraq, 4, 403-414.
36. Hadi, N.H., Ibraheem, I.K.(2021). Speed control of an SPMSM using a tracking differentiator-PID controller scheme with a genetic algorithm. *International Journal of Electrical and Computer Engineering*, 11(2), 1728-1741.
37. Al-Obaidi, A.S.M.; Al-Qassar, A.A; Nasser A.R.; Alkhayyat A.; Humaidi, A.J.; and Ibraheem, I.K.(2021). Embedded design and implementation of mobile robot for surveillance applications. *Indonesian Journal of Science and Technology* (*IJOST*), 6(2), 427-440.
38. Al-Qassar, A.A.; Al-Obaidi, A.S.M.; Hasan, A.F.; Humaidi, A.J.; Nasser, A.R; Alkhayyat, A.; Ibraheem, I.K. (2021). Finite-time control of wing-rock motion for delta wing aircraft based on whale-optimization algorithm. *Indonesian Journal of Science and Technology* (*IJOST*), 6(3), 441-456.