Performance comparisons of RBFN and PRNN in river flows forecasting at multiple sections

Joseph, Parthajit Roy
NIT Silchar, Assam, India

Abstract-The present study demonstrated the maneuver of Radial Basis Function Network (RBFN) and Partial Recurrent Neural Network (PRNN) in river flows forecasting at multiple sections. The current research focused on developing significant model relatively to predict short range hourly flows in real-time scenarios. The model were applied to Barak River network, in Assam and was trained and validated with large numbers of monsoon seasonal data, collected from central water commission, Shillong in Meghalaya. The model performance was evaluated using statistical measures namely root mean square error (RMSE), coefficient of efficiency (COE), peak flow criteria (PFC) and coefficient of correlation (COR). The result shows that PRNN has an insignificant competence to RBFN and is exceedingly reliable in forecasting real time concurrent flood flows simultaneously with improved performance measure.

Keywords- river flows; modeling; network training; forecasting

I. INTRODUCTION

Forecasting river flows simultaneously at multiple sections in a river system is an important issue in water resources engineering which is essential for mitigating damages caused by flood as well as management of water resources. The literature is currently rich with different hydrologic and hydraulic river flow models which particularly focus on predicting river flows at single sections only. Timely and accurate forecasting is an important issue for flow forecasting models. Thus it is important that besides considering the modeling strategy performance outcomes of the models also need to be analyzed and compared. This is particularly more important for short range flow forecasting to provide sufficient time for precautionary measures to be taken for the safety of public life and property.

There have been many recent research works proving the ability and potentials of RBFN and RNNs modeling approaches in modeling rainfall-runoff and many time series forecasting. However the applications of PRNN were not elucidated generously for river flow forecasting. PRNN starts with a fully recurrent network and add a feed forward connection that bypasses the recurrence effectively treating the recurrent part as state memory [13]. E.Derya Ubeyli and M. Ubeyli [9], briefly describe the working of PRNN network and its application for forecasting in different cases of engineering problems. D. Nagesh Kumar, K. Srinivasa Raju, T. Sathish [7] forecasted monthly stream flows of the river Hemavathi (India) using two ANN techniques viz., Feed Forward Neural Networks and Recurrent Neural Networks (i.e. the context RNNs also called PRNN). They suggested that RNNs were found to perform better in forecasting monthly river flows. Parthajit Roy, Manabendra Saharia & P. Choudhury[17] investigates the applicability of PRNN models with and without memory attached to the input layer for predicting concurrent flood flows at multiple locations in a river system.

Many of the literature suggested that the RBFN considerably outperformed the traditional gradient-descent ANNs [3],[16]. A. W.Jayawardena, D. A. K. Fernando & M. C. Zhour [1] made a contrast between Multilayer Perceptron (MLP) and Radial Basis Function (RBFN) networks, in flood forecasting. They described the basic difference between the two methods and suggested that the parameters of the former network are nonlinear and those of the latter are linear. The optimum model parameters are therefore guaranteed in the latter, whereas it is not so in case of former approach. The RBFNs based models give predictions comparable in accuracy to those from the MLP based models [2]. It is also observed that the RBF approach requires less time for model development as no extra recurrence activity is needed to reach optimal model parameters. RBFN can be used effectively for developing rainfall–runoff models that provide high accuracy of flood flow prediction [10]. RBFNs are the universal approaches which provide techniques in approximating the nonlinear functional mappings for multidimensional spaces. In the sense of neural learning RBFNs are faster than other artificial neural network models [20].

ANN partakes high functioning with fast computation and a significant memory in solving problems concerning particularly nonlinear and complex effective variables. The present study aims to make a comparison between two types of ANN techniques: RBFNs and PRNNs suitable for forecasting concurrent river flows simultaneously at multiple sections.
II. DESCRIPTIONS OF STUDY AREA AND DATA

The study area alludes to Barak River system which is the second largest river system next to river Brahmaputra in Assam. Barak basin covers some parts of India, Bangladesh and Myanmar. It spreads in Indian states which include Meghalaya, Manipur, Mizoram, Assam, etc. with an area of 41,723 sq. km. It extends between 89°50' to 94°0' east longitudes and 22°44' to 25°58' north latitudes with maximum length and width of 460 km and 350 km. The Barak River raises from the Manipur hills, in Senapati district of Manipur at an elevation of 2,331 m. It then flows along the Nagaland to Manipur border through hilly terrains and enters Assam, and turns westward again near Lakhipur where it enters the plains of Barak valley and then flows west past the Silchar town.

The six gauging sites are emphasized in the present study out of which three are stressed stringently for their main contribution to flooding. They are Badarpur Ghat (BPG), Matijuri (MTJ) and Annapurna Ghat (APG) of which one hour interval concurrent flow data for five years (2000 to 2005) pertaining to monsoon period starting from 1st June to 15th October has been collected from the Central Water Commission (CWC), Shillong.

III. RBFN MODULE

Radial basis function networks (RBFN) is having a strong mathematical foundation and is one of the popular neural networks with diverse applications in various fields. RBFNs provide a technique in mappings multidimensional spaces for the approximations of non-linear functions and in the sense of neural learning they are likely to be faster than other ANN [20]. One major advantage of RBFN is possibility of choosing suitable hidden unit without having performed a full nonlinear optimization of the whole network. The centers as well as the widths of the Gaussian’s are set by unsupervised learning rules and then the supervised learning is applied to the output layer [12].

RBFN typically is a three layers neural network having an input layer, a hidden layer with the non-linear RBFN activation function and with a linear output layer as shown in Fig.1.

![Fig.1: Architecture of RBFN network](image)

The input can be modeled as a vector of real numbers. The output of the network is then a scalar function of the input vector. Some of the commonly used radial basis functions are: the Gaussian, the multi-quadratic and the inverse multi-quadratic. The Gaussian function, which is the most widely used one, is used in this study; the output of the RBF network can best described by the equation (1) below:

\[ y(t+x) = \sum_{j=0}^{k} w_j \phi_i \] (1)

Where, \( w_j \) are the weights between the \( j \)th input layer neuron to the \( j \)th hidden layer neuron and \( y(t+x) \) is the predicted output signal at time \( t \) with delay \( x \).

\( \phi_i = \frac{1}{\left( \frac{\| x(t) - c_i \|^2}{\sigma_i^2} \right)^{\frac{m}{2}}} \) (2)

Here \( \phi_i \) is the Gaussian function with input \( x(t) \) and \( c_i \) and \( \sigma_i \) are the center and the width of the \( i \)th neuron in the hidden layer, and \( \| \cdot \| \) denotes the Euclidean distance. The weight vector \( w_i \) between the input layer \( i \)th and the hidden layer neuron corresponds to the center \( c_i \) in equation (2). In relating to the time sequence of the flood flows and its complexity, the reconstructed phase of the both the input \( x(t) \) and center \( c_i \) are defined by A. W. Jayawardena, P. Xu and W. K. Li [6]:

\[ X[t] = [x(t), x(t-\tau), x(t-2\tau), x(t-3\tau), \ldots, x(t-(q-1)\tau)] \] (3)

\[ C[t] = [c(t), c(t-\tau), c(t-2\tau), \ldots, c(t-(q-1)\tau)] \] (4)

Where, \( q \) is the dimension embedded into a time series phase space and sub-scripted \( i \) with center \( C \) is the number of centers (decided as per the given pattern of the problem in search of its optimal value).

The basic learning paradigms from the given structure of RBFN are: (i) the centers of the RBF activation functions, (ii) the spreads of the Gaussian RBF activation functions and (iii) the weights from the hidden to the output layer.
IV. PRNN MODULE

Generally, recurrent neural networks are mainly of two types (i) partially recurrent neural network (PRNN) and (ii) fully recurrent neural network (FRNN). The present study focuses on PRNN with the feedback from the output to the input layers in modeling flood forecasting task. These PRNN models retain the past output of nodes instead of retaining the past raw inputs [18]. As also described in Almeida [5] fully recurrent networks (FRNN) use unconstrained fully interconnected architectures and learning algorithms that can deal with time-varying input and output in non-trivial ways. Many literatures suggested that fully recurrent networks are still complicated when dealing with complex problems even though several modifications are made in learning algorithms to reduce the computational expense. Therefore, partially recurrent networks (PRNN) are chosen, whose connections are mainly feed forward, but include a carefully chosen set of feedback connections. These recurrences allow the network to remember cues from the past without complicating the learning excessively. The structure of the PRNN in the study is shown fig.4.

![Fig.4: Architecture of PRNN network](image)

To forecast hydrologic time series the previous values of the series are required depending on the number of persistence component. PRNN provide this facility through its feedback loop. Thus, the output of the network not only depends on the current inputs it receives but also on the state of the neurons in the previous time step. A PRNN can learn temporal sequences, either online or in batch mode. In general, a PRNN can be expressed with the following equations:

\[ x(t) = f_{\phi}(x(t-1), u(t)) \] (5)

\[ y(t) = f_{\phi}(x(t-1), u(t)) \] (6)

Where, the equations above are for single-input single output and have N hidden state units. The state vector and the network output at time t are denoted by x[t] and y[t] respectively. However, for the multi-input multi-output units the equations may be written as:

\[ y_j(t) = g(\text{net}_y) \] (7)

\[ \text{net}_y = \sum_j w_{yj} y_j(t) + \sum_k u_{yk} y_k(t-1) + \phi \] (8)

Where j and k are the index variable for the hidden and the output layer, \( w_{yj} \) and \( u_{yk} \) are the weights through the output layer with the additional recurrent weights from output to the input layer and \( \phi \) is the bias.

V. MODEL APPLICATIONS

The models are applied to lower network of the Barak river system. Recorded hourly discharge data for three gauging sites of Badarpur Ghat, Matijuri and Annapurna Ghat and are graphically shown in Fig.5.

![Fig.5: Recorded hourly discharges at the three gauge stations (a), (b) and (c) in Barak basins](image)

It was observed that the discharge fluxes altered in all the gauging sites pertaining to the monsoon seasonal flooding. For sustaining the accuracy level with the models large numbers of data are collected (described in section II) and used in models prediction so that the models can capture the underlying nonlinear interactions and complex effective variables.

VI. NETWORK TRAINING AND VALIDATION

The network is trained using the hourly concurrent discharge data for the three gauging sites. First 50% of the data set is used for training, next 25% for cross validation and remaining 25% for testing the performance of trained network. The validation and testing data sets are extracted randomly from the existing training data. The networks are trained with flow rates at time t as input and flow rates at time t+1 as output.
desire output. In both the case one year data are simultaneously applied for training, validation as well as the simulation part. Determination of suitable network architecture is one of the most significant and as well one of the most difficult tasks in the model building process[8], thus, for both the models, the networks are designed on trial and error basis. The parametric constituents includes variations of processing elements from 3-8, momentum rate α=0.7, iteration 200 – 5000 using the tan h and sigmoid transfer functions at the hidden and output layer. Now for the RBFN part, the network are designed with learning rate μ varying from 0.1-0.3, clusters center from 15-50, iterations from 200-10,000.

The PRNN is trained with back propagation through time (BPTT) while the RBFNs are trained by adjusting the centers and widths of the Gaussian’s set by unsupervised learning rules and the supervised learning is applied to the output layer. For standard RBFN, the supervised segment of the network only needs to produce a linear combination of the output at the unsupervised layer. Finding an appropriate number of Gaussians is impossible, because it is problem dependent that is selected for the training of hidden and output layers. The PRNN steps size were varied from 0.01 to 3 whereas they are kept as default at 1.0 and 0.1 in RBFN for the unsupervised and supervised learning rules, because of their non-effective with their changes in training the network.

VII. RESULTS AND DISCUSSIONS

The two models are successfully implemented in forecasting the flood flows at a multiple sections with several numbers of parameters. The parameters are adjusted/tuned by trial and error. To evaluate the model performances various statistical performance indices are employed; root mean square error (RMSE), coefficient of efficiency, (CE) and correlation coefficient (COR) and peak flow criteria (PFC). After careful examination and observations during the network training the outcomes for both the models are halt with different parameters to give the best output. The best network output for both RBFN and PRNN at 1-hour lead ahead are displayed in Fig.6 and Fig.7 respectively, however the next 2-hour and 3-hour lead time ahead could not be display due to limited page but the results are summarized in Table 1 for proof of validation.
TABLE 1: PERFORMANCES OF PRNN AND RBFN MODELS IN FORECASTING RIVER FLOWS FOR MONSOONS PERIOD WITH A LEAD TIME OF 1-HOUR, 2-HOUR AND 3-HOUR AHEAD ON BARAK BASIN NETWORKS

<table>
<thead>
<tr>
<th>Model</th>
<th>Model form</th>
<th>Lead Time</th>
<th>Forecasting station</th>
<th>RMSE</th>
<th>COE</th>
<th>CORR</th>
<th>PFC</th>
</tr>
</thead>
<tbody>
<tr>
<td>RBFN</td>
<td>MIMO (Multiple input multiple output)</td>
<td>1h lead</td>
<td>BPG</td>
<td>19.66177476</td>
<td>0.995183114</td>
<td>0.997792626</td>
<td>0.017675423</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2h lead</td>
<td>MTJ</td>
<td>9.423227685</td>
<td>0.981952368</td>
<td>0.991201919</td>
<td>0.015533181</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3h lead</td>
<td>APG</td>
<td>16.39620834</td>
<td>0.993375026</td>
<td>0.997088645</td>
<td>0.020013451</td>
</tr>
<tr>
<td>PRNN</td>
<td>MIMO (Multiple input multiple output)</td>
<td>1h lead</td>
<td>BPG</td>
<td>27.2054427</td>
<td>0.990849197</td>
<td>0.996453867</td>
<td>0.020768751</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2h lead</td>
<td>MTJ</td>
<td>18.72755829</td>
<td>0.932885255</td>
<td>0.980518338</td>
<td>0.055887367</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3h lead</td>
<td>APG</td>
<td>16.32201154</td>
<td>0.988622015</td>
<td>0.9955146</td>
<td>0.023922604</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1h lead</td>
<td>BPG</td>
<td>43.07683825</td>
<td>0.977066206</td>
<td>0.991971761</td>
<td>0.026513636</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2h lead</td>
<td>MTJ</td>
<td>26.36831773</td>
<td>0.866985352</td>
<td>0.932594824</td>
<td>0.058030904</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3h lead</td>
<td>APG</td>
<td>21.96523607</td>
<td>0.979393229</td>
<td>0.986968535</td>
<td>0.029275705</td>
</tr>
</tbody>
</table>

RMSE=Root mean square error, COE= Coefficient of efficiency, CORR=coefficient of correlation & PFC=Peak flow criteria

The processing elements (PEs) for the RBFN are found to be satisfactory at 5, 50 and the learning rate =0.3, 0.01 the cluster center is found to be satisfactory at 35, 40 with 10000, 2000 numbers of iterations. The analysis of results clearly shows that changing the value of α (i.e. momentum factor) affected the total training time slightly for both the models which resulted in no significant differences in the final weights of the calibrated network. Further, the iterations for the training set for the RBFN network is associated with both supervised and the unsupervised learning rules out which both rules are found to be satisfactory at 5000, 1000 epochs each that contribute the overall 10000 and 2000 iterations for a solitary training session. The decay of the weights are made trial and error as discussed in section VI, however, the acceptable decayed is found to be 0.0001 that precedes the previous values of 0.001. Accordingly, the Table 1 gives a clear discrepancy between the two models that provide their accuracy levels in terms of different performance indices viz. RMSE, COE, CORR and PFC.

The number of hidden layer in all cases is selected as 1(one) since the performance of the forecasting models with single-hidden-layer network is sufficient in approximating most classes of nonlinearities encountered in practice [4],[15] and [19].

VIII. CONCLUSIONS

It is observed that the performances of both the models are better almost in all counts considered in the study. However the indication for the increase in time horizon for both the models leads to alternation in their performances outcomes. For RBFN the number of K basis functions do not need to be equivalent to the N number of the training data points, as it is much better with lesser K than that of N.

Successfully both the model achieves high accuracy forecasting results at 1 to 3 h ahead for MIMO-model where PRNN gives slight satisfactory performance results than that of RBFN at higher time horizons. Besides, the PRNN is found to be faster in training but with steady convergence.

REFERENCES

neural networks”, Neural Networks 13 (2000) 765±786, PII: S0893-6080(00)00048-4


