P2B.11 ADVANCES IN RADIAL BASIS FUNCTION NEURAL NETWORK ALGORITHM FOR RADAR RAINFALL ESTIMATION

Gang Xu * V. Chandrasekar Wanyu Li

Colorado State University
Fort Collins, CO 80523, USA

1. INTRODUCTION

The rainfall on the ground can be potentially dependent on the four dimensional (space-time) structure of precipitation aloft. Therefore the rainfall estimation on the ground can be treated as a function approximation problem. It has been demonstrated that neural networks can be successfully applied to the problem of radar rainfall estimation, modeling it as a function approximation problem. Xiao and Chandrasekar (1995, 1997) first successfully applied the back-propagation (BP) neural networks into ground rainfall estimation using the horizontal radar reflectivity of 1km aloft. One disadvantage of the BP neural network is the slow training speed. To overcome this problem, Liu et al (2001) developed the adaptive Radial-Basis-Function (RBF) neural network. This algorithm can also be incorporated with adaptivity such that the neural network can respond to variation in the relationship between the radar reflectivity and the ground rainfall rate. This type of RBF neural network has also been successfully applied to ground rainfall estimation using vertical profile of the radar reflectivity (Xu et al, 2001 and Li et al, 2002). Li et al. (2003) recently demonstrated that equispaced vertical radar reflectivity from 1km to 4km height above the gage (see Figure 2) is the best input vector to RBF neural network for radar rainfall estimation, compared to several other choices,

The training process for adaptive RBF neural networks involves two steps: building the initial RBF model and adaptively updating RBF models (Liu et al, 2001). It is a nontrivial task to determine the parameters of the initial neural network such as the number of centers in RBF neural networks and the widths of RBF centers (Liu et al, 2001). To address this problem Orr (1998) proposed a method to optimize the width of RBF centers by explicit search. Two principles are important during training neural networks: the first one is to avoid overfitting to training data and the second one is to tune the neural network for optimal performance. It will be desirable to develop an automatic scheme to build the initial network and adaptively update the models to eliminate the manual tuning effort. This paper describes an alternate RBF algorithm (Blevins et al, 1993) to automatically build the initial RBF model. Subsequently this model is adaptively updated for radar rainfall estimation. RBF internal parameters, including centers and widths of the RBF as well as the number of centers, can be autonomously determined during the training phase. This aspect is the main distinction compared to the algorithm of Liu et al (2001). This algorithm also includes an embedded mechanism to avoid the overfitting during the training process.

We have applied this algorithm to the ground rainfall estimation from the WSR-88D radar data in 1998. We adaptively train and apply RBF models to estimating the daily and hourly rainfall accumulations from vertical profiles of reflectivity (1km to 4km) and compare them with ground gage data. Application to one year of WSR-88D radar data over Melbourne, FL shows that this new radar rainfall neural network can estimate average daily accumulation to a normalized standard error of 25% with negligible bias. Results are also compared with WSR-88D Z-R and the best Z-R algorithm that can be determined after the fact.

The following sections of this paper are organized as follows: Section 2 briefly reviews the RBF algorithms proposed by Blevins et al (1993) and describes how we apply these algorithms into the adaptive scheme for radar rainfall estimation. Section 3 presents the application of the new RBF algorithms to observational data from WSR-88D radar during 1998 and the results of performance evaluation. Section 4 demonstrates the tuning procedure to optimize RBF networks. Section 5 summarizes conclusion of this study.

2. ALGORITHM AND METHODOLOGY

Blevins et al (1993) proposed a training algorithm for automatically determining the number, the loca-

*Corresponding author address: Gang Xu, 1373 Campus Delivery, Colorado State University, Ft. Collins, CO 80523; e-mail: xgang@lamar.colostate.edu.

tion and the widths of RBF centers with an embedded mechanism to avoide overfitting. During the training process, the neural network can grow by appropriately adding the new neuron at the location in the input domain where the estimation error is the largest. On the other hand, overfitting may be prevented by building networks that are limited to the level of uncertainty in the system being modeled.

This training algorithm involves two steps, one for initializing and the other for updating the neural network, described as follows:

Step 1. Initialization of network: We first select n initial centers by clustering input samples using k-means. Then we set initial widths of RBF centers by the maximum Euclidean distance over all input vectors in each cluster. n unknown weights are subsequently determined by solving the linear system with n samples.

Step 2. Adjustment of network parameters and addition of new neurons: We first compute the output of the network and examine if the total squared error meets the given threshold. If the threshold is not met, the parameters are adjusted using the gradient-based methods and all the RBF parameters are updated accordingly. The RBF centers can be automatically determined by periodically examining the rate of convergence. The rate of convergence is defined as the normalized difference between total squared errors of the last and first epochs normalized by the training period. When the rate of convergence is below the threshold a new RBF node could be added at the location in the input domain where the estimation error is the largest. The overfitting is reduced by the addition of a new node contingent on a statistical test providing allowance for the error due to system uncertainty. The training is complete when the rate of convergence is small enough and no new node need be added. The other criterion to decide the network convergence is the conventional threshold of the total squared error, as described above.

The block diagram for adaptive radar rainfall neural network (RRN) is shown in Figure 1. After we have established the initial RBF model, which includes RBF centers, widths and weights, we can switch the neural network between two modes: application mode for radar rainfall estimation and the updating mode when new data sets are available. In the updating mode, the RBF network can be incrementally trained from the existing model using the new data sets.

The new RBF algorithm with the adaptive scheme for radar rainfall estimation is applied as follows: first both step 1 and step 2 are applied on data from the first day to obtain an initial RBF network. Subsequently step 2 is applied to new data to progressively train and update the network on a daily basis. The input to the neural network is the vertical reflectivity profile at the height from 1km to 4km above the gage and with the 1km resolution (Figure 2). During the application phase, we have each day's vertical reflectivity profiles as the input to the RBF network, which is configured using the model (RBF centers, widths and weights) adaptively trained up to the previous day. The output of the network is the rainfall rate estimation and will be used to compute average hourly and daily rainfall accumulations over the gages. The estimated rainfall accumulations are compared with gage observations to evaluate the RRN performance. The statistics of comparison (bias, correlation coefficient and normalized standard error) are evaluated to quantify the RRN performance.

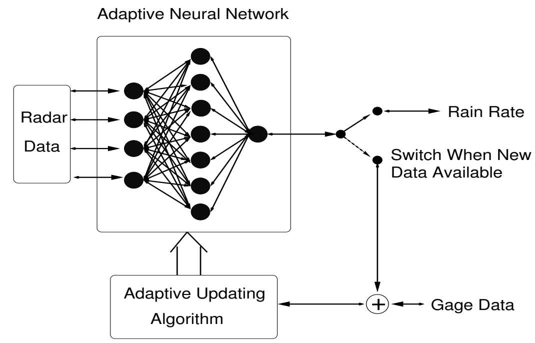

Figure 1: Adaptive neural network scheme. After the initial neural network is established, it can be switched between two modes: application mode and updating mode. When the new radar and gage data are available, the network can be adaptively updated.

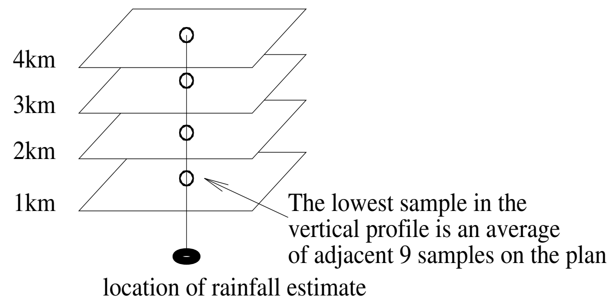

Figure 2: Vertical profile of radar reflectivity above gage as the input vector to RBF neural networks.

3. Data Analysis

3.1 Radar And Rain Gage Data

The data sets are collected by WSR-88D radar located at Melbourne, Florida and the gage networks in the vicinity of the radar, which are part of the Tropical Rainfall Measuring Mission (TRMM) ground validation program. The gages are from Kennedy Space Center (KSC), South Florida Water Management District (SFL) and St. Johns Water Management District (STJ). Totally 109 gages within the range of 125km from the radar are used in this study. The sampling resolution of gage data is one minute. Since the WSR-88D radar takes approximately five minutes to finish a volume scan, five-minute integration is performed on rain gage samples to obtain the average rainfall rate during each radar volume scan. The five-minute average rainfall rate is used as the target output to RBF neural network. The RBF neural network is adaptively trained and tested, as described in the previous section, using the data sets collected during the twelve months of 1998. The daily or hourly rainfall accumulations are averaged over all the testing gages that observe rainfall.

3.2 Results

The average daily and hourly rainfall accumulations from RRN are compared with gage observations. The performance is quantitatively evaluated by computing statistics such as bias, correlation coefficient (CORR) and normalized standard error (NSE). The results of the adaptive RBF algorithm described in this paper-are also compared with best Z-R fitting. Comparison against WSR-88D standard Z-R algorithm ($R = 0.017 * Z^{0.714}$ mm/hour) was also done for completion. The best Z-R relationship, $R = a * Z^b$, is obtained by the nonlinear fitting of the data collected by the randomly selected half of the gage sites. All estimation and tests are conducted over the rest of the gage sites. For the best Z-R fitting, the fitting data and the testing data are from the same day. Table 1 shows the average hourly and daily rainfall accumulations by three algorithms. From Table 1 it can be seen that the new RBF algorithm can estimate average daily accumulation to a normalized standard error of 25% with negligible bias. The adaptive RBF algorithm can perform as well as the best Z-R that can be determined after the fact.

Table 1: Average rainfall estimation comparisons between three algorithms. Data are from WSR-88D radar during a one-year period (January 9, 1998 - December 31, 1998). CORR is the correlation coefficient, NSE is the normalized standard error.

Average daily rainfall accumulation			
algorithm	bias (%)	CORR	NSE
WSR-88D Z-R	28	0.87	0.34
best Z-R fitting	-5.0	0.87	0.22
adaptive RBF	2.1	0.83	0.25
Average hourly rainfall accumulation			
algorithm	bias (%)	CORR	NSE
WSR-88D Z-R	32	0.87	0.42
best Z-R fitting	0.1	0.82	0.35
adaptive RBF	2.7	0.83	0.38

4. OPTIMIZATION OF RBF NETWORK PERFORMANCE

As discussed above, the neural network need be tuned for optimal performance while avoiding overfitting. Optimizing the network performance is a tedious and nontrivial process. We desire to design the algorithm such that, the free parameters that can be altered are kept very few.

In the new RBF algorithm described here, there are mainly two free parameters that are used for the RBF network learning, namely, the learning rate and the period. The period refers to the number of training epochs in which the rate of convergence needs to be calculated and examined, which leads to the decision, if a new RBF node need be added. Learning rate is the conventional updating rate for any gradient-based neural network learning algorithm (Haykin, 1999). Generally the learning rate has to be adjusted for optimizing the network's performance to a specific application. Small learning rate results in slow training, while large learning rate results in over-adjustment and unstable learning. When we decrease the learning rate to ensure the smooth learning, we should accordingly increase the period to ensure enough learning cycles for a fair examination on the rate of convergence. The optimizing procedure for the RBF network is conducted by starting with a relatively large value of learning rate and a small period. Then we stepwise decrease the value of learning rate and increase the period. The adjustment of these two parameters reflects the trade-off between the two learning scenarios of finer and faster learning. In the current study we have used the the data during the first half year of 1998 to optimize the RRN and the results are presented in Figure 3.

Starting with a learning rate of 10^{-2} and the value of 10 epochs for period, we gradually decrease the value of learning rate to 10^{-5} and accordingly increase the value of period to 60 epochs. For learning rate less than 10^{-4}, the performance of the network does not show significant enhancement. The values used in the analysis presented in the previous section are learning rate of 10^{-4} and a period of 30 epochs.

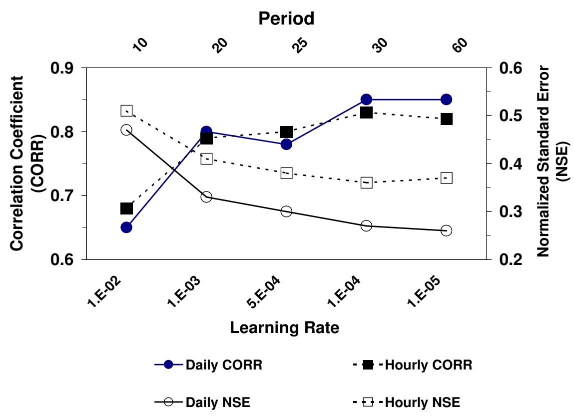

Figure 3: Performance (CORR and NSE) versus tuning parameters (learning rate and period) for optimizing RBF neural network. The biases are all negligible (less than 5%). Data are from WSR-88D radar during a half-year period (January 9, 1998 - June 30, 1998). CORR is the correlation coefficient, NSE is the normalized standard error.

5. SUMMARY

This paper describes an algorithm to automatically build the initial RBF model as well as adaptively update the models for radar rainfall estimation. This model differs from that of Liu et al (2001) that both centers and widths of RBFs are adaptively updated. The input to the network is the equispaced vertical radar reflectivity from 1km to 4km height above the gage, while the output is radar rainfall rate averaged over five minutes. This algorithm also includes an embedded mechanism to avoid the overfitting during the training. We have applied this algorithm to the ground rainfall estimation from the WSR-88D radar located at Melbourne, FL for a whole year of the 1998. We adaptively train and apply the RBF models to estimate the average daily and hourly rainfall accumulations over the gage location. Application to one year of WSR-88D radar data over Melbourne, FL shows that this radar rainfall neural network can estimate average daily accumulation to a normalized standard error of 25% with negligible bias and performs as well as the best Z-R that can be determined after each rainfall event.

6. ACKNOWLEDGMENT

This work is supported by the NASA TRMM program.

7. REFERENCES

Blevins, W. A. and Clair, D. C. ST., 1993: Determining the number and placement of functions for radial basis approximation networks. In *Proceedings of the 1993 Artificial Neural Networks in Engineering conference*, St. Louis, MO.

Haykin, S., 1999: Neural Networks: A Comprehensive Foundation. 2nd Edition, Prentice-Hall, Inc., 1999.

Li, W. and Chandrasekar, V., 2002: Rainfall estimation from vertical profiles of reflectivity using neural networks. *IEEE 2002 Int. Geosci. Rem. Sen. Symp., IGARSS '02*, **6**, 3477–3479.

Li, W., Chandrasekar, V. and Xu, G., 2003: Investigations in radar rainfall estimation using neural networks. *IEEE 2003 Int. Geosci. Rem. Sen. Symp., IGARSS '03* (to be published).

Liu, H., Chandrasekar, V. and Xu, G., 2001: An adaptive neural network scheme for radar rainfall estimation from WSR-88D observations. *J. Appl. Meteor.*, **40(11)**, 2038–2050.

Orr, M. J. L., 1998: Optimising the widths of radial basis functions. In *Vth Brazilian Symposium on Neural Networks*, Belo Horizonte, MG, Brazil.

Xiao, R. and Chandrasekar, V., 1995: Multiparameter radar rainfall estimation using neural network techniques. Preprints, *27th Conf. on Radar Meteorology, Amer. Meteor. Soc.*, 199–204, Vail, CO.

Xiao, R. and Chandrasekar, V., 1997: Development of a neural network based algorithm for rainfall estimation from radar observations. *IEEE Trans. Geosci. Rem. Sen.*, **35(1)**, 160–171.

Xu, G. and Chandrasekar, V., 2001: Radar rainfall estimation from vertical reflectivity profile using neural network. *IEEE 2001 Int. Geosci. Rem. Sen. Symp., IGARSS '01*, **7**, 3280–3281.