Selection of an Artificial Neural Network Model for the Post-calibration of Weather Radar Rainfall Estimation

Masoud Hessami, Francois Anctil and Alain A. Viau
Universite Laval

Abstract: A statistical approach, based on artificial neural networks, is proposed for the post-calibration of weather radar rainfall estimation. Tested artificial neural networks include multilayer feedforward networks and radial basis functions. The multilayer feedforward training algorithms consisted of four variants of the gradient descent method, four variants of the conjugate gradient method, Quasi-Newton, One Step Secant, Resilient backpropagation, Levenberg-Marquardt method and Levenberg-Marquardt method using Bayesian regularization. The radial basis networks were the radial basis functions and the generalized regression networks. In general, results showed that the Levenberg-Marquardt algorithm using Bayesian regularization can be introduced as a robust and reliable algorithm for post-calibration of weather radar rainfall estimation. This method benefits from the convergence speed of the Levenberg-Marquardt algorithm and from the over fitting control of Bayes' theorem. All the other multilayer feedforward training algorithms result in failure since they often lead to over fitting or converged to a local minimum, which prevents them from generalizing the data. Radial basis networks are also problematic since they are very sensitive when used with sparse data.

Key words: Artificial neural network, rainfall, weather radar.

1. Introduction

   Accurate measurements of rainfall over time and space are critical for many hydrological and meteorological projects. The most usual tools to monitor rainfall events are raingauges and weather radar. Networks of raingauges provide accurate point estimates of rainfall, when appropriately set, but their usual low density restricts considerably the spatial resolution of the gathered information. The quality of raingauge observations is also susceptible to some error sources especially biological and mechanical fouling, and human and environmental interference (Steiner, Smith, Burges, Alonso and Darden, 1999). Weather radar are much more efficient in providing the space-time evolution of a rainfall event, but they can be contaminated by many factors including ground clutter, bright band,

anomalous propagation, beam blockage, and attenuation (e.g. Zawadski, 1984; Andrieu, Creutin, Delrieu, and Faure, 1997). The effectiveness of weather radar operation is strongly linked to a rigorous calibration (Serafin and Wilson, 2000). The performance of radar rainfall estimation mainly depends on a proper choice of Z-R relationship (Anagnostou and Krajewski, 1998), which may vary from event to event or even within a single storm - where Z is the radar reflectivity factor (mm^6 m^{-3}) and R is the precipitation rate (mm h^{-1}). A recent experience on a proper choice of the Z-R relationship returns to the work of Rongrui and Chandrasekar (1997) who have proposed a neural network based approach to determine a Z-R relationship.

Early on, Wilson (1970) has recognized the strengths and weaknesses of both observation systems and proposed to integrate them in order to enhance the space-time quality of the rainfall information. Since then, various methods have been proposed to achieve such data driven post-calibration of the weather radar rainfall estimation. They can be classified into two main categories: deterministic and statistical. The deterministic approach involves the post-calibration of radar rainfall estimations against raingauge observations (Wilson, 1970; Andrieu, Creutin, Delrieu, and Faure, 1997). The statistical approach includes multivariate analysis (Eddy, 1979) and cokriging (Krajewski, 1987; Seo, Krajewski and Bowles, 1990). Geostatistical approaches are known as the best methods for radar-raingauges data integration but they are usually inefficient in real time, especially when dealing with the sampling rates of one hour or less necessary for urban and small watershed applications. Such methods also rely on a strong human expertise, which can lead to user-dependent results (Bollivier, Dubois, Maingnan and Kanevsky, 1997). Overall, these methods share a similar objective: to somehow perform a post-calibration of the radar estimation using raingauges as ground truth - note that on occasions, raingauges also depart from truth (Steiner, Smith, Burges, Alonso and Darden, 1999).

Several authors have reported the usefulness of artificial neural networks for spatial data analysis. Rizzo and Dougherty (1994) have introduced a method of pattern completion for hydrogeological applications called neural kriging. The possible use of neural networks for geostatistical simulation has been suggested by Dowd and Sarac (1994). Bollivier, Dubois, Maingnan and Kanevsky (1997) have discussed the application of artificial neural networks in the case of the interpolation of a geo-referenced variable. More recently, Hessami, Anctil and Viau (2002) have used a combination of fuzzy inference system and artificial neural networks based on a Jack-Knife regularization for the post-calibration of weather radar data.

In this paper, a statistical-like approach, based on artificial neural networks, is investigated for merging radar rainfall estimations and raingauges observations.
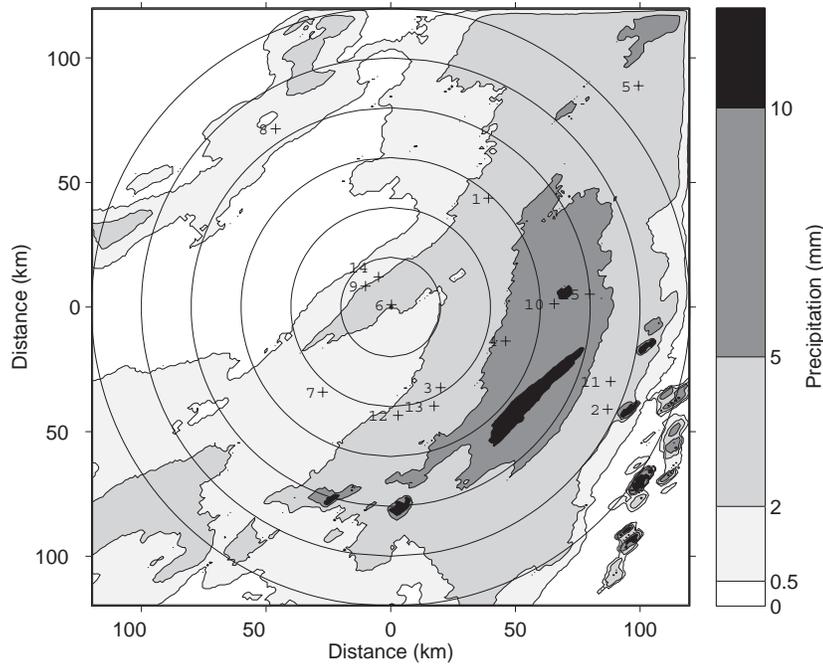
Figure 1: $CAPPI$ of the 8 AM to 9 AM rainfall accumulation: June 17 1997. Crosses indicate raingauge locations.

The object of this approach is to map the input space (radar) to the output space (raingauges) through a proper artificial neural network model, in order to achieve a post-calibration of the weather radar rainfall estimation.

The remainder of the paper is organized as follows. In section 2, a brief description of the selected data is presented. Section 3 introduces the artificial neural networks used for data integration, namely multilayer feedforward networks and radial basis functions. Section 4 provides the results of the comparison of the different neural network models tested. In section 5, the artificial neural networks evaluation is discussed and conclusions are reported in section 6.

2. Data Selection

McGill Radar Weather Observatory has provided the radar rainfall estimations for this study. The radar, located at the western tip of the Island of Montreal, at Sainte-Anne-de-Bellevue, transmits in the S-band (10 cm). It scans the atmosphere using a regular strategy. Data are collected at 24 elevation angles from $0.5°$ to $34.4°$ every 5 minutes. The reflectivity $CAPPI$ (Constant Altitude Plan Position Indicator) is the radar image for displaying precipitation intensity. The unit of $CAPPI$ is in dBZ or $10 \log_{10} Z$ where Z is the reflectivity

in mm^6/m^3. The $CAPPI$ used for this study is the one-hour rainfall accumulation from 8 AM to 9 AM of 17 June 1997, with a resolution 1 km × 1 km, obtained from an altitude of 2 km (Figure 1). This event has been selected because it is typical of large systems rainfall and because it is well centred on the radar. The precipitation melting layer (bright band) has been avoided by choosing the altitude of 2 km, but the image still suffers from some common problems in particular ground clutter and anomalous propagation. Nevertheless, these imperfections do not affect the general data integration methodology presented here ——the complete radar image correction is outside the scope of this paper. The corresponding hourly raingauge observations used for this study were collected by Environment Canada's network, supplemented by a private raingauge network (Figure 1). These low density networks — 15 raingauges to post-calibrate a weather radar image of 120 km radius — impose a constraint on the methodology selection.

3. Artificial Neural Network

Artificial neural networks are mathematical models of human cognition (Govindaraju, 2000), which can be trained to perform a specific task based on available experiential knowledge. They are typically composed of three parts: inputs, one or many hidden layers, and an output layer. Hidden and output neuron layers include the combination of weights, biases, and transfer functions. The weights are connections between neurons while the transfer functions are linear or nonlinear algebraic functions. When a pattern is presented to the network, weights and biases are adjusted so that a particular output is obtained. Artificial neural networks provide a learning rule for modifying their weights and biases. Once a neural network is trained to a satisfactory level, it can be used on novel data.

Training techniques can either be supervised or unsupervised. Supervised training methods are well adapted for interpolation and extrapolation problems. In this paper, the artificial neural networks used for the post-calibration of weather radar estimations include supervised back propagation (Rumelhart, Hinton and Williams, 1986) and supervised radial basis functions (Powell, 1987).

3.1 Backpropagation algorithm

Backpropagation is the generalization of the least mean square (LMS) algorithm to multiple-layer networks and nonlinear differentiable transfer functions. The multilayer feedforward network is the most used architecture of backpropagation. Feedforward networks typically consist of one or more hidden layers of sigmoid neurons followed by a layer of linear neurons. Such network can approximate any function with a finite number of discontinuities. The general equation,

which describes it, is (Hagan, 1996)

$$a^{(i+1)} = f^{(i+1)}(W^{(i+1)}a^{(i)} + b^{(i+1)}), \quad i = 0, 1, \ldots, m-1$$

where m is the number of layers in the network , $f^{(i+1)}$ are the transfer functions, $W^{(i+1)}$ are neuron weights and $b^{(i+1)}$ are neuron biases. For this study, feedforward networks consist of one hidden log-sigmoid layer and one linear output layer:

$$\begin{aligned} f^{(1)}(s) &= \frac{1}{1 + e^{-s}} \\ f^{(2)}(s) &= s \end{aligned}$$

Neurons in the log-sigmoid layer receive radar estimations (x_i, y_i, r_i) as external inputs where x_i and y_i are coordinates and r_i is the radar rainfall:

$$a^{(0)} = p = \begin{bmatrix} x_1 & x_2 & \ldots & x_n \\ y_1 & y_2 & \ldots & y_n \\ r_1 & r_2 & \ldots & r_n \end{bmatrix}$$

Outputs of the second layer are the network outputs $(a = a^{(2)})$, where the corresponding targets are the raingauge observations (t_i)

$$t = [t_1, t_2, \ldots, t_n]$$

where n is the number of raingauge observations. The feedforward algorithm is provided with the following training set $\{p_1, t_1\}, \{p_2, t_2\}, \ldots, \{p_n, t_n\}$. In other words, networks are trained with data from all 15 raingauges and the corresponding 15 radar rainfall estimations (pixels). The algorithm adjusts the network weights and biases in order to minimize the performance function:

$$F(x) = E[(t-a)^2]$$

where x is the vector of weights and biases. There are several algorithms to update feedforward weights and biases. In this paper, we have used four different variations of gradient descent algorithms [basic gradient descent (GD), the gradient descent with adaptive learning rate (GDA) algorithm (Fine, 1999), the gradient descent with momentum (GDM), and the gradient descent with momentum and adaptive learning rate (GDX)], four different variations of conjugate gradient algorithms [Powell-Beele (CGB) algorithm (Powell, 1977), the Fletcher-Reeves (CGF) algorithm (Fletcher and Reeves 1964), the Polak-Ribiere (CGP) algorithm (Fletcher, 1987) and the scaled conjugate gradient (SCG) algorithm (Moller 93)], the quasi-Newton (BFG), One Step Secant (OSS) algorithm (Battiti, 1992), Resilient backpropagation (RB) algorithm (Riedmiller and

Braun, 1993), Levenberg-Marquardt (LM) algorithm (Hagan and Menhaj, 1994) and Levenberg-Marquardt algorithm using Bayesian regularization ($LMBR$) algorithm.

The performance function, which is commonly used for training a feedforward neural network, is the mean square errors

$$F_e = \frac{1}{n} \sum_{i=1}^{n} (t_i - a_i)^2$$

During training, F_e is minimized. The nonlinear properties of neural network allow fitting the training set to very small errors. However, we call over fitting the process of minimizing F_e to extremes while failing to generalize to novel data. One technique to improve network generalization is called regularization. This method modifies the performance function F by adding an additional term, which consists of the mean sum of squares of the network weights F_w

$$F = \beta F_e + \alpha F_w \tag{3.1}$$

where α and β are objective function parameters (Foresee and Hagan, 1997). Usage of this performance function results in smaller weights, which produce a smoother network response. The problem with regularization is that it is difficult to set the optimum values for the objective function parameters. The Bayesian regularization (BR) proposed by Mackay (1992) automatically set optimum values for objective function parameters. The Levenberg-Marquart algorithm based on Bayesian regularization produces a smooth network at the expense of the sum squared error of network.

3.2 Radial basis networks

Radial basis functions (RBF) are designed to find a surface in a multidimensional space that provides a best fit to the training data (Haykin, 1999). The principle of radial basis functions originates from the theory of functional approximation. Radial basis networks perform non-linear mapping from the input space to the hidden layer (radial basis layer) and linear mapping from the hidden layer to the output space (linear layer). The general equations which describe this network are

$$\begin{aligned} a^{(1)} &= f^{(1)}(||W^{(1)} - p||b^{(1)}) \\ a^{(2)} &= f^{(2)}(W^{(2)}a^{(1)} + b^{(2)}) \end{aligned}$$

where $f^{(1)}(n) = e^{-n^2}$ and $f^{(2)}(n) = n$. The bias b^1 determines the width of the area in the input space to which each neuron responds by using a constant called

spread

$$b^1 = \frac{(-\log(0.5))^{0.5}}{spread}$$
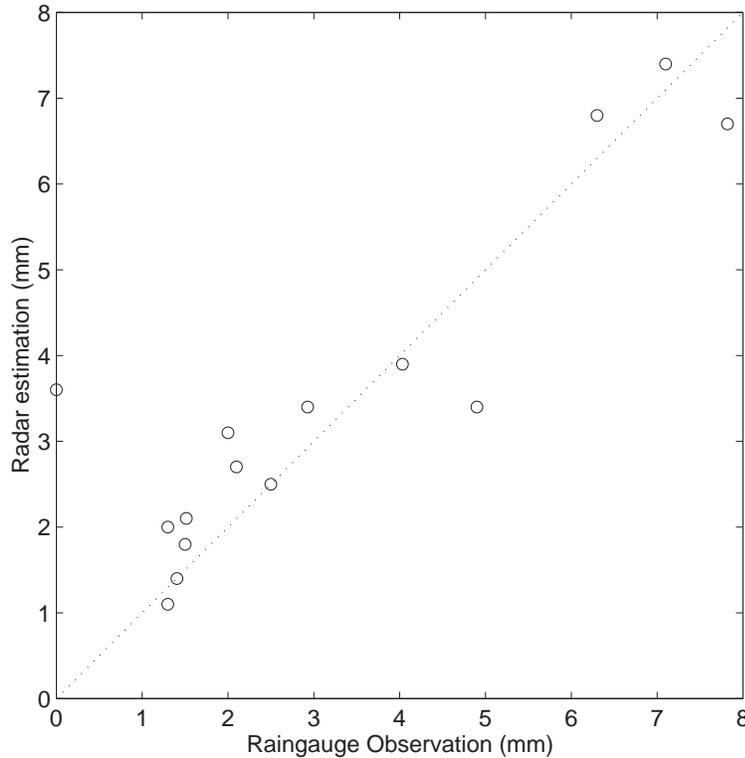
Figure 2: Scatter plot of the raingauge observations and of the weather radar estimations: June 17, 1997 from 8 AM to 9 AM.

It is important to make sure that the *spread* is large enough such that the network can generalize well. The lager the *spread* is, the smoother the network will be. In the radial basis functions (RBF), the number of neurons is determined according to the value of the sum squared error of the training set. In this paper, a variant of radial basis networks called Generalized Regression (GR) Networks is also used. This network has a radial basis layer in the first layer and a special linear layer in the second layer. See Demuth and Beale (1997) and Govindaraju and Ramachandra Rao (2000) for more detailed explanation of the radial basis algorithms.

4. Results

For the selected event, 17 June 1997 from 8 AM to 9 AM, the radar over-estimates the raingauges in most locations (Figure 2), leading to a correlation coefficient of 0.88 between the weather radar estimations and the raingauge observations. Stationary objects nearby raingauge 2 may be the reason for the presence of a null observation versus 3.6 mm estimation. The corresponding mean square error is 1.30 mm^2:

$$F_p = \frac{1}{n} \sum_{i=1}^{n} (t_i - r_i)^2$$

Before training the networks, inputs and targets were normalized to fall within a specified range. All networks were trained with the following 15 training sets

$$\{p_i, t_i\}, \quad i = 1, 2, \dots, 15$$

For multilayer feedforward networks, 1 to 10 neurons were used in the hidden layer and one neuron in the output layer. The selected performance functions was the sum squared error (SSE). The gradient descent algorithms GD, GDA, GDM and GDX (especially GD) were the slowest training algorithms. The advantage of these algorithms is that they only required the computation of gradients. The Levenberg-Marquardt algorithm produced the fastest convergence, but relied on the calculation of second-order derivatives. The quasi-Newton algorithm and Levenberg-Marquardt algorithm often converged too quickly, thus overshooting the point at which the error on the training set is optimum. The conjugate algorithms (CGB, CGF, CGP and SCG), OSS algorithm, RB algorithm, RBF and GR were faster than the gradient descent algorithms but slower than $BFGS$. Table 1 gives the number of hidden neurons (n_1), mean square error (F_e), correlation coefficient between the raingauges and network outputs (R_n), number of epochs and the total training and simulation time for the various algorithms. The feedforward networks have been trained until the mean square error of 0.1 mm^2 was obtained or the number of epochs has reached 1000 or the sum squared error was relatively constant over several iterations. The computations have been performed on a Pentium III 500 MHz.

The common problem with all of these algorithms was that they often led to over fitting or converged to local minimum, which prevented them to generalize the data. These algorithms also required several individual runs before determining the best results. They may be useful if a technique such as early stopping (Coulibaly, Anctil and Bobée, 2000) is used for improving their generalization capability. However, such technique asks for the division of the available data into three subsets: training, validation and test. When the overall data base is small —— for example, in this study, there is 15 raingauges to post-calibrate a 240 km × 240 km weather radar image —— other ways to achieve generalization

Table 1: Post-calibration results for various training algorithms

Algorithm	n_1	F_e (mm^2)	R_n	Epochs	Time (s)
GD	5	0.65	0.93	1000	14.6
GDA	5	0.30	0.97	1000	14.5
GDM	5	0.64	0.94	1000	14.3
GDX	5	0.16	0.98	1000	14.0
CGB	5	0.10	0.99	30	4.7
CGF	5	0.10	0.99	97	5.0
CGP	5	0.10	0.99	118	6.3
SCG	5	0.10	0.99	96	4.6
BFG	5	0.10	0.99	13	2.5
OSS	5	0.10	0.99	172	7.0
RB	5	0.10	0.99	190	3.9
LM	5	0.10	0.99	6	1.4
LMBR	5	0.50	0.95	105	3.7
RBF	10	0.10	0.99	11	2.9
GR	15	0.10	0.99	1	4.2

of the model must be sought. In general, post-calibration using these algorithms resulted in failure.

The Levenberg-Marquardt algorithm using Bayesian regularization was tested with 1 to 10 neurons in the hidden layer. The 5-neuron network was selected since no improvement was made to the network when using more hidden neurons and they lead to similar mean sum of squared errors ($F_e = 0.50$ mm^2). The network response never over fitted the data when the network was over trained or when more neurons were added in the hidden layer. This feature was not true for the other tested training algorithms. In fact, this is the only algorithm which training was not stopped either because convergence was reached ($F_e = 0.10$ mm^2) or because a 1000 epochs had elapsed (see Table 1). Training was stopped after about 100 epochs since F_e was relatively constant over several epochs and the actual training had ceased.

Figure 3 presents the post-calibrated $CAPPI$ using $LMBR$ algorithm with 5 hidden neurons. This figure shows the good generalization capability of the Levenberg-Marquardt algorithm using Bayesian regularization. Figure 4 shows
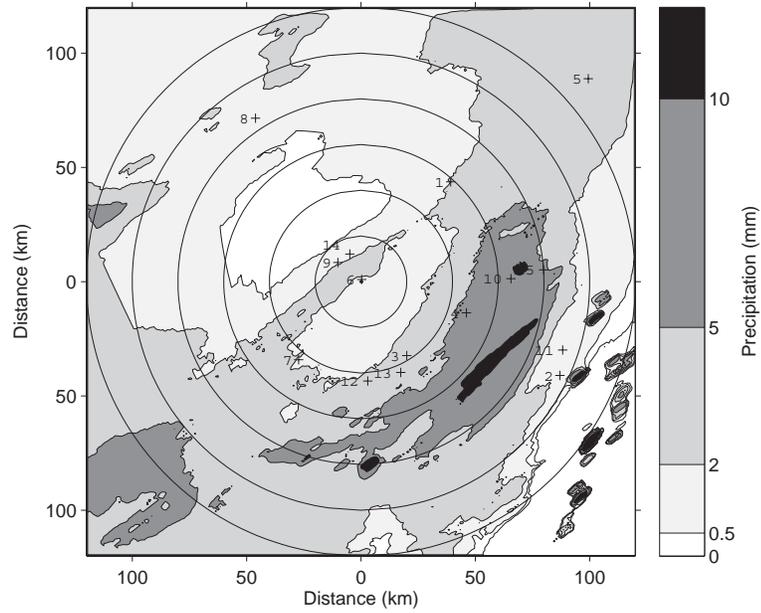
Figure 3: Post-calibrated $CAPPI$ derived from a 5-neuron $LMBR$ network: June 17, 1997 from 8 AM to 9 AM.
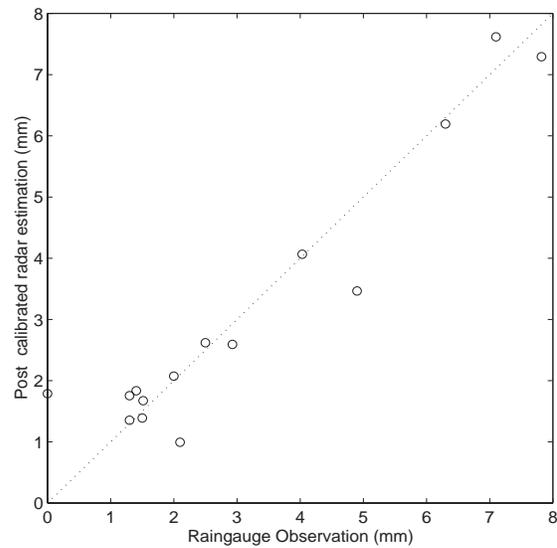
Figure 4: Scatter plot of the raingauge observations and of the post-calibrated weather radar estimations derived from a 5-neuron $LMBR$ network: June 17, 1997 from 8 AM to 9 AM.
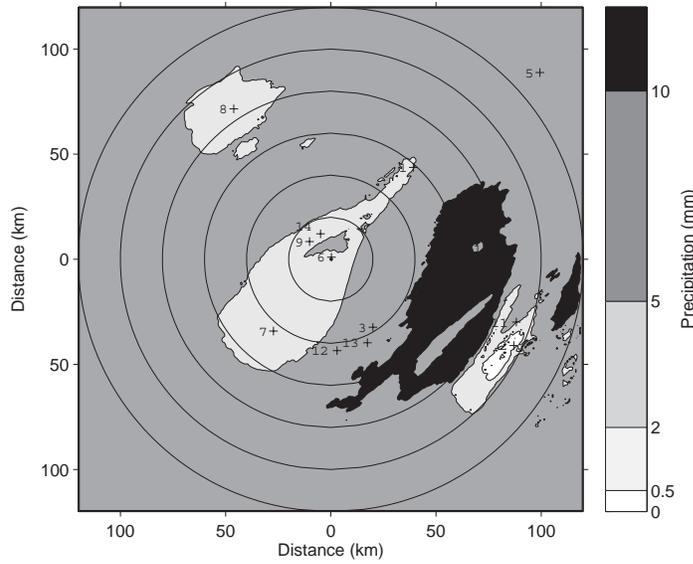
Figure 5: Post-calibrated $CAPPI$ derived from a RB network: June 17, 1997 from 8 AM to 9 AM.
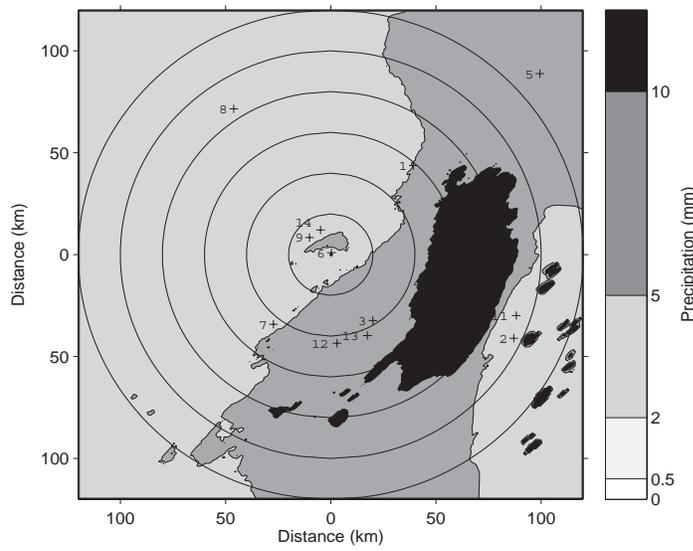
Figure 6: Post-calibrated $CAPPI$ derived from a GR network: June 17, 1997 from 8 AM to 9 AM.

the scatter plot of the raingauge observations and the post-calibrated radar estimations. The correlation coefficient between the outputs of the network and raingauges data was 0.95 compared to 0.88 for the correlation coefficient between the radar and raingauges data (Figure 2). Training results show that the Levenberg-Marquardt algorithm using Bayesian regularization lead to a compromise between both data set —— remember that the raingauge observations are also prone to errors. Note that the post-calibrated value corresponding to raingauge 2 has been significantly adjusted.

In comparison, Figures 5 and 6 show the post-calibration results using two variants of radial basis functions RBF and GR respectively. The RBF network was constructed with a spread constant of 0.9 and a mean square error of 0.1, while a spread of 0.55 was used for the GR network. Radial basis functions networks are very sensitive to *spread*. The optimum value of *spread* must be obtained by trial and error. This can cause problems in the automation process of data integration. Another problem with radial basis functions networks was that they have local support. If we compare Figures 5 and 6 with the original radar image (Figure 1), we can see the networks responses are not reasonable for points far from the raingauges stations. This happens essentially because the value of radial basis function decreases with distance away from its centre. So, radial basis functions are very susceptible to errors when used with sparse data, which is usually the case for the post-calibration of weather radar rainfall estimation.

All trained networks were fed with only 15 raingauges and 15 radar rainfall estimations. Simulations using all radar pixels are ascertain by visual comparison of the post-calibrated radar image with the original calibrated radar images. Any over fitting of the networks is easily detected. In general, results show that the Levenberg-Marquardt algorithm using Bayesian regularization can be introduced as a robust and reliable algorithm for radar-raingauge data integration. This method benefits from the convergence speed of Levenberg-Marquardt algorithm and from the over fitting control of Bayes' theorem. Similar satisfactory results were obtained for the post-calibration of other radar images using the Levenberg-Marquardt algorithm with Bayesian regularization. For example, Figures 7 and 8 are for 6 AM to 7 AM of 17 June 1997 ($F_p = 1.80$ mm^2 and $F_e = 0.68$ mm^2). Table 2 has summarized the results of post-calibration of radar images for the complete event from 5 AM to 1 PM of 17 June 1997.

5. Discussion

The usual approach for evaluating the generalization performance of an artificial neural network is to divide the available data into three subsets: training, validating and testing. The main problem with this approach for the post-
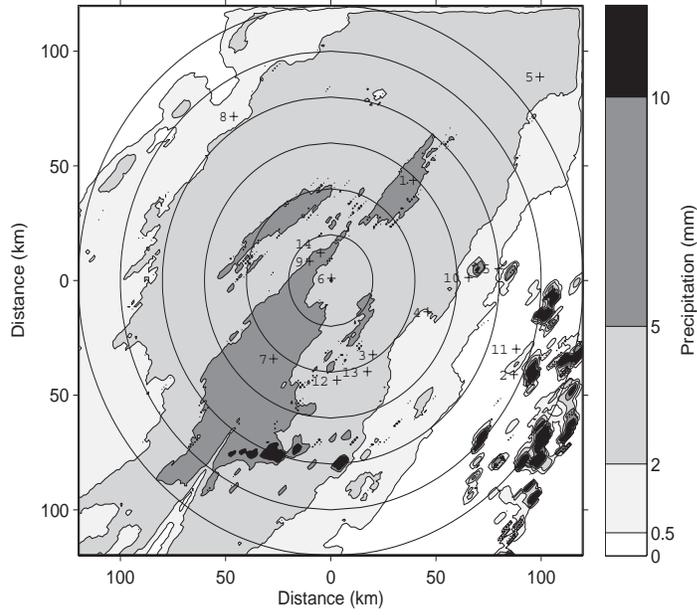
Figure 7: *CAPPI* of the 6 AM to 7 AM rainfall accumulation: June 17 1997.
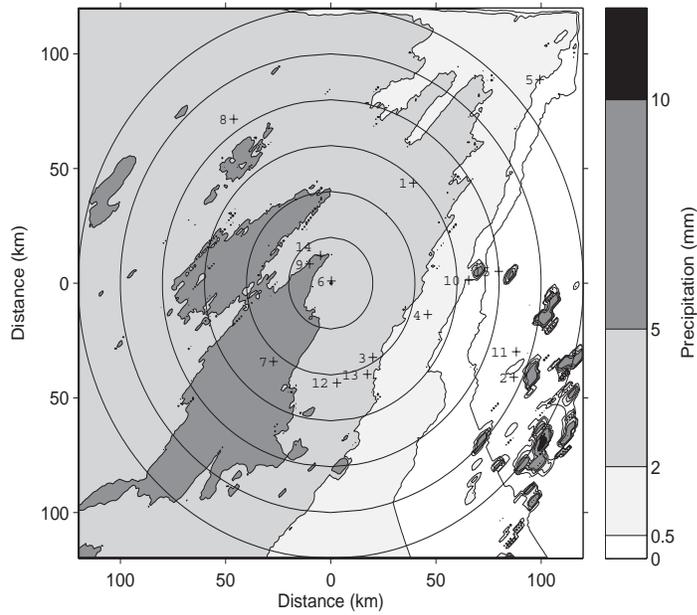
Figure 8: Post-calibrated *CAPPI* derived from a 5-neuron *LMBR* network: June 17, 1997 from 6 AM to 7 AM.

Table 2: Post-calibration results for the LMBR algorithm: June 17,1997, from 5 AM to 1 PM.

Event	Radar vs raingauges Correlation coefficient	RMSE (mm)	LMBR vs raingauges Correlation coefficient	RMSE (mm)
5:00 to 6:00	0.63	1.24	0.84	0.75
6:00 to 7:00	0.85	1.34	0.94	0.82
7:00 to 8:00	0.92	1.48	1.00	0.29
8:00 to 9:00	0.88	1.14	0.95	0.71
9:00 to 10:00	0.87	0.87	0.93	0.63
10:00 to 11:00	0.87	0.65	0.97	0.20
11:00 to 12:00	0.64	1.33	0.82	0.81
12:00 to 13:00	0.67	0.96	0.70	0.79

calibration of weather radar rainfall estimation is that there exists usually a very limited number of raingauge observations to train the network. For example, in this study, there are only 15 raingauge observations available for training. It is thus a waste of valuable information to train the network with a part of the database. Bayesian optimization in the artificial neural network offers an interesting alternative to the standard approach. Since it does not need a testing set nor a validating set, all available raingauge observations can be used for training (MacKay, 1995). The objective function parameters in equation (3.1) control the complexity of the model. These parameters have been already optimized by applying the Bayes' theorem, which gives a probabilistic interpretation to the model.

The Levenberg-Marquardt algorithm using Bayesian regularization offers important advantages over the other interpolation schemes for the post-calibration of weather radar rainfall estimation:

•: This algorithm does not necessarily force the radar image to fit the raingauge observations. The algorithm optimizes the network parameters according to the statistical approach and searches a network response, which is a compromise between the radar estimations and the raingauge observations. This means that the algorithm does not necessarily consider raingauges ground truth.

•: Bayesian regularization provides a probabilistic model to improve the generalization performance of neural networks. Under assumption of normal distribution for network's weights, Bayesian regularization optimises weight decay rates and the neural network produces smooth predicted values for weights.

•: Training and interpolation results can be obtained within just a few seconds using an ordinary personal computer, which is incomparably faster than most interpolation methods, cokriging in particular. Therefore, the algorithm is suitable

for real-time post-calibration.

It is proposed that the proposed artificial neural network may be used as a general data integration and data calibration tool. Tests with other type of database are envisaged.

6. Conclusion

We have tested various artificial neural networks, namely multilayer feedforward networks and radial basis functions networks, for the post-calibration of weather radar rainfall estimation. The process of combining raingauge observations and radar estimations has been performed for hourly accumulation. Except for Levenberg-Marquardt using Bayesian regularization training algorithm, all the other feedforward neural network algorithms presented in this paper required reinitializing and retraining several times to determine the best results. The radial basis functions networks gave poor results because of data sparseness. Results showed that the Levenberg-Marquardt algorithm using Bayesian regularization is robust and reliable for radar-raingauges data integration. This algorithm prevented over fitting the training data set by applying Bayes' theorem, so no testing set or validating set is necessary and all available data can be used for training. Additional tests on several radar images have demonstrated the Levenberg-Marquardt algorithm using Bayesian regularization capability to generalize new situations. It is believed that the proposed artificial neural network may be used as a general data integration and data calibration.

References

Anagnostou, E. N., and Krajewski, W. E. (1998). Calibration of the WSR-88D precipitation processing subsystem. *Weather Forecasting* **13**, 396-406.

Andrieu, H., Creutin, J. D., Delrieu, G., and Faure, D. (1997). Use of a weather radar for the hydrology of a mountainous area, Part I: Radar measurement interpretation. *Journal of. Hydrology* **193**, 1-25.

Battiti, R. (1992). First and second order methods for learning: Between steepest descent and Newton's method. *Neural Computation*, **4**, 141-166.

Bollivier, M., Dubois, G., Maingnan, M., and Kanevsky, M. (1997). Multilayer perceptron with local constraint as an emerging method in spatial data analysis. *Nuclear Instruments and Methods in Physics Research A* **389**, 226-229.

Chiles, J. P., and Delfiner, P. (1999). *Geostatistics: Modeling Spatial Uncertainty.* John Wiley.

Coulibaly, P., Anctil, F., and Bobee, B. (2000). Daily reservoir inflow forecasting using artificial neural networks with stopped training aapproach. *Journal of Hydrology* **230**, 244-257.

Creutin, J. D., Andrieu, H. and Delrieu, G. (1989). Une simplification du cokrigeage appliquee a l'Etalonnage d'Image de teledetection en Hydrometeorologie. *Proceeding, Third International Geostatistics Congress*, Avignon, France. **2**, 761-772.

Creutin, J. D., Delrieu, G. and Lebel, T. (1988). Rain measurement by raingage-radar combination: A geostatistical approach. *Journal of Atmospheric and Oceanic Technology* **5**, 102-115.

Demuth, H., and Beale, M. (1997). *Neural Network Toolbox for User's Guide.* The MathWorks.

Dowd, P. A., and Sarac, C. (1994). A neural network approach to geostatistical simulation. *Mathematical Geology* **26**, 491-503.

Eddy, A. (1979). Objective analysis of convective scale rainfall using gauges and radar. *Journal of Hydrology* **44**, 125-134.

Fine T. L. (1999). *Feedforward Neural Network Methodology.* Springer-Verlag.

Fletcher, R. (1987). *Practical Methods of Optimization*, 2nd edition. Wiley.

Fletcher, R., and Reeves, C. M. (1964). Function minimization by conjugate gradients. *Computer Journal* **7**, 149-154.

Foresee, F. D., and Hagan M. T. (1997). Gauss-Newton approximation to Bayesian learning. *Proceeding of the 1997 International Joint Conference on Neural Networks*, 1930-1935.

Govindaraju, R. S. (2000). Artificial neural networks in hydrology I: Preliminary concepts. *Journal of Hydrologic Engineering* **5**, 115-123.

Govindaraju, R. S., and Ramachandra Rao, A. (2000). *Artificial Neural Networks in Hydrology.* Kluwer Academic Publisher.

Hagan, M. T. (1996). *Neural Network Design.* PWS Publishing Company.

Hagan, M. T., and Menhaj, M. (1994). Training feedforward networks with the Marquardt algorithm. *IEEE Transaction on Neural Networks* **5**, 989-993.

Haykin S. (1999). *Neural Network; A Comprehensive Foundation.* Prentice Hall.

Hessami, M., Anctil, F., and Viau A. A. (2003). An adaptive neuro-fuzzy inference system for the post-calibration of weather radar rainfall estimation. *Journal of Hydroinformatics* **4**, 63-70.

Krajewski, W. F. (1987). Cokriging radar-rainfall and rain gage data. *Journal of Geophysical Research* **92** (D8), 9571-9580.

MacKay, D. J. C. (1992) Baysian interpolation. *Neural Computation* **4**, 415-447.

Mackay, D. J. C. (1995). *Bayesian Non-Linear Modelling with Neural Networks.* University of Cambridge: Programme for Industry, 1-26. http://www.inference.phy.cam.ac.uk/mackay/BayesNets.html

Matheron, G. (1965). *Les Variables Régionalisées et leur Estimation: Une Application de la Theorie des Fonctions Aleatoires aux Science de la Nature.* Masson.

Moller, M. F. (1993). A scaled conjugate gradient algorithm for fast supervised learning. *Neural Network* **6**, 525-533.

Powell, M. J. D. (1977). Restart procedures for the conjugate gradient method. *Mathematical Programming* **12** 241-254.

Powell, M. J. D. (1987). Radial basis function approximation to polynomials. *Numerical Analysis Proceeding.* Dundee, UK, 223-241.

Riedmiller, M., and Braun, H. (1993). A direct adaptive method for faster backpropagation learning: The RPROP algorithm. In *Proceedings of the IEEE International Conference on Neural Networks.*

Rizzo, M. D., and Dougherty, D. E. (1994). Characterization of aquifer properties using artificial neural networks: Neural Kriging. *Water Resources Research* **30**, 483-497.

Rongrui, X., and Chandrasekar, V. (1997). Development of a neural network based algorithm for rainfall estimation from radar observations. *IEEE Transactions on Geoscience and Remote Sensing* **35**, 160-171.

Rumelhart, D. E., Hinton, E., and Williams, J. (1986). *Learning Internal Representation by Error Propagation, Parallel Distributed Processing I.* MIT Press.

Seo, D. J., Krajewski, W. F., and Bowles, D. S. (1990). Stochastic interpolation of rainfall data from rain gauges and radar using cokriging, 1: Design of experiments. *Water Resources Research* **26**, 469-477.

Serafin R. J., and Wilson, J. W. (2000). Operational weather radar in the United States: Progress and opportunity. *Bulletin of the American Meteorological Society* **81**, 501-518.

Steiner, M., Smith, J. A., Burges, S. J., Alonso, C. V., and Darden, R. W. (1999). Effect of bias adjustment and rain gauge data quality control on radar rainfall estimation. *Water Resources Research* **35**, 2487-2503.

Wakernagel, H. (1995). *Mutivariate Geostatistics.* Stringer, Berlin.

Wilson, J. W. (1970). Integration of radar and rain gauge data for improved rainfall measurement. *Journal of Applied Meteorology* **9**, 489-497.

Zawadski, I. (1984). Factors affecting the precision of radar measurement of rain. *22nd Conference on Radar Meteorology*, Zurich, 251-256.

Masoud Hessami
Department of Civil Engineering
Université Laval
Sainte-Foy, Canada, G1K 7P4
masoud.hessami@gci.ulaval.ca

Francois Anctil
Department of Civil Engineering
Université Laval
Sainte-Foy, Canada, G1K 7P4
francois.anctil@gci.ulaval.ca

Alain A. Viau
Department of Geomatic Sciences
Centre de recherche en Géomatique
Université Laval
Sainte-Foy, Canada, G1K 7P4
alain.viau@scg.ulaval.ca