STORMWATER MODELLING USING ARTIFICIAL NEURAL NETWORK

P. C. Nayak, Scientist 'C'
National institute of Hydrology
Kakinada-533 003, AP

Introduction

Simplified rainfall-runoff models are usually applied to represent hydrologic processes over watersheds with scarce hydrometeorological data (Viessman Jr. and Lewis, 1996). Even watersheds with good monitoring systems and more elaborated models are limited in face of complex hydrological processes (Singh, 1988) such as precipitation, evaporation, infiltration and streamflow that are difficult to quantify with good accuracy. Moreover, some surface processes such as streamflow are strongly depended upon the physical characteristics of the watershed like topography, vegetation cover, soil type, hydraulic structures, urbanization, and many others.

Watershed analysis and modeling require extensive time series of streamflow and rainfall data, not frequently available. Many analytical methods have been developed to estimate streamflow from rainfall measurements over the watershed (Viessman Jr. and Lewis, 1996; Franchini and Pacciani, 1991), as well as stochastic and statistical ones where no measurements are available. These simplified methods are usually applied to engineering projects, water resources management, flood forecasting and others. A great deal of the hydrological and hydraulic processes is neglected. Other more elaborated methods such as the Shamseldin et al. (1996) ensemble streamflow estimates from several models. Each of them represents certain important physical aspects. More recently, the ANN method was successfully utilized to model time series in a variety of applications in science and engineering (Vemuri, 1994), also applied to convert rainfall into streamflow (Luk et al., 2000; Hsu et al., 1995; French et al., 1992) without prescribing the hydrological processes; nonlinear systems are handled without explicitly solving differential equations (Hsu et al., 1995).

Urban hydrology are not well represented by deterministic models since it is difficult to quantify processes with equations due to missing or incomplete data. Overall, processes and formulae are simplified and fewer parameters are used to keep the model and required data manageable. Simplified models can make modeling feasible and enhance insight in the processes. When the natural process is not sufficiently known, but data exist, a statistical approach can also be applied. Watersheds with long data time-series can be represented by stochastic models, but noise and changes can reduce modelling capabilities. Furthermore, some issues are not addressed: (1) unknown processes; (2) decision making; (3) data intensive tasks; (4) changing environments.

The ANN can deal with such issues, namely, processing speed, fault tolerance, adaptability and learning capabilities (Loke, 1995). As described by Loke (1995), ANNs relate an input vector (e.g., rainfall accumulation) to an output vector (streamflow) by means of a large number of highly interconnected, simple input and output devices termed neurons. These neurons can be connected in several ways to form special groups of neurons with distinct architectures. The ANN has to be trained through a learning procedure that uses a data set of

input and output vectors to adjust the networks internal parameters. After training, the ANN is verified against an independent data set. If the verification procedure is not satisfactory, the network has to be retrained, using different data sets or modified internal settings. Training and verification are repeated until the ANN performs adequately. As the performance of an ANN is influenced by many parameters (e.g. network structure or learning algorithm), numerous types of ANNs exist, all with their specific application purposes. ANNs are divided into prediction, simulation, classification, optimization and identification problems. ANNs can be used in urban hydrology applications for runoff and flow forecast, flow and pollution simulation, control strategy definition or system parameter identification.

Therefore, the objective of this paper is to apply and ANN to simulate and to forecast flash floods in the Tamanduateı´ river watershed, a heavily urbanized area in Eastern Sa˜o Paulo State (Fig. 1). Rainfall data were obtained from the Sa˜o Paulo weather radar—SPWR (Pereira Filho, 1999). Streamflow were estimated from data measured by an automatic stage level station at the outlet of the watershed (Fig. 1). Athree-layer feed forward ANN (Hsu et

al., 1996)was trained with the LLSSIM(Hsu et al., 1995). Noteworthy, feed forward ANNs have been widely used (Karunanithi et al., 1994; Crespo and Mora, 1993). Their structure is post-defined. Simple structures might not have enough degrees of freedom to learn the process adequately, while complex ones might not converge to realistic patterns.

Artificial Neuron Model

Artificial Neural Networks (ANN) are computational systems whose architecture and operation are inspired from our knowledge about biological neural cells (neurons) in the brain. The transmission of a signal from one neuron to another through synapses is a complex chemical process in which specific transmitter substances are released from the sending side of the junction. The effect is to raise or lower the electrical potential inside the body of the receiving cell. If this graded potential reaches a threshold, the neuron fires. It is this characteristic that the artificial neuron model proposed by McCulloch and Pitts, [McCulloch and Pitts 1943] attempt to reproduce. The neuron model shown in Figure 1 is the one that widely used in artificial neural networks with some minor modifications on it.

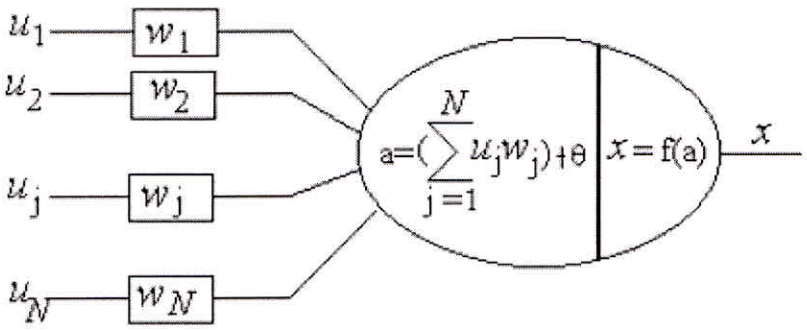

Fig 1 Structure of Artificial Neural Network

The artificial neuron given in this figure has N input, denoted as $u_1, u_2, \ldots u_N$. Each line connecting these inputs to the neuron is assigned a weight, which are denoted as $w_1, w_2, .., w_N$ respectively. Weights in the artificial model correspond to the synaptic connections in biological neurons. The ***threshold*** in artificial neuron is usually represented by θ and the activation corresponding to the graded potential is given by the formula:

$$a = \left(\sum_{j=1}^{N} w_j u_j \right) + \theta \qquad (1)$$

The inputs and the weights are real values. A negative value for a weight indicates an inhibitory connection while a positive value indicates an excitatory one. Although in biological neurons, θ has a negative value, it may be assigned a positive value in artificial neuron models. If θ is positive, it is usually referred as ***bias***. For its mathematical convenience we will use (+) sign in the activation formula. Sometimes, the threshold is combined for simplicity into the summation part by assuming an imaginary input u_0 =+1 and a connection weight w_0 = θ . Hence the activation formula becomes:

$$a = \left(\sum_{j=0}^{N} w_j u_j \right) \qquad (2)$$

The output value of the neuron is a function of its activation in an analogy to the firing frequency of the biological neurons:

$$x = f(a) \qquad (3)$$

Furthermore the vector notation

$$a = w^T u + \theta \qquad (4)$$

is useful for expressing the activation for a neuron. Here, the j^{th} element of the input vector **u** is *uj* and the j^{th} element of the weight vector of w is *wj*. Both of these vectors are of size *N*. Notice that, **w**T**u** is the inner product of the vectors **w** and **u**, resulting in a scalar value. The inner product is an operation defined on equal sized vectors. In the case these vectors have unit length, the inner product is a measure of similarity of these vectors. Originally the neuron output function *f*(*a*) in McCulloch Pitts model proposed as threshold function, however linear, ramp and sigmoid and functions (Figure 1.4.) are also widely used output functions.

Backpropagation Neural Network

Several algorithms of neural network model exist. However, back propagation which belongs to supervised learning algorithm that compares the actual output to the target or specified output and then readjust the weights backward in the network. The same input is specified to the network in the next iteration, so the actual output will be closer to the target output. One of the basic requirements of the BP training is that the transfer function be continuous and differentiable. The sigmoid logistic non-linear function which fulfills the above requirement is used with the value ranging between 0 to1. The standard back propagation training algorithm (Figure 2) is as followed:

(1) Initialize all weights and bias factors to small random values.

(2) Forward pass: Present input vector *(I_1, $I_{2,...}$, I_{no})* and specify the desired output *(t_1, $t_{2,...}$, t_{n1})*

(3) For layer m = 1,2,…,l: , according to Figure 2, we can compute

$$N_{j,m} = \sum_{i=1}^{n_{m-1}} W_{ji,m} \cdot O_{i,m-1} + \theta_{j,m} \qquad (5)$$

where $O_{i,0} = I_i$, t_j = target value of neuron j in output layer, $O_{j,m}$= output of neuron j in layer m, $N_{j,m}$= activation of neuron j in layer m, $\theta_{j,m}$ = bias value for neuron j in layer m , $W_{ji,m}$ = synaptic weight between node j in layer m and node i in layer m-1

The output $O_{j,m}$ of the j^{th} unit in the layer m is computed using a sigmoid function as

$$O_{j,m} = \frac{1}{1+e^{-N_{j,m}}}; j = 1,2,\ldots,n_m \qquad (6)$$

(4) Compute the final output ($O_{1,1}, O_{2,1}, \ldots\ldots, O_{n1,1}$) and compared with the desired output *(t_1, $t_{2,...}$ t_{n1})*. If the difference is acceptable, the process is terminated and the system has learned. Otherwise, continue to next step. When the number of epochs is reached while the difference

is not acceptable, the convergence is not attained. One should try with a new set of initial values, or even modify the structure of the network.

(5) Backpass: for layer m = l, l-1, l-2, …, 1, let $\delta_{j,m} = \delta$ for neuron j in layer m,

For output layer m $\delta_{j,m} = O_{j,m}(1-O_{j,m})(t_j - O_{j,m})$; (7a)

For hidden layer m $\delta_{j,m} = O_{j,m}(1-O_{j,m})\sum_{k=1}^{n_{m+1}} W_{kj,m+1}\delta_{k,m+1}$ (7b)

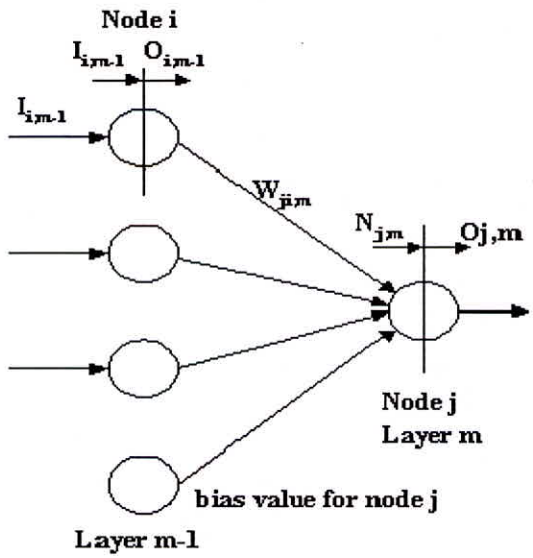

Fig. 2 Transfer of Input of Layer m-1 to A Nodal Output of Layer m

To compute the weight increments:

$$\Delta W_{ji,m}(n+1) = \eta \cdot \delta_{j,m} \cdot O_{i,m+1} + \alpha \cdot \Delta W_{ji,m}(n) \quad (8)$$

where η = learning parameter, α = momentum constant, $\Delta W_{ji,m}(n)$ = weight change between node j in layer m and node i at n iteration, $\Delta W_{ji,m}(n+1)$ = weight change between node j in layer m and node i at $n+1$ iteration, $O_{i,m-1} = I_{i,m}$, n = number of iteration (n = 1,2,3,…).The new values of the weight is computed as

$$W_{ji,m}(n+1) = W_{ji,m}(n) + \Delta W_{ji,m}(n+1) \quad (9)$$

where $W_{ji,m}$ (n) = weight value between node j in layer m and node i at n iteration, $W_{ji,m}$ $(n+1)$= weight value between node j in layer m and node i at n+1 iteration

(6) Go to step 2.

The training algorithm

The linear least squares simplex or the LLSSIM algorithm (Hsu et al., 1995; Gupta et al., 1997)isa hybrid method that optimizes the training of feed forward ANNs. Basically, the LLSSIM splits the weights into two training strategies to minimize the search for the global minima (Scalero and Tepedelenlioglu,1992). The simplex algorithm (Duanetal.,1992) is a random multi-initialization procedure that reduces trapping by local minima (Hsu et al., 1996). For hydrological applications, the data sets are normalized:

$$N_i = N_{\min} + \frac{X_i - X_{\min}}{X_{\max} - X_{\min}}(N_{\max} - N_{\min})$$

where, Xi hydrological variable at the ith time step;

Xmin lowest value of X in the hydrological time series;

Xmax highest value of X in the hydrological time series;

Ni normalized value of X at the ith time step;

Nmin lowest value of N allowed; Nmax highest value of N allowed;

The imposed lower and upper limits of the normalized variables prevent over-shooting (Smith, 1993). The time series of precipitation and streamflow were normalized in the range of 0 to 1 and 0.1 to 0.9, respectively.

Study Area

Fig. 3 shows the Tamanduatei´ watershed, located in Eastern Sa˜o Paulo State, Brazil. It is an important tributary of the Alto Tiete River (Water and Electrical Energy Department (DAEE), 1988). This densely urbanized watershed has a drainage area of 310 km^2 with an

estimated time of concentration of about 4 h. More than 80% of its area is impermeable, especially upstream from the outlet (CTO, 1997). The Tamanduatei´river channel has a regular concrete cross-section that was projected to support a peak flow of around 485 m^3/s for a return period of 500 years (CTO, 1997). As the population grows every year without any effective public policy, so do the urbanization and the flood waves that cause severe inundation.

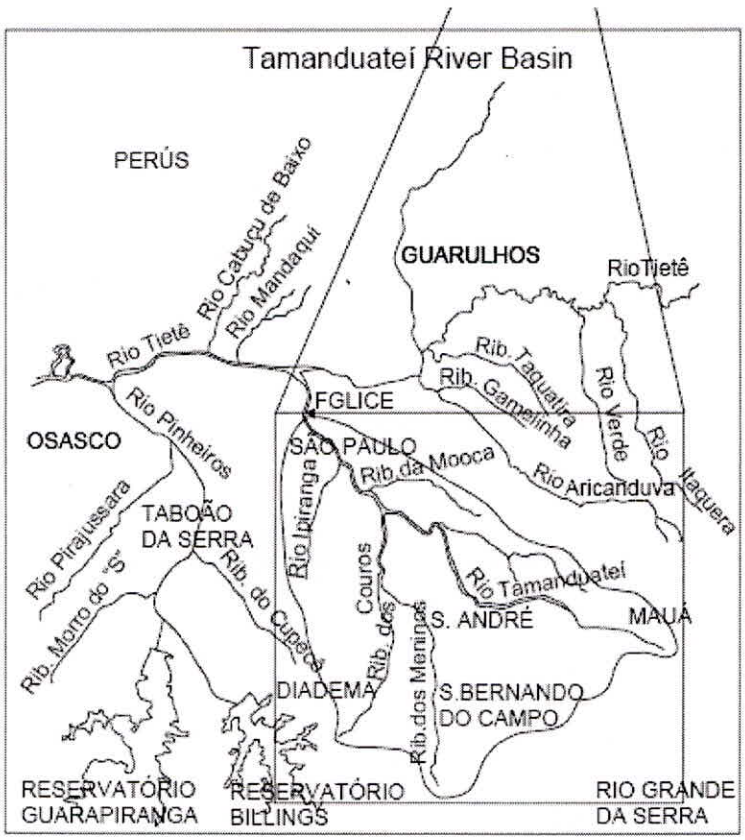

Fig 3: Map showing Study area

The Tamanduatei´ watershed is a part of a much larger urban area termed the Metropolitan Area of Sa˜o Paulo (MASP) with a population of more than 17 million inhabitants. Thermodynamically, it constitutes a huge heat island. Under quiescent synoptic conditions during the rainy season from October to March it tends to induce stronger convection and so more rainfall associated with the local sea breeze circulation which brings moisture from the Atlantic Ocean (Pereira Filho, 1999). With increased precipitation and lower infiltration rates, flash floods are becoming more and more severe with increasing social and economical

loses. Thus, hydrological models such the one proposed in the present work are of great assistance to give support to the civil defense and to the local and state governments.

Data sets

Precipitation accumulations were estimated from radar measurements of reflectivity. The Sa˜o Paulo weather radar (SPWR) is located in the Upper Alto Tiete^ River (Fig. 2). Radar volume scans were made every 10 min. Reflectivity measurements were transformed into rainfall rates by the classical Marshall and Palmer (MP, 1948) ZR relationship. Afterwards, rainfall rates were interpolated to a constant altitude of 3.0 km with a horizontal resolution of 2 kmX2 km. Precipitation accumulation were obtained as follows:

$$P = \sum_{i=1}^{n} \frac{R}{6}$$

where, P precipitation accumulation (mm); R rainfall rate (mm/ h); n number of time steps of ten minutes each.

The SPWR rainfall estimates are susceptible to many sources of errors such as electronic calibration, the use of the MP relationship, bright band and range effect (Pereira Filho, 2003). In this work, radar rainfall estimates were not analyzed together with rain gauges to minimize errors since the available network of rain gauges is sparse and presented many data gaps.

Moreover, rain gauge measurement errors can be large because of the rainfall spatial variability and wind effects for intense convective systems as the ones associated with floods in the MASP and so could not be used as 'ground truth'. In the present work, the SPWR data were used without any corrections because of the limitations above. In this work, flood events with radar rainfall estimations not consistent with the respective flood waves were arbitrarily eliminated. Radar rainfall estimates can be improved by integrating both radar estimates and rain gauge measurements through a statistical objective analysis scheme (Pereira Filho et al., 1998). Pereira Filho and Crawford (1999) have shown that a 30% error in radar rainfall estimation can result in a peak flow error greater than 300% for small flood waves.

The Tamanduateı´ watershed was mapped to the SPWR matrix coordinate system (Fig. 4) to form a mosaic of radar rainfall estimations of 2 kmX 2km horizontal resolution. The area of the watershed was divided up into eight isochrones (Ponce, 1989) of thirty minutes each. Thus, aerial precipitation averages were obtained by simple average of all corresponding 2 kmX2 km rainfall accumulations. This was done to reduce the number of nodes in the ANN input-layer so to improve the convergence rate of the ANN training.

Stage level data were measured every ten minutes at the outlet of the watershed (Fig. 3). Since the river channel has a regular cross-section, an empirical rating curve was used to estimate the streamflow. Time series of both radar derived rainfall and streamflow estimates were taken from the data sets available between 1991 and 1995. All selected flood events are plotted in Fig. 5 one after the other side-by side.

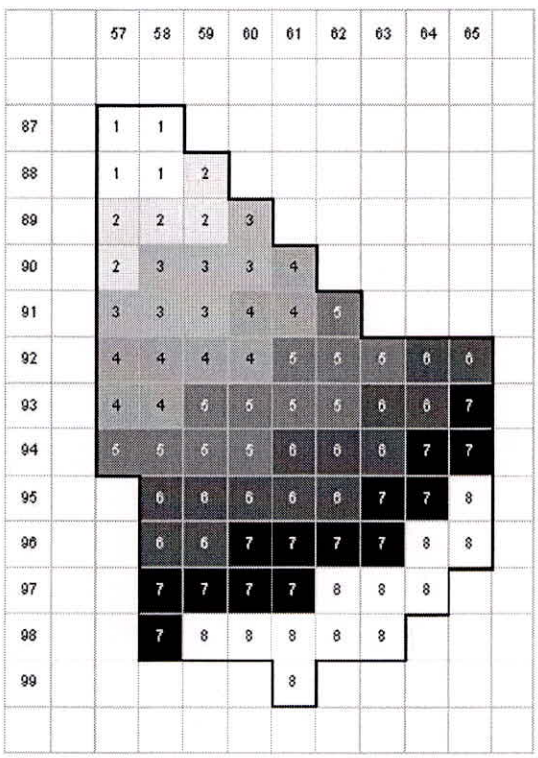

Fig. 4. Radar based coordinate system over the area of the Tamanduateı´ watershed shown in Fig. 3. The squares represent 2 kmX2 km grid cells divided up into eight isochrones.

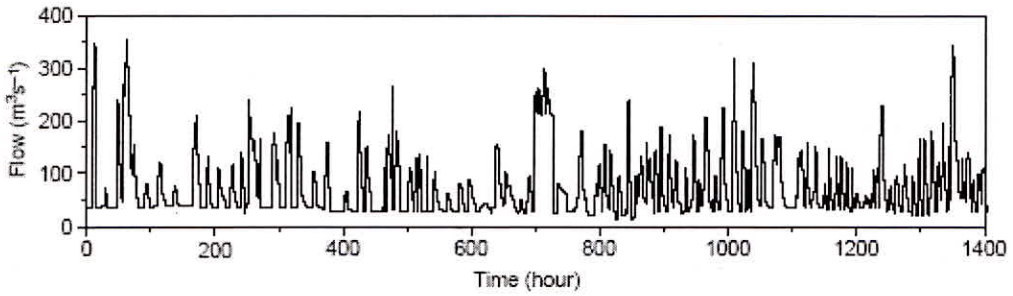

Fig. 5. Time series of streamflow measurements at the outlet of the Tamanduateı´ watershed (Fig. 2). A total of one-hundred flood events were selected from 1991 to 1995.

The feed forward ANN was trained with the LLSSIM method for different input data (Table 1):

1. Rainfall for each isochrone at time steps t to t-Δt and streamflow as the output at time step t.
2. Similar to (1) for streamflow at time step t-Δt asan input.
3. Similar to (1) for stage level.
4. Similar to (2) for stage level.

Table 1
The ANN training options

Training option	Input			Output	
	Rainfall	Streamflow	State level	Streamflow	Stage level
ANN1	Yes	No	–	Yes	–
ANN2	Yes	Yes	–	Yes	–
ANN3	Yes	–	No	–	Yes
ANN4	Yes	–	Yes	–	Yes

Half-hour rainfall accumulations for isochrones 1 to 8 (Fig. 4) for time steps t to $t-7\Delta t$, respectively, are input to the ANN at time step t. Streamflow or stage level input and output to the ANN are given at time steps $t-\Delta t$ e t, respectively.

Initially, a data quality control was performed withthe ANN to identified flood events with poor quality due to precipitation and streamflow measurement errors. Thus, the ANN was initially training with all available data sets in a total of 100 events (Fig. 6) with the configuration (8,1,1). Most events with very larger discrepancies between observed and estimated streamflow were eliminated. This data quality control procedure reduced the number of flood events by 25%. The selected events were randomly divide up into three groups for training, verification and forecast. The later used the observed rainfall as forecasted (perfect forecast).

The verification procedure

Observed and simulated streamflow and stage level time series were compared by means of the mean square difference (MSD). Difference instead of absolute error because the observations such stream- flow or stage level are not free of errors. The MSD can be written in terms of phase and amplitude differences:

$MSD=MSD_a+MSD_p$

where, MSD_a amplitude difference between observed and estimated time series;

MSD_p phase difference between observed and estimated time series.

Fig. 6 shows time series of normalized observed and simulated streamflow or stage level for training performed with ANN1(8,1,1), ANN3(8,1,1), ANN1(8,5,1), ANN2(9,1,1) and ANN4(9,1,1), respectively. The results indicate that the training performed without streamflow (Fig. 6a) or stage level (Fig. 6b) data as an input to the ANN tend to overestimate the lower output values while some higher flood wave peaks are underestimated. These amplitude differences between observations and simulations are most likely related to the use of a single radar ZR relationship. In fact, the MP relationship tends to overestimate lower rainfall rates and underestimate higher ones. Moreover, it is more likely to have larger errors in radar rainfall estimation than in stage level or streamflow since the later represents an integrate measurement of the rainfall over the watershed plus other superficial hydrological processes. The increase in the number of nodes in the hidden-layer (e.g. Fig. 6c) slightly improves results, though not always, as discussed later in this section.

The addition of streamflow (Fig. 6d) or stage level (Fig. 6e) at time step t-Δt in the input layer improved the ANN performance by more than 40% variance-wise. This addition works as a memory of the watershed since it is the result of all hydrological processes. One might ask whether the radar data is need at all. With the radar data one can perform quantitative short-term rainfall forecast (Pereira Filho et al., 1999). It can be input to the hydrological model so increasing its lead-time. Furthermore, improving radar rainfall estimates was not a goal in this work. Since for these two configurations the ANN well simulated all flood waves,

one can infer that the quality of the data sets and the arbitrary subdivision of the watershed into isochrones limited the performance of the ANN. In this sense, notice that the stage level data yielded slightly better results because no additional uncertainty is introduced by the use a rating curve.

Normally, the first guess weights are selected in between specified limits. An appropriate selection can reduce the search time significantly while the opposite can lead to a premature saturation of the ANN. Initially, the weights are chosen randomly close to zero. Sometimes, it might lead to a local minimum. The ANN training process is completed when the solution yields an error less than an arbitrary small value.

Verification

A larger number of independent flood events depicted in Fig. 6 were utilized to verify the ANN training. Fig. 6 shows the simulated and observed time series of flood waves for the same options used in the training (Fig. 7). An inspection of Fig. 7 indicates that the ANN yielded fairly good flood wave simulations. The effect of errors and uncertainties are similar to the ones in the ANN training. The highest observed flood wave in between 300 and 350 h in Fig. 7a–c was not simulated at all most probably due to radar rainfall underestimation. Again, the inclusion of streamflow (Fig. 7d) or stage level (Fig. 7e) as an input variable greatly improved the simulations.

Table 2 shows variance coefficients of each of the training and verification experiments performed with the ANN shown in Figs. 6 and 7. More than 95% of the variance is explained with the inclusion of streamflow or stage level at the input-layer. On the other hand, variance coefficients of the ANN verification are 6 to 8% lower than the ones for training without the addition of either streamflow or stage level at the input-layer. As in the training, the use of stage level yields slightly better results than with streamflow.

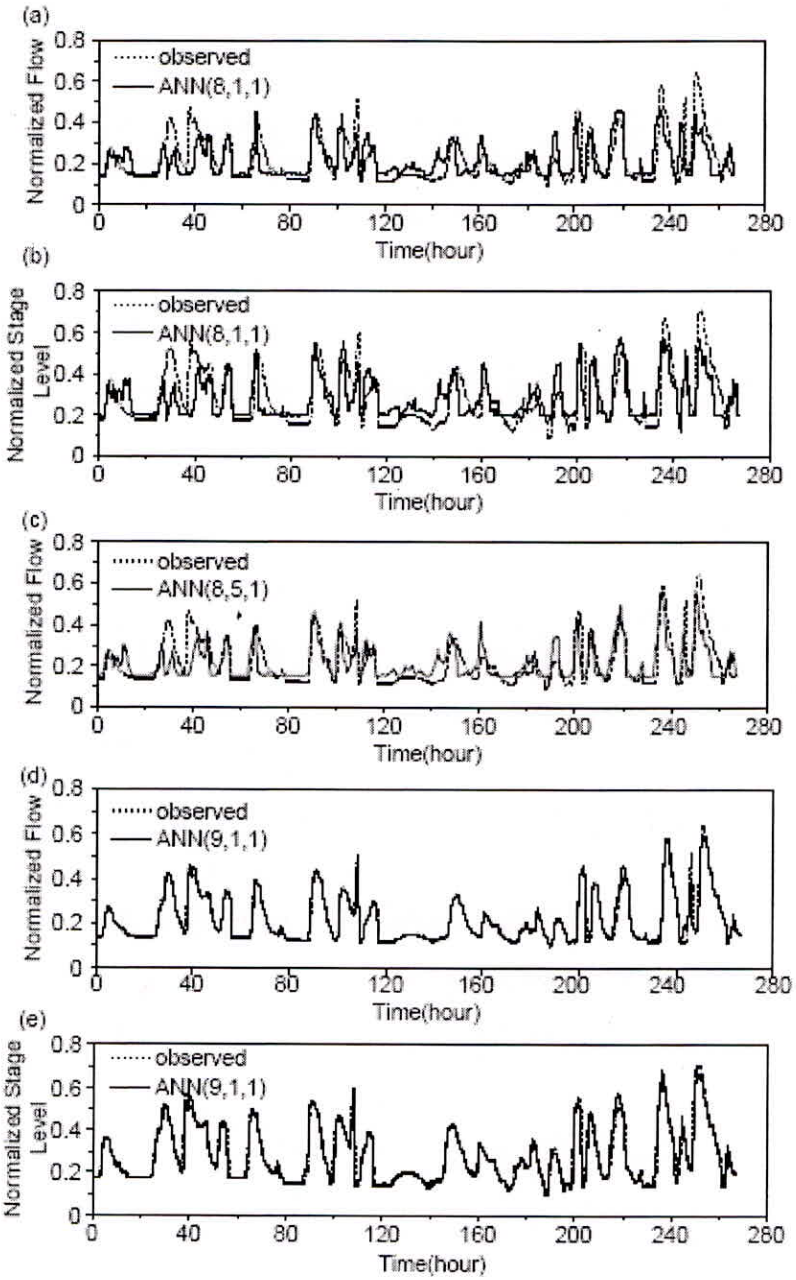

Fig. 6. Times series of normalized streamflow (a) and stage level (b) obtained with the ANN trained with options 1 and 3 (Table 1), respectively, as well as with option 1 for five nodes in the hidden-layer (c). Time series of normalized observations are indicated in each plot by the dashed lines.

Flood forecasting

The ANN was utilized to forecast flood waves with the configuration (9,1,1). Observed precipitation was used as forecasted together with stage level data. Since from the previous section it yielded slightly better results. Besides, it is straightforward to estimate flood levels in this manner. Observed stage level and precipitation at the initial time step were input to the ANN to forecast the stage level at the second time step. It was then input to the ANN with the observed precipitation as if it was forecasted to estimate the stage level at the third time step. This procedure was used up to the six time levels or up to 3 h in advance. Thirty-four independent events were utilized in the forecast (not shown). Stage level forecasts were compared to measurements.

Covariance coefficients (r^2) between observed and forecasted stage levels at 30-minute time steps are shown in Fig. 7. The covariance decreases with leadtime. Considering that the forecast has skill down to r^2=0.5, the ANN forecast is useful almost up to 1.5 h in advance. Of course, the skill is rapidly reduced by errors caused by data errors and hydrological uncertainties. Spatial and temporal variations in rainfall distribution are in general very significant. As mentioned before, radar rainfall estimation is subject to several sources of errors. Unfortunately, no independent rainfall data sets were available to perform a more in depth analysis of errors in radar rainfall estimation. Though, even with these limitations, the ANN performed better than current AR models used in the MASP (Pereira Filho and de Barros, 1998).

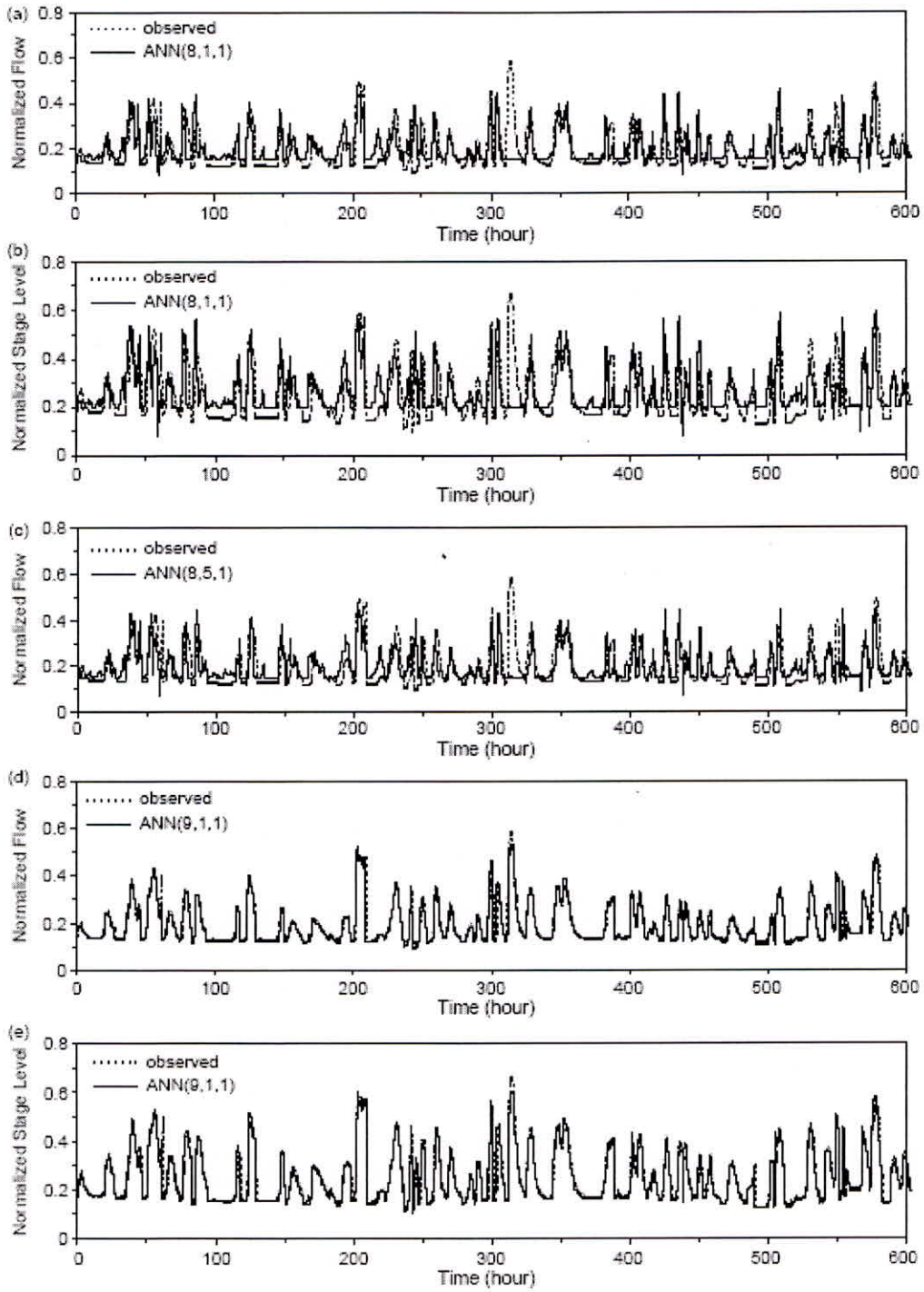

Fig 7 Observed and computed plot during validation period

Table 2
Variance coefficients between observed and ANN obtained stage level or streamflow for training (left) an verification (right) options specified in Table 1

Option	r^2 Training	r^2 Verification
ANN1 (8,1,1)	0.53	0.45
ANN3 (8,1,1)	0.54	0.49
ANN1 (8,5,1)	0.57	0.45
ANN2 (9,1,1)	0.95	0.95
ANN4 (9,1,1)	0.96	0.95

The 3D vector indicates from left to right the number of nodes in the input, hidden and output layers.

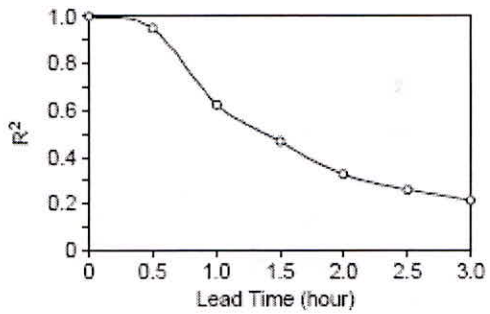

Fig. 8. Covariance coefficients between observed and the ANN4 forecasted stage level for 30, 60, 90, 120, 150 and 180 min lead time. Curve fitted by the least square method.

Conclusions

An three-layer feed forward ANN was applied to model a urban watershed using weather radar and telemetric streamflow data. Results indicate its usefulness for hydrologic simulation and forecast without explicitly resolving important surface hydrology processes. Even with better spatial and temporal data resolution the performance of the ANN was limited by data uncertainties and errors.

REFERENCES

Crespo, J.L., Mora, E., 1993. Drought estimation with neural networks. Advances in Engineering Software 18, 167–170.

Duan, Q., Sorooshian, S., Gupta, V.K., 1992. Effective and efficient global optimization for conceptual rainfall-runoff models. Water Resources Research 28 (4), 1015–1031.

Ferraz, M.I.F., Safadi, T., Lage, G., 1999. Use of series models in forecasting the series of monthly rainfall in the city of Lavras, state of Minas Gerais, Brazil. Revista Brasileira de Agrometeorologia 7 (2), 259–267 (in portuguese).

Fiering, M.B., Jackson, B.B., 1971. Synthetic streamflows. 2nd ed, American Geophysical Union, Washington (Water Resources Monograph Series).

Franchini, M., Pacciani, M., 1991. Comparative analysis of several conceptual rainfall-runoff models. Journal of Hydrology 122, 161–219.

French, M.N., Krajewski, W.F., Cuykendall, R.R., 1992. Rainfall forecasting in space and time using a neural network. Journal of Hydrology 137, 1–31.

Gupta, V.H., Hsu, K., Sorooshian, S., 1997. Superior training of artificial neural networks using weight-space partitioning. Proceed. IEEE (ICNN'97) 3, 1919–1923.

Haykin, S., 1994. Neural Network A—Comprehensive Foundation. Prentice-Hall International, Inc., New Jersey.

Hsu, K., Gupta, H.V., Sorooshian, S., 1995. Artificial neural network modeling of the rainfall-runoff process. Water Resources Research 31 (10), 2517–2530.

Hsu, K.L., Gupta, H.V., Sorooshian, S., 1996. A superior training strategy for three-layer feed forward artificial neural networks. Tucson, University of Arizona. (Technique report, HWR no. 96- 030, Department of Hydrology and Water Resources).

Karunanithi, N., Grenney, J.W., Whitley, D., Bovee, K., 1994. Neural networks for river flow prediction. Journal of Computing in Civil Engineering 8 (2), 1994.

Loke, E., 1995. Introduction to Artificial Neural Networks in Urban Hydrology, MATECH, Institute of Environmental Science and Engineering. Technical University of Denmark, Lyngby pp. 1–47, (report).

Luk, K.C., Ball, J.E., Sharma, 2000. A study of optimal model lag and spatial inputs to artificial neural network for rainfall forecasting. Journal of Hydrology 227, 56–65.

Marshall, J.S., Palmer, W.M.K., 1948. The distribution of raindrops with size. Journal of Meteorology 5, 165–166.

Morettin, P.A., 1982. Time series. In Short course of the National Congress of Applied Mathematics and Computational Sciences—SBMAC, 5th, Joa˜o Pessoa, Brazil (in Portuguese).

Morettin, P.A., Toloi, C.M.C., 1982. Models for the prediction of time series. Vol 1, Rio de Janeiro, IMPA-CNPq (in Portuguese). Operational Technical Center (CTO), (1997). Hydraulic and Hydrologic Study of the Tamanduatei´ watershed. Technical Report (in portuguese).

Oso´rio, F.S., Vieira, R., 1999. Tutorial of systems hybridos inteligentes. In: Cong. Da Soc. Bras. Computac¸a˜o-ENIA'99- Encontro Nacional de Intelige^ncia Artificial, Rio de Janeiro.

Pedrollo, O.C., 1999. Real time hydrological forecasting with effective rainfall estimated from the logistic function. RBRH— Revista Brasileira de Recursos Hı´dricos 4 (2), 19–30 (in portuguese).

Pereira Filho, A.J., 1999. Radar measurements of tropical summer convection: urban feedback on flash floods.29th radar conference, AMS, Montreal, Canada, July 1999. Paper 17.4, 939–940.

Pereira Filho, A.J, de Barros, M.T.L., 1998. Flood warning systemfor Megacities: a Brazilian Perspective. In: Wheater, H., Kirby, C.J. (Eds.), Hydrology in a Changing Environment,vol. III. Wiley, pp. 331–337 (ISBN 0-471-98680-6).

Pereira Filho, A.J., Crawford, K.C., 1999. Mesoscale precipitation fields: part I: statistical analysis and hydrologic response. Journal of Applied Meteorology 38 (1), 82–101.

Pereira Filho, A.J., Crawford, K.C., Hartzell, C., 1998. Improving WSR-88D hourly rainfall estimates. Weather and Forecasting 13 (4), 1016–1023.

Pereira Filho, A.J., Crawford, K.C., Stensrud, D.J., 1999. Mesoscale precipitation fields: part II: hydrometeorologic modeling. Journal of Applied Meteorology 38 (1), 101–125.

Pereira Filho, A.J., Negri, A., Nakayama, P.T., 2003. An intercomparison of gauge, radar and satellite rainfall in the tropics. First Workshop on Precipitation Measurements, IPWG/CGMS/WMO, Madrid, Spain. Proceedings. 275–283.

Ponce, V.M., 1989. Engineering Hydrology—Principles and Practices. Prentice Hall p. 640. Scalero, R.S., Tepedelenliogiu, N., 1992. A fast new algorithm for training feed forward neural networks. IEEE Trans. Signal Process. 40 (1), 202–210.

Shamseldin, A.Y., O'Connor, K.M., Liang, G.C., 1996. Methods for combining the outputs of different rainfall-runoff models. Journal of Hydrology 197, 203–229.

Shamseldin, A.Y., 1997. Application of a neural network technique to rainfall-runoff modelling. Journal of Hydrology 199, 272–294.

Smith, M., 1993. Neural Networks for Statistical Modeling. Van Nostrand Reinhold, New York. Singh, V.P., 1988. Hydrologic Systems-Rainfall-Runoff Modeling, vol. I. Prentice-Hall, Inc., New Jersey.