River Flow Forecast for Reservoir Management Using Neural Networks

B. Cannas[a], A. Fanni[a], M. Pintus[b] and G. M. Sechi[b]

(a) *Department of Electrical and Electronic Engineering, University of Cagliari, Cagliari, ITALY (cannas@diee.unica.it)*

(b) *Department of Land Engineering, University of Cagliari, Cagliari, ITALY*

Abstract: River flow forecasts are required to provide basic information for reservoir management in a multipurpose water system optimization framework. An accurate prediction of flow rates in tributary streams is crucial to optimize the management of water resources considering extended time horizons. In the paper different NN approaches will be analyzed to model the rainfall-runoff process when different time step durations have to be considered in reservoir management. Alternative neural models of the rainfall-runoff process are presented and discussed. Some numerical results are provided for runoff prediction in the Tirso basin at the S.Chiara section in Sardinia (Italy), using combinations of area and point-based measurements.

Keywords: Neural Networks; Rainfall-runoff process; Flow prediction

1. INTRODUCTION

Artificial Neural Networks (ANN) are widely accepted as a potentially useful way of modeling complex non-linear systems with a large amount of data, at times noisy. They are particularly useful in situations where the underlying physical process relationships are not fully understood, modeling complex problems both as substitutes for more conventional mathematical and statistical models, and in association with them.

In recent years ANN have increasingly been used for the prediction and forecasting of variables involved in hydrologic processes [Maier and Dandy, 2000]. Though the general tendency among users is to throw a problem blindly at an ANN in the hope that it will formulate an acceptable solution [Flood and Kartam, 1994], nevertheless it is essential to investigate different aspects of the ANN approach in order to improve prediction accuracy of the hydrologic processes, such as network architecture, modeling process, and efficiency estimation for model validation.

In this paper the ANN approach will be utilized to model the rainfall-runoff processes. The aim of this paper is to evaluate surface water resources dealing with water management problems when only information about basic input variables, i.e. rainfall and temperature, are available. Therefore, in the first phase of the investigation, the time step has been established in monthly periods. Further investigations have been made considering a daily time step.

Since the actual rainfall over an area is a termination stage of a number of different processes occurring on different scales, the derivation of area estimations over the basin from point observations remains one of the most difficult issues in hydrology [Berndtsson and Niemczynowicz, 1988]. The space and time scale of precipitation (as well as other hydrological variables) is related to the resolution of measurements and extensions of watersheds as well as to the type of underlying general hydrologic problem.

Using the monthly extended time-step, it is possible to disregard the rainfall kinematics of the single storm event, while the more classical problem of mean area precipitation estimation on an extended time period is being considered. Using the daily time step, the distances between point rainfall gauges and the runoff section have been considered to weight point rainfall in the ANN model.

The proposed ANN methodology can be used mainly to generate an extended hydrologic framework for water resource system planning and management problems referred to an extended time horizon. It is clear that using the monthly time step, the reconstructed hydrologic behavior of the runoff will be suitable for water resource studies where storage-yield sequences are frequently related to monthly periods. The daily time step could be adequate to deal with other proposals such as those related to flood flow problems only for large basins.

2. NEURAL NETWORKS FOR RAINFALL-RUNOFF PROCESSES

In the paper, traditional feedforward, multilayer perceptron (MLP) networks are used. Even though in recent works [Cannas et al., 2000; 2001] alternative architectures, such as locally recurrent neural networks (LRNN), have been proposed for rainfall-runoff processes, the results showed no significant improvement compared to the MLP approach, probably due to the limited time-dependency of the process. Thus, the considered ANN refers to the MLP even though different network architectures have been used.

In a previous paper [Lorrai and Sechi, 1995] the rainfall-runoff process was modeled using MLP for the Araxisi watershed in Sardinia (Italy). The examined ANN to model the hydrologic process were built considering mean area values of rainfall and temperature over the basin, and also taking into account point rainfall and temperature values. The authors showed that ANN gained in efficiency when point data inputs were considered. Moreover, the results obtained with ANN were significantly better than those reached using the conceptual model to simulate the rainfall-runoff process and multivariate AR models. In this paper we considered the Tirso basin at the S. Chiara section in Sardinia (Italy), which is very close to the Araxisi basin. This is a particularly interesting basin on account of its geographic configuration and water resource management. In fact a dam was built in this section in 1924 providing water resources for central-western Sardinia. Recently, the new "Cantoniera Tirso" dam was built a few kilometres down the river, creating a reservoir of a storage volume of 780 Mm^3. The basin area is 2,082.01 km^2, and is characterised by the availability of detailed data from several rainfall gauges. In this study we consider the data from 61 rainfall stations and 2 temperature stations.

The data of the training and validation sets were standardized using the following expression:

$$x_i = \frac{x_i - \min + eps}{\max - \min + 2 \cdot eps} \qquad (1)$$

where *min* and *max* are the minimum and the maximum values of the training data. The constant *eps* was introduced to avoid flat regions in the sigmoidal function and it was set equal to 0.1. Considering $y_{i(simulated)}$ as the model output in the i^{th} period and $y_{i(observed)}$ the observed data, the usual expression of

$$E = \sum_{i=1,N} (y_{i(observed)} - y_{i(simulated)})^2 \qquad (2)$$

is used to find the best ANN configuration to fit the observed data.

Nevertheless, the E index cannot easily be used to obtain a fitness criterion to be used to compare both model performances for the same watershed in different time stages and the reliability of model forecasting in different watersheds.

Moreover, in dealing with hydrological regimes with marked seasonal variations, model efficiency indexes were used following the definition given in [Lorrai and Sechi, 1995] and [Cannas et al., 2000].

In such a climate, efficiency should be defined taking into account the intrinsic variation of the estimated values from the general climate periodicity.
Considering $d=1, \ldots ,D$ periods and N_d values in the d^{th} period, we can define the square deviations of the observed runoff data within the period:

$$E_d = \sum_{i=1,N_d} (y_{i(observed)} - \overline{y}_d)^2 \qquad (3)$$

where $\overline{y}_d$ is the mean of the observed values in the d^{th} period.
Seasonal efficiency R_D can be written as:

$$R_D = \frac{(\sum_{d=1,D} E_d) - E}{\sum_{d=1,D} E_d} \qquad (4)$$

As E_d are known values from the training series, the criterion of maximizing R_D is equivalent to minimizing the residual square error E between observed and generated data.

Comparisons between models have been made using the efficiency expressions (4) to verify performances of ANN.

As mentioned above, the experiments have been divided considering point and averaged variables. Most procedures to evaluate the mean area precipitation $\overline{P}_i$ over the i-time period can be expressed as a linear combination of the observations $P_{i,j}$ at the gauges $j = 1,n$.

$$\overline{P_i} = \sum_{j=1,n} a_j \; P_{i,j} \qquad (5)$$

where the station weights a_j, $j = 1,n$, are non-negative, constantly varying the i-period, and sum to 1. In this application we refer to mean area rainfall evaluations as obtained in [Cao et al.,1983] modifying a procedure originally exposed by [Akin, 1971]. The method requires the determination of a reference triangle network of the gauge stations in the basin with vertices in the measurement locations. Using this approach, the baricentrical value $P_{i,k}$ of the k-element rainfall triangle plane takes place in *(5)* of the vertex $P_{i,j}$ observed value; the areas of the triangle reference network are the weights a_k in (5). Moreover, using the daily time step, we also compute a weighted precipitation for the current day using the inverse-distance method to evaluate a_k in (5).

3. APPLICATIONS AND RESULTS

Considering monthly time steps, preliminary experiments have shown that a best fit to the observed data may be obtained by introducing previous month runoff nodes in the input layer in order to represent the flows of the previous periods. This can effectively reproduce the preceding phase of the hydrographs, which is strongly related to groundwater storage and deep storage changes in the basin after rainy periods many time steps earlier. For the S.Chiara basin, it is shown that there is no special advantage in taking into account more than the data of six previous months.

As described in [Lorrai and Sechi, 1995], evapotranspiration losses in the basin can be modeled easily, considering the observed temperature values. As a matter of fact, even if approaches based on mass-transfer and energy balance methods are more detailed in estimating evapotranspiration, for the extended period and practical purpose when not all the data are available, empirical temperature-based equations, such as the Blaney-Criddle and Thornthwaite equations, can be used [Bras, 1990]. As losses in previous months are globally represented by the behavior of rainfall and runoff, the model will incorporate only the present month temperature.

In order to model the rainfall-runoff process related to Mediterranean climate watersheds, it will be important to take into account a month dependent general climate variable. This variable could easily be associated with analytic periodic functions with periods equal to one year, as in [Abraharth, 1999]). Giving better results, in this work we refer to different cyclic independent variables: a variable equal to the standardized monthly averaged runoff values, and the sine and cosine component of the annual monthly count.

Sixty-nine years of monthly flow data for the S.Chiara basin are available, i.e. from 1924 to 1992 for a total of 828 data values. The data set was split into two parts, the first 49 years (588 monthly values) were used as a training set, while the remaining 20 years (240 monthly values) were used as a validation set. The analyses were carried out splitting the 20 years of the validation into 10 years of cross-validation and 10 years of final validation.

In modeling the monthly rainfall-runoff process, we refer to case A, and the following alternatives have been assessed:

1. A data preprocessor has been used to calculate the mean area rainfall for the entire basin. As described below, the experiments have been carried out considering only rainfall in the first case, and adding respectively runoff, temperature, and climate information in the input layer.
2. Point rainfall values have been used, as gains in efficiency have been pointed out in previous studies. These cases refer to the inclusion of raw point rainfall, which was observed in the MLP input layer in the present and in previous months.
3. As can be understood easily, strong inter-correlations characterize rainfall values in the same period measured at different stations in the basin. The principal components have been preprocessed to reduce the number of independent rainfall variables. The component numbers will be different as the number of previous months varies. The accepted variance error has been fixed at 2%.
4. An MLP with a lumped structure has been considered to obtain an averaged rainfall value at the same t-period. Point rainfall nodes in the input layer have been connected only to the t-node in the first hidden layer corresponding to the same observed time period. The network contains two hidden layers, and the node number in each layer equals the period considered.

The four model alternatives were split into the following subclasses, according to the types of inputs used:

a) only rainfall data;
b) rainfall and runoff data in previous periods;
c) rainfall, runoff, and climate cyclic variable of the present month, defined as standardized monthly averaged runoff;

d) rainfall, runoff, climate variable, and temperature of the present month.

In particular, cases **A1***x* refer to the first class of MLP models using an average data preprocessor, and the results for each subclass are given in Table 1.

It can be seen from the Table that, as expected, we obtain better results when we consider the runoff of the previous months and the climate monthly cycling variable.

Table 1. MLP results for the rainfall-runoff process using mean area rainfall

Model	MLP layers	R_D (20 years)	R_D (1st 10years)	R_D (2nd10 years)
A1a	7:7:1	0.83	0.78	0.86
A1b	13:7:1	0.82	0.76	0.86
A1c	14:7:1	0.86	0.84	0.88
A1d	16:7:1	0.85	0.83	0.87

At the same time, temperature does not have a significant role in the model. This can be related to the fact that the model can understand the losses from the rainfall-runoff behavior of the previous months and from the present value of the climate variable.

The second class of MLP models, considering point value rainfall at 61 gauging stations, was therefore limited to subclasses **b** and **c**. The results are shown in the Table 2.

Table 2. MLP results for the rainfall-runoff process using point rainfall

Model	MLP layers	R_D (20 years)	R_D (1st10yrs)	R_D (2nd10yrs)
A2b	247:3:1	0.38	0.40	0.37
A2c	248:3:1	0.41	0.43	0.39

As pointed out in the Table, because of the large number of distributed inputs, and consequently the reduced ratio between learning data and number of weights, a general estimate cannot be obtained, not even by reducing the number of hidden nodes.

It seems evident therefore that we need to use pre-processors that should allow to reduce or appropriately organise information to the network. Two different strategies can be used. One is to pre-process the input data to reduce its dimensionality. The second is to leave the data compression task to the network itself.

As previously stated, class **A3** of the MLP network refers to a Principal Component Analysis (PCA) preprocessor, while class **A4** refers to an input layer organized with selected connections to a hidden layer with a lumped structure. The latter model needs two hidden layers. The results are presented in the following table:

Table 3. MLP results for the rainfall-runoff process using the PCA preprocessor and lumped structured ANN:

Model	MLP layers	R_D (20 years)	R_D (1st 10years)	R_D (2nd10 years)
A3b	7:7:1	0.46	0.58	0.32
A4b	247:7:3:1	0.55	0.59	0.51

From the above experiments it seems clear that the mean area rainfall preprocessors give the best results. Indeed they reach a better generalization of the process due to the smaller number of weights in the model. Furthermore they manage the information on the topology of the measurement system, which other approaches do not consider. Figure 1 shows the observed and generated runoff using model **A1c**.

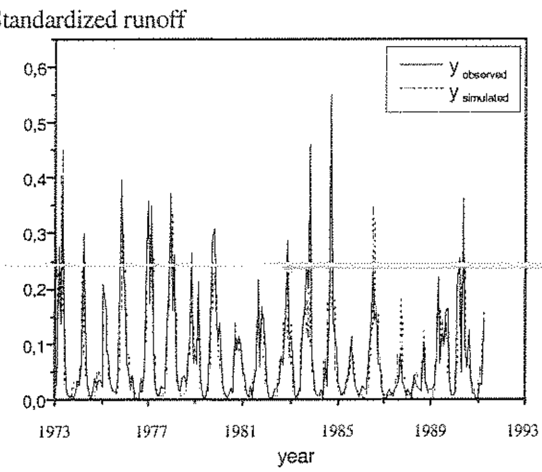

Figure 1. The standardized observed and generated runoff (Model **A1c**).

The **A1c** model was used to generate a synthetic runoff series using feedback connections between runoff generations in lagged periods. Experiments have been carried out using a recurrent MLP network architecture. In the testing phase no historical information on runoff was used during the process prediction. To characterize this procedure, we refer to model subclass **e** as runoff generated by the network is used as an input in the ANN. The results are shown in Table 4.

Table 4. Results for runoff generation using a recurrent MLP network:

Model	MLP layers	R_D (20 years)	R_D (1st10yrs)	R_D (2nd10yrs)
A1e	8:4:1	0.81	0.81	0.82

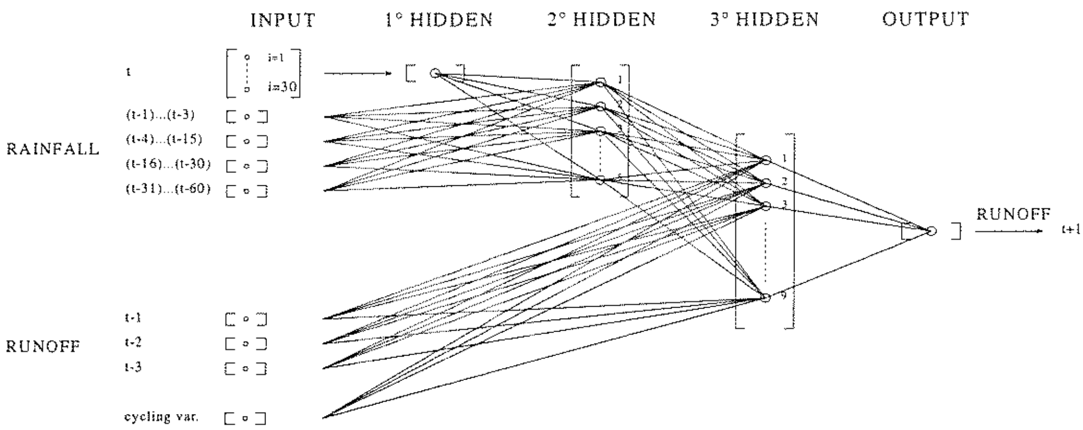

Figure 2. Lumped MLP structure.

It can be noted from the above tables that the monthly model performances slightly deteriorate on removing some input information. In particular when observed runoff information is removed, the performance variations are negligible. These results suggest that this approach could provide a useful tool for synthetic stream-flow series generation, which is particularly helpful when the historical series are incomplete, as is often the case in Sardinian basins. The results on series generated by models **A1e** show that this model can be recommended for long term time series generations.

Considering daily time steps, ten years of daily flow data have been examined. The data set was split into two parts: the first 4 years were used for validation, the last 6 years were used for training.

In modeling the daily rainfall-runoff process we refer to case **B**, and the same 4 alternatives (codes from **1** to **4**) and 6 subclasses (codes from **a** to **d**) previously considered for the monthly models. Moreover, in case **B1** to model the daily time steps, we compute the weighted precipitation for the current day using the inverse-distance method.

Distances have been evaluated between rainfall gauges and the runoff measurement station.

After a few preliminary experiments to define the number of previous periods in alternatives and subclasses, the considered potential ANN model inputs include rainfall from the previous 60 days and runoff from the previous 3 days:

P_1 : present day rainfall;
P_2 : cumulated rainfall from *(t-1)* to *(t-3)* previous days;
P_3: cumulated rainfall from *(t-4)* to *(t-15)* previous days;
P_4 cumulated rainfall from *(t-16)* to *(t-30)* previous days;
P_5 cumulated rainfall from *(t-31)* to *(t-60)* previous days;

In the following table we summarize the most significant results obtained using the preprocessor for mean area runoff estimation.

Table 5. MLP results for the rainfall-runoff process using mean area rainfall (daily model).

Model	MLP layers	R_D (10years)	R_D (4 years testing)
B1a	5:5:1	0.53	0.48
B1b	8:8:1	0.71	0.51
B1c	9:9:1	0.77	0.78

It can be seen that significantly better results can be obtained using information about rainfall and runoff in previous time steps as well as the climate cyclic variable.

As pointed out for the monthly models, because of the large number of distributed inputs, and consequently the reduced ratio between learning data and number of weights, good results cannot be obtained using the point rainfall values, not even if the MLP network referring to a Principal Component Analysis preprocessor is reduced.

Far more interesting are the results obtained using models **B4** and referring to an input layer organized with selected connections to hidden layers with a lumped structure. Model **B4c** needs three hidden layers and has been schematized in Figure 2.

The results obtained with the lumped structure are summarized in Table 6.

Table 6. MLP results for the rainfall-runoff process using the lumped structured ANN.

Model	MLP layers	R_D (10years)	R_D (4 years testing)
B4b	37:1:5:8:1	0.73	0.67
B4c	38:1:5:9:1	0.73	0.79

As previously pointed out, the MLP with a lumped structure gives a balanced average of rainfall at the current *t*-day-period.

4. CONCLUSIONS

The neural models applied to the rainfall-runoff transformation problem provide a useful tool for the prediction of runoff to generate an extended hydrologic framework for water resource system planning and management problems referred to monthly data. In this case, it has been shown in the paper that even when only information about basic input variables is available, the performance of ANN deteriorates only slightly. Referring to daily data, preliminary results show the necessity to utilize other alternative preprocessors than simple mean-area evaluation to treat rainfall data adequately. Results obtained using lumped structured ANN for daily data seem promising in this direction. When shorter time-steps have to be managed in order to face flood forecast in extended watersheds, more sophisticated preprocessing methods and network structures have to be developed in order to reduce the dimensions of the data. Nevertheless, ANN can be used successfully in many practical engineering applications where the main aim would be to make accurate hydrologic predictions, in cases where a physically-based description of the rainfall-runoff process is not possible.

5. REFERENCES

Abrahart R.J., L. See and P.E. Kneale, Applying saliency analysis to neural network rainfall-runoff modeling, GeoComputation 3, 1999.

Akin J.E., Calculation of mean annual depth of precipitation, *Journal of Hydrology*, 12, 1971.

Berndtsson R. and J. Niemczynowicz, Spatial and temporal scales in rainfall analysis - Some aspects and future perspectives, *Journal of Hydrology*, 100, 293-313,1988.

Bras R.L., Hydrology, *An introduction to hydrologic Science*, Addison-Wesley, 1990.

Cannas B., A. Carboni, A. Fanni and G. M. Sechi, Dynamic neural networks for water flow forecasting, Sixth International Conference on Engineering Applications of Neural Networks EANN 2000, Kingston Upon Thames, UK, 17-19 July 2000.

Cannas B., A. Fanni , M. Pintus and G. M. Sechi, Alternative Neural Networks Models for the Rainfall-Runoff Process, Seventh International Conference on Engineering Applications of Neural Networks EANN 2001, Cagliari, (Italy), 16-18 July 2001.

Cao C., E. Piga, M. Salis, G. M. Sechi and R. Silvano, Regional Resources Planning and Management: The Water Plan of Sardinia; Preliminary Report, Int. Ass. Hydr. Res. Conference, Moscow, 5-9 Sept. 1983.

Flood I.and N. Kartam, Neural networks in civil engineering. I: Principles and understanding, *Journal of Computing in Civil Engineering*, 8 (2), 131-148, 1994.

Lorrai M. and G. M. Sechi, Neural nets for modelling rainfall-runoff transformations, *Water Resources Management*, 9, 299-313, 1995.

Maier H.R. and G.D. Dandy, Neural Networks for the prediction and forecasting of water resources variables: a review of modeling issues and applications. *Environmental Modeling and Software*, 15, 101-124, 2000.