DE GRUYTER
OPEN

Mathematical Modelling in Civil Engineering
Vol. 13-No. 3: 10-20 - 2017
Doi:10.1515/mmce-2017-0008

FLASH-FLOOD MODELLING WITH ARTIFICIAL NEURAL NETWORKS USING RADAR RAINFALL ESTIMATES

Cristian DINU - Department of Hydrotechnic Engineering Technical University of Civil Engineering Bucharest, Romania, e-mail: krisitan_dinu@yahoo.com

Radu DROBOT - Department of Hydrotechnic Engineering Technical University of Civil Engineering Bucharest, Romania, e-mail: drobot@utcb.ro

Claudiu PRICOP - Water Basin Administration Prut-Bârlad, Romania, email:claudiu.pricop@dap.ro; water.ro@yahoo.com

Tudor Viorel BLIDARU - Water Basin Administration Prut-Bârlad, Romania

Abstract: The use of artificial neural networks (ANNs) in modelling the hydrological processes has become a common approach in the last two decades, among side the traditional methods. In regard to the rainfall-runoff modelling, in both traditional and ANN models the use of ground rainfall measurements is prevalent, which can be challenging in areas with low rain gauging station density, especially in catchments where strong focused rainfall can generate flash-floods. The weather radar technology can prove to be a solution for such areas by providing rain estimates with good time and space resolution. This paper presents a comparison between different ANN setups using as input both ground and radar observations for modelling the rainfall-runoff process for Bahluet catchment, with focus on a flash-flood observed in the catchment.

Keywords: Artificial neural network, flash-floods, radar rainfall estimates, rainfall-runoff modelling.

1. Introduction

Radar technology is a central instrument in meteorological forecasting, generating important information regarding the intensity, movement and future locations of precipitations. Combined with a ground-based monitoring network, it can provide enhanced short-term forecast data. Beside meteorological applications, the radar observations are of great significance in hydrological field due to the good temporal and spatial resolution. Rainfall-runoff modelling tools can benefit from spatial coverage given by the radar products in areas where the ground-based monitoring network does not provide a good representation of the rainfall over the catchments. Moreover, if the area is prone to flash-floods, the chances are that a low density network will not provide good qualitative and quantitative rainfall data for modelling instruments and in this case radar products can represent a viable solution.

Amongst the most important instruments as practical solution for discharge and flood estimation are the rainfall-runoff simulation models. Using different assumptions and approximations of the hydrological processes inside a catchment there were developed numerous rainfall-runoff simulation models, designed to estimate the discharge rates in a catchment. In practice, conceptual and physically based models are the most used tools in modelling the runoff generation, making them the traditional approach.

During the last 20 years, new instruments developed in machine learning found their way into hydrology. Due to their flexible structures and their capabilities to approximate nonlinear relations, ANNs were among the first such tools used for modelling the hydrological processes. An excellent introduction and review of ANN theories and applications in hydrology is published by the ASCE Task Committee [1], [2]. ANNs have been tested on multiple occasions for rainfall-runoff modelling, examples of studies with successful results are given by [3], [4], [5], [6].

The study presented in this paper focuses on the development of ANN models for simulating the rainfall-runoff process using an event-based approach for the upstream part of Bahluet catchment with the closing section at Targu Frumos gauging station. Due to the fact that ANNs are capable of mapping complex input-output relations, the main objective of the study was the development of an ANN able to simulate a flash-flood event measured at the Targu Frumos using three types of input data. The input vectors presented to the networks are given by ground rainfall data, radar rainfall estimates and precipitated volumes calculated from radar data, with the purpose of identifying which of data sets leads to better results. In case of ground and radar precipitation the intention was to identify if a better rainfall representation over the catchment produces superior results when use in data-driven models such as ANN, or acceptable results are obtained based only on ground measurements. In case of precipitated volumes, the purpose was to identify if the networks generate improved results if provided with information about the modelled area, considering that precipitated volumes contain data about the catchment area.

2. Method

2.1 Artificial neural network model

Artificial neural networks (ANN) are computational models developed to emulate the processes found in the biological brain. ANNs are composed by a set of interconnected simple processing elements, also referred as nodes or neurons, operating in parallel. Similar to the biological counterpart, the ANNs abilities to approximate the relations inside datasets is determined by the connections between the neurons. ANNs can be considered non-linear regression models which, following a training process, can perform a mapping of input-output relationship [7]. ANNs have been applied in various science fields with numerous real-world applications such as pattern recognition, classification, time series prediction or rainfall-runoff modelling.

There are different ways of classifying the ANNs, depending either on neurons arrangement or on the direction of information propagation within networks. Usually the neurons are arranged in layers and based on their number the ANNs can be classified as single layers, bilayers and multilayers, the latter being the most used type of network. For multilayer ANN, based on the layer position they are usually referred as input layer, responsible for presenting the inputs to the network, hidden layer/layers responsible for the processing on information, and the output layer generating the results of the network.

Based on the signal propagation direction inside the network, ANNs can be broadly classified as static or dynamic networks. The static ANN, usually known as feed-forward networks, are unidirectional in regard with signal propagation, and the information passes from input to output layer. The output of a neuron in a layer depends only on the current inputs it receives from a previous layer. In case of dynamic ANNs, the information usually flows through the network in both directions, from input to output layer and vice versa [1] by using feedback loops, allowing previous network outputs as current inputs. These types of networks are commonly referred as recurrent networks and in such cases, the output depends on the current input as well as on previous inputs or/and outputs.

One of the most known ANN model is the multi-layer percepton network (MLP), applied successfully to solve a wide range of difficult problems. They are feed-forward networks, composed typically by 3 layers (Fig. 1).

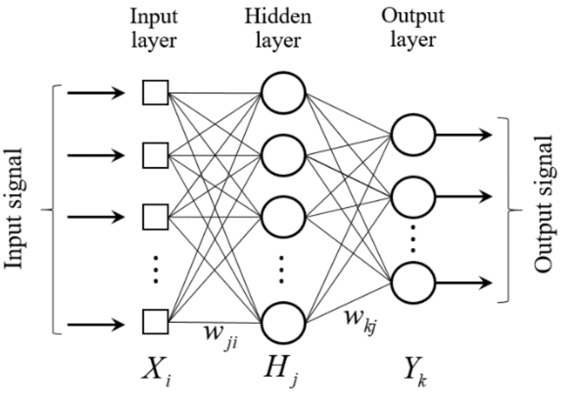

Fig. 1 - Three-layer ANN schematization

The mathematical form of the output of a MLP network is given by the expression:

$$y_k = f_2(w_{kj} f_1(w_{ji} x_i + b_j) + b_k) \tag{1}$$

where y_k is the output value of node *k* from output layer, x_i is the input value of neuron *i* of the input layer, w_{ji} is the weights assigned to the connection of node *i* of the input layer to the node *j* of the output layer and similarly w_{kj} is the weights assigned to the connection between the neuron *j* of the hidden layer to the node *k* of the output layer. The terms b_j and b_k are called bias or thresholds associated with *j* and *k* nodes. If they are used, the node needs to exceed this value before it can be activated and in some situations it is neglected. The functions f_1 and f_2 are called activations functions (in this case for hidden and output layer) with the role of determining the output of a neuron when it receives the input signal [1]. The most frequently used function is the logistic (sigmoid) function or transformation of this such as hyperbolic tangent or bipolar sigmoid. The use of logistic activation functions allows the network to map out the nonlinear relationships between input and output data, an important feature since the hydrological processes such as rainfall-runoff are known to be highly nonlinear [3].

In order to generate accurate results with ANNs, they need to undergo a training process, which can be described, in general lines, as the process of searching the optimal values for the ANN internal parameters (weights and biases). The most popular training algorithm is the back-propagation algorithm which employs the gradient descent technique for minimizing the objective functions. Due to the slow convergence of backpropagation algorithm, there are available alternatives such as quasi-Newton algorithm, conjugate gradient or Levenberg-Marquardt (L-M) algorithm and detailed mathematical aspects of these algorithms can be found in [8].

The main issue of static ANN is the limitation when it comes to approximate time dependent functions or processes (e.g. rainfall-runoff process). When ANNs are employed to approximate processes in which both spatial and temporal variability are equally important, there is need for dynamic estimation procedures. In such cases the time dimension needs to be implemented into ANNs structure, a process that is also referred to as providing memory to the network. One way of time implementation in ANNs can be achieved at external level, where the time dimension is transformed in spatial dimension via delay elements grouped in so called tapped delay lines (TDL), resulting in time delay networks (TDNN). Another manner is to implement the time dimension at internal level, with implicit or explicit representations, in this case resulting in partial recurrent networks (context networks) and fully recurrent networks. Time delay ANNs and partial recurrent ANNs can be trained using static algorithms like backpropagation, with some advantages for the partial recurrent network due to the fact that it uses previous outputs as current inputs, providing better solutions than TDNN [1]. The fully recurrent networks are more difficult to implement due to the fact that the training algorithms needs to be modified in order to include the time dimension.

For the study presented here the nonlinear autoregressive network with external inputs (NARX) was used, a partial recurrent dynamic network with good results in time series prediction [9]. NARX network can be seen as a combination between TDNN and context networks and is defined by the expression:

$$y(t+1) = f\left(y(t), y(t-1), ..., y\left(t-n_y\right), u(t), u(t-1), ..., u\left(t-n_u\right)\right) \qquad (2)$$

where *y(t)* and *u(t)* represents the output and input to the network at discrete time steps *t*, n_y and n_u represents the output and input memory orders. The future value of the dependent variable *y(t+1)* is regressed on the current and past values of the output signal along with current and past value of the independent variable.

According to the network diagram (fig. 2) the general configuration of NARX model is essentially a feedforward network with one hidden layer and an output layer, equipped with TDLs and a feedback loop from the output to the inputs.

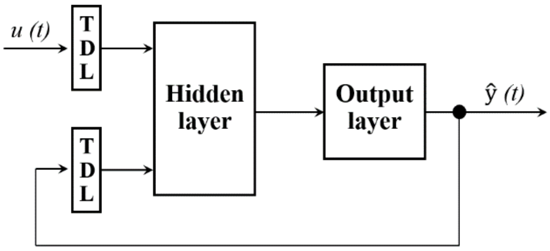

Fig. 2 - NARX architecture schematization (after Hagan et al. 2014)

The advantage of NARX is given by the fact that it can be trained using standard static training algorithms, achieved due to the particularity of this network that can be implemented using two configurations, known as series-parallel architecture and parallel architecture. Since the true output (measurements) of the system is available in the training phase, the network can be implemented with a series-parallel architecture (fig. 3-a), which is essentially a feedforward networks with delayed inputs. After the training is done, the parallel architecture can be used to simulate the behaviour of the nonlinear dynamic system.

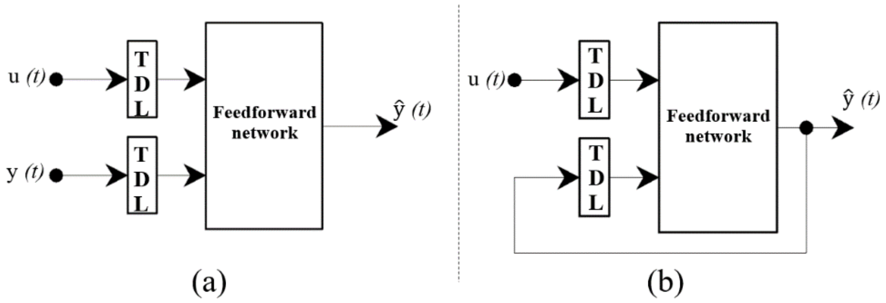

Fig. 3 - Series-parallel (a) and parallel (b) NARX configurations (after Hagan et al., 2014)

Due to its implementation mode, the NARX network can be used as a time series predictor in two ways [9]. The series-parallel architecture makes the network to function as a one step ahead predictor, generating the next value of the dependent variable based on previous observation. On the other hand, the parallel architecture (fig. 3-b), makes the network to work as a multistep ahead predictor based on the output memory order (number of time steps regressed).

For this study, the L-M algorithm [10]-[11] was used for network training. This is a fast and reliable algorithm and it was developed as a variation of Newton’s method.

The Hessian matrix is approximated as:

$$\mathbf{H} = \mathbf{J}^T \mathbf{J} \qquad (3)$$

where J is the Jacobian matrix containing the ANN first derivatives of errors with respect to weights and thresholds. With expression 3, the correction of the network weights is:

$$\Delta \mathbf{w} = -\left[\mathbf{J}^T\mathbf{J} + \mu\mathbf{I}\right]^{-1}\mathbf{J}^T\mathbf{e} \tag{4}$$

where μ is a variable scalar that controls the learning process and e is a vector of networks error. Based on the value of μ, L-M varies between gradient descent method and Gauss-Newton method. For small values of μ, the algorithm approaches Gauss-Newton method, using the Hessian matrix approximation. For large values of μ, the algorithm becomes a gradient descent method. Due to faster convergence times and higher accuracy of the second order methods, the aim is to shift the algorithm towards the Gauss-Newton method as quickly as possible. Detailed mathematical aspects of L-M algorithm implementation for neural networks are presented in [9].

2.2 Study area

The study area is located in Bahluet basin, a sub-catchment of Bahlui watershed, situated in the north-eastern part of Romania in Iasi County. The Bahluet catchment has an area of roughly 551 km^2, the length of Bahluet River is 41 km, the average elevation is around 150 m.a.s.l. and the average slope of the basin is 7.4 ‰. The study was focused on the upstream part of Bahluet catchment with the outlet at Targu Frumos gauge station (fig. 4). This part of the basin has an area of 66 km^2 and a river length around 21 km. Based on historical measurements, the average multiannual discharge is 0.15m^3/s and the average multiannual precipitation has a value of 550 mm/year.

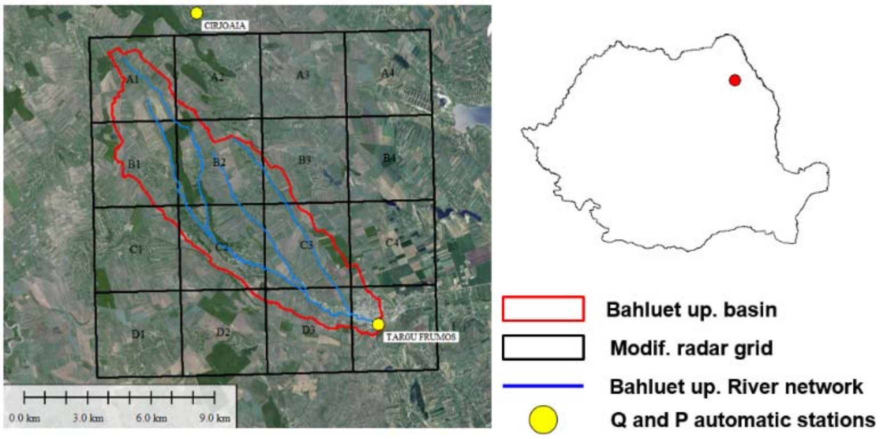

Fig. 4 - Bahluet catchment map

The hydrological regime of Bahluet river is characterised by low flows all year round, with increases in discharges in early spring due to snow melting and in last period of autumn due to rainfall. Usually, in summer months (July and August) the watershed is affected by scattered short strong rain events which represent the dominant process in flood generation. Due to the fast response and short lag times, this sub catchment can be catalogued as a flashy.

The fast response of the catchment, correlated with the fact that there is no other station upstream of Targu Frumos, the possibilities for early flood warning are limited, which makes this area to be of high interest regarding this issue.

2.3. Available data

The primary collected data consist of discharge observation, rain measurements from ground stations and radar rainfall estimates. Based on the event-based modelling, the data sets were available for 2 events that occurred in 2014. First flood event was measured in 21.05.2014 and the second one was registered in 21-22.07.2014. All datasets were homogenised in regard to time resolution at 10 minutes per time step.

2.3.1 Rain gauge and runoff data

The monitoring network density made of automatic stations for Bahlui catchment is of fairly medium size compared to the density values suggested in [12], with 14 stations covering roughly 2000 km^2. The runoff measurements were taken from Targu Frumos automatic situated at the closing section of the studied catchment. For precipitation measurements, due to the position of the study area inside Bahlui basin, only 2 station are representative of Bahluet sub-catchment, namely Targu Frumos and Carjoaia. The former is located at the same place as the gauge station for runoff observations and the latter is situated just outside of the basin, close to the most upstream part of sub-catchment. The automatic rain gauges use the tipping bucket principle, having a 0.1 mm resolution, and a temporal resolution of 10 minute, the same as the discharge observations. The automatic gauging station were implemented inside DESWAT project, and are officially controlled and verified by ABA Prut-Barlad (regional water basin administration).

In the graphs below the 2 flood hydrographs with the corresponding rainfall from ground measurements are presented.

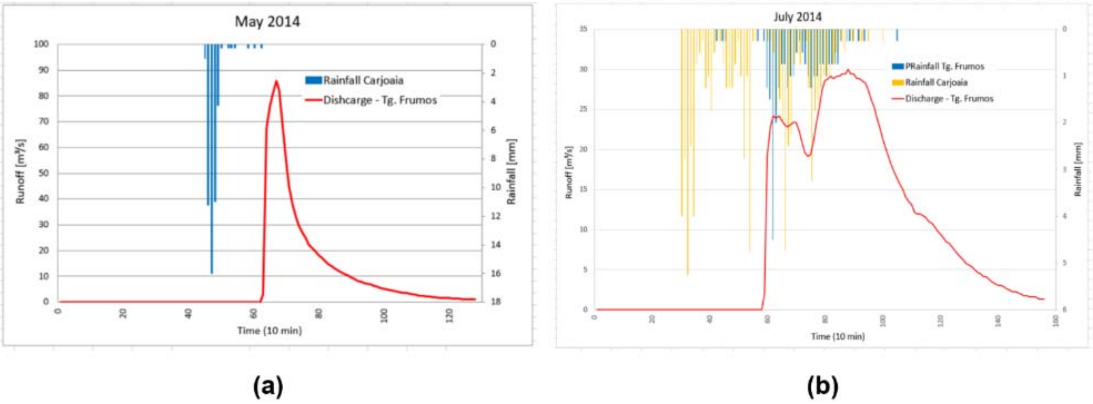

Fig. 5 - Registered flood waves at Tg. Frumos

The event observed in May 2014 (fig. 5-a) has the characteristics of a flash-flood and is of high interest for this study. The flood event was generated by a strong precipitation over a short period of time, with the nucleus of the rain in the most upstream part of the basin. The ground measurements were available only for Carjoaia station and no rainfall was measured at Targu Frumos station for this event. The peak water level observed at Targu Frumos was 336 cm and it exceeded the maximum flood threshold defined for this section (330 cm). Also, according regional water basin administration there were reported damages in the area around Targu Frumos for this event.

The second event, measured in July 2014 (fig. 5-b), did not had the same order of magnitude as the previous one but, according to the flooding levels determined for Targu Frumos station, the measured water levels (202 cm) exceed the first flood warning level (200 cm). In this case, the rainfall was distributed over the whole catchment area, with rain observations from both ground stations.

2.3.2 Radar data

The radar rainfall estimates for the studied area are obtained from the Doppler weather radar station located in Barnova, Iasi. This station uses high-resolution S-band weather radar (operating on a 6-20 cm wavelength), with a range of 230km, situated between 46 - 65 km from the catchment. The tower height is 38m, taking measurements every 6 minutes with a spatial resolution of the volume element of 1 km^3.

The measuring unit of the radar is the reflectivity factor Z which is used to estimate the rain rate using the empirical relationship:

$$Z = aR^b \qquad (5)$$

where Z is the reflectivity factor in mm^{6}m^3, R is the rain rate in mm/h and the coefficients a, b are empirical parameters. For this weather radar station the relation is not fixed and the empirical parameters change their values depending on the season. In this case, for summer the values are $a = 300$, $b = 1.4$ and for winter $a = 200$ and $b = 1.6$.

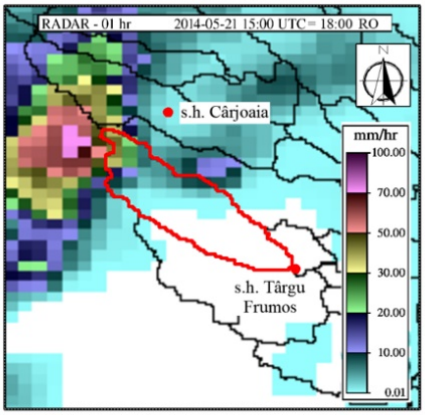

Fig. 6 - Example of radar precipitation (May 2014 event)

The radar data for the events mentioned before were collected from ABA Prut-Barlad and they are a product of SIMIN system (National Integrated Meteorological System). Radar precipitation estimates are given as accumulated values over one hour and an example of radar product for May 2014 event is presented in fig. 6.

2.4 Data processing and proposed data structure

Part of pre-processing process was the extraction of precipitation rates values from radar data using GIS tools. The radar files provide the accumulated precipitation estimates on a 1km x 1km grid. Due to the large number of cells covering the studied area, the precipitation data were extracted using a 4km x 4km grid in order to obtain a more manageable input vector for ANN model, without losing important information about the precipitation data. As a result, there were obtained 12 rainfall time series with 1hour time step corresponding to the cells that intersected the catchment, for each of the two events. Fig. 7 shows the extraction grid superimposed over the catchment and examples of rainfall hyetographs obtained for cells A1 and A2 corresponding to May 2014 event.

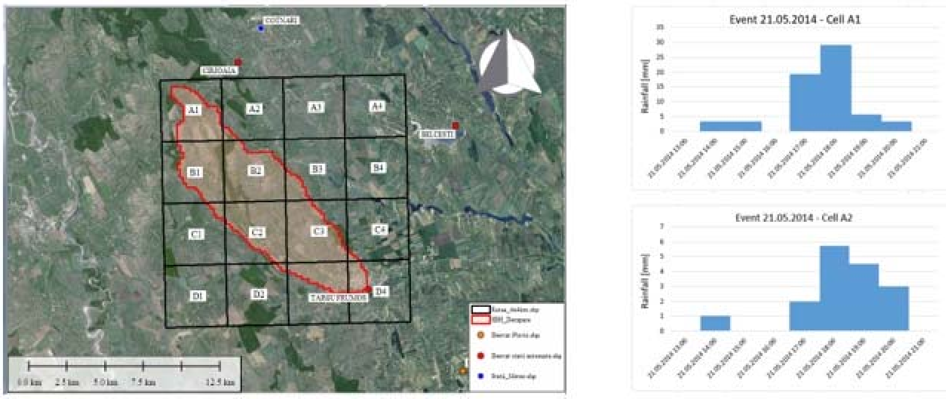

Fig. 7 -Example of the grid used to extract rainfall data and hyetographs resulted for A1 and A2 grid cells

The precipitation volumes were calculated based on the rainfall data extracted from radar files considering the areas determined by intersecting the 4km x 4km grid with the catchment surface. As in the previous situation, for each of the two events were obtained 12 volume time series with 1-hour time resolution.

In order to homogenise the datasets, the information determined from radar files is processed in regard to time resolution, generating series with 10 minute time resolution, same as automatic sensors data. In this sense, the radar data were considered to have a uniform time distribution during each time interval of 60 minutes.

Both ground and radar data used as inputs vector for NARX model were arranged in matrix form. The ground rainfall observations were structured as a matrix of 2 columns (for each station) by *n* rows, representing the number of time steps for each event. The radar data (rainfall and volumes) were structured as a matrix of 12 columns (corresponding to the grid cell intersecting the catchment) by the same number of time steps as the ground rainfall observations. The stations or grid cells with no precipitation observed were included in the input vectors with zero value.

3. Results

As mentioned before, one of the objectives was to identify the architecture of the neural network that will best simulate the catchment behaviour. For a NARX model, this process is achieved by determining the number of neurons in the hidden layer. A trial and error approach was adopted, testing three networks architecture, with 10, 15 and 20 neurons on the hidden layer. The activation function for the hidden layer was hyperbolic tangent, while for the output layer an identity function was used. For the training process the Levenberg-Marquardt algorithm was employed.

NARX model needs also the definition for delay units, applied to the inputs in order to use antecedent rainfall and runoff data for simulation. The number of delay units was determined using cross-correlation from signal processing and was applied between runoff and rainfall. Due to the fast response of the catchment during May 2014 event, the lag time was determined for the dataset describing this event. The average delay found after processing the signals was of 60 minutes. Based on this, for the NARX models 6 time steps or 6 delay units for all three architectures were used.

For better generalisation the input data were divided in three sets, training, validation and testing sets. The samples for the sets were randomly selected form the training data, using a 0,7 weight for the test set and 0,15 for each of testing and validation sets.

The study was conducted using Matlab software package.

3.1 ANN training

Using an event-based approach, for the training phase the July 2014 event was used, due to better spatial representation of the rainfall over the analyzed catchment. The model performances were evaluated using graphical methods (simulated versus measured hydrographs) together with root mean square error (RMSE) index and coefficient of efficiency (CE) [13]. Regarding the efficiency coefficient, values above 0.9 for CE suggests a very good model performance. If it is in 0.8-0.9 interval indicates a good performance. Values less than 0.8 signify a model with unsatisfactory performances [14].

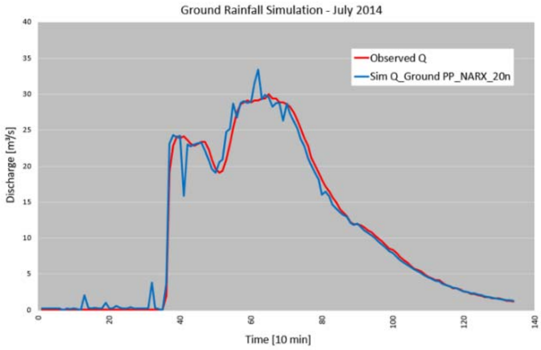

Fig. 8 -Observed and simulated runoff (July 2014) – ground precipitation measurements

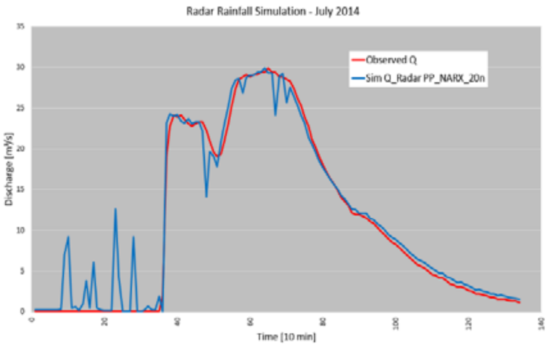

Fig. 9 - Observed and simulated runoff (July 2014) – radar precipitations

Table 1

RMSE and CE for network training

	RMSE	CE
Ground PP - NARX 20n	1,423	0,986
Radar PP - NARX 20n	2,156	0,96
RadarVol - NARX 15n	2,005	0,965

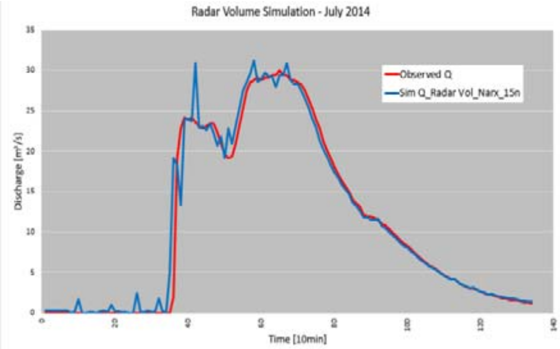

Fig. 10 - Observed and simulated runoff (July 2014) – precipitated volumes

As mentioned before, nine ANNs were tested, but here only the results with the best performance for each training dataset are presented.

The computed hydrographs in figures 8 and 10 are satisfactory, with a good agreement between simulated and observed discharges. The computed hydrograph from fig. 9 presents large perturbation prior to high discharges, but the overall hydrograph can be considered as acceptable. Perturbations affecting the simulated hydrographs are due to two main reasons. One reason is the use of nosy data and the second is the use of limited number of training patterns, given by the event-based approach. Still, the computed results are acceptable, with the overall shape of the hydrograph well simulated by all networks. The maximum simulated discharges are, in reasonable limits, close to the observed one for all results. The RMSE and CE coefficients (table 1) have satisfactory values, suggesting good performances for all networks.

3.2 ANN validation

For validating the networks, the event measured in May 2014 was used and, as mentioned before, the simulation of this event represents the main objective of the study.

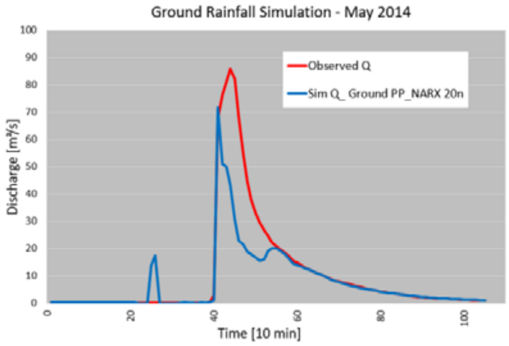

Fig. 11 - Observed and simulated runoff (May 2014) – ground precipitation measurements

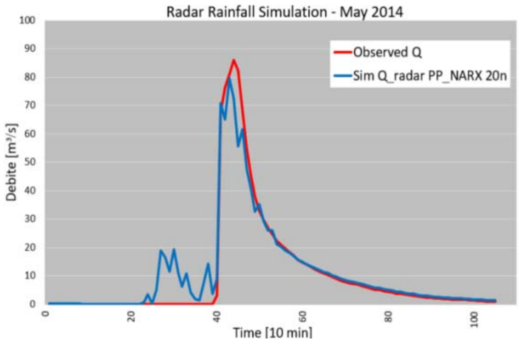

Fig. 12 - Observed and simulated runoff (May 2014) – radar rainfall

Table 2

RMSE and CE for network validation

	RMSE	CE
Ground PP - NARX 20n	10,475	0,706
Radar PP - NARX 20n	5,307	0,924
RadarVol - NARX 15n	7,106	0,863

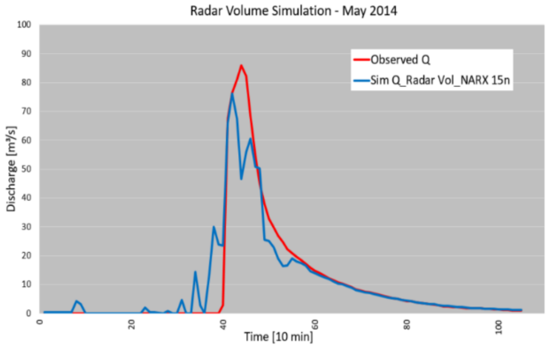

Fig. 13 - Observed and simulated runoff (May 2014) – precipitated volumes

The results for the validation dataset are presented in figures 11 - 13. Even though the ANN model that used ground precipitation had the best performances in the training phase, for the validation sets it has the poorest performances. Since there were available rainfall data only from one station, this result was expected and the poor performance can be attributed to weak spatial representation of the ground rainfall data. It can be observed that none of ANN models could simulate the maximum discharge, the results of the second network having the smallest error in this regard. Also, all results are affected by some perturbations before the rising limb of the hydrographs, this being more prominent for the last 2 results (fig. 12 and 13). The results generated for radar precipitations and volumes are similar, with some better performances for the network using radar precipitations. This is a little surprising since the same network had the weakest performances in the training phase. Regarding the CE index (table 2), the second NARX model suggest very good performance and the third network presents a good performance. Based on this, it can be suggested that the last 2 networks generated acceptable results, from hydrological point of view.

4. Conclusions

Using an event-based approach, the aim of the study was to test the ANNs capabilities of simulating a flash-flood using different types of inputs such as ground rainfall observations, radar rainfall estimates and precipitated volumes. Based on the obtained results, the NARX network was capable of modelling the rainfall-runoff process, with acceptable results for input vectors derived from radar data. For this study, the architecture setup with 15-20 neurons in the hidden layer is satisfactory for simulating the behaviour of the catchment.

Another important remark concerns the results obtained with the validation data sets, all the networks being able to extrapolate the maximum value from training phase beyond the range of the training datasets and generating discharges higher than 30 m^3/s,. There are two elements that can explain the ability to extrapolate beyond certain values, one being the autoregressive component used by the network model and another is given by the use of a linear activation function in the output layer as suggested in [14].

In regard to spatial representation of the rain, it is obvious that better spatial distribution generates improved results. The results generated by ground measurements had the poorest performance, but it is important to notice that for the validation event ground rainfall observations were available only for one station. The results obtained with radar products were superior and closer to the measured runoff, showing the importance of spatial distributed data. It can be concluded that, in places where the ground-based monitoring network has a low density, the use of radar products for rainfall-runoff simulations with ANN represents one of the best solutions.

An important remark can be made concerning the default spatial resolution of radar data. One of the modelling principles promotes the use of as much data as possible in order to obtain the best results. But, in order to satisfy this principle. there is also a need for high computational power which, depending on certain factors, can't always be achieved. Such is the case of radar data, that provides a considerable amount of rainfall information at pixel level, but which makes it hard to process without significant computing resources. Even for small catchments, such as the one analysed here, the use of default radar resolution in ANN models requires great computational power and the adopted solution to cope with this shortcoming was to average the rainfall data over a grid with a coarser resolution. Even if some information was lost during the averaging process, the results obtained for the two events are acceptable and this method can be seen as a viable solution when dealing with radar rainfall data.

Acknowledgments

The authors wish to express their sincere gratitude to ABA Prut-Barlad staff for providing the basic data involved in mathematical modelling presented in this paper.

References

[1] ASCE Task Committee on Application of Artificial Neural Networks in Hydrology (2000 I) *Artificial Neural Networks in hydrology: I: preliminary concepts*. Journal of Hydrology Engineering. 5(2), 115-123. DOI: 10.1061/(ASCE)1084-0699(2000)5:2(115)

[2] ASCE Task Committee on Application of Artificial Neural Networks in Hydrology (2000 II) *Artificial Neural Networks in hydrology: II: hydrological applications*. Journal of Hydrology Engineering. 5(2), 124-137. DOI: 10.1061/(ASCE)1084-0699(2000)5:2(124)

[3] Hsu, K., Vijai Gupta, H., & Sorooshian, S. (1995), *Artificial neural network modeling of the rainfall-runoff process*. Water Resources Research. 31(10), 2517–2530. DOI: 10.1029/95WR01955

[4] Minns, A. W. & Hall, M. J. (1996). Artificial neural networks as rainfall–runoff models. Hydrological Sciences Journal, 41(3), 399–417.

[5] Dawson, C. W. & Wilby, R. (1998), *An artificial neural network approach to rainfall-runoff modelling*. Hydrological Sciences Journal. 43(1), 47-66. DOI: 10.1080/02626669809492102

[6] Abrahart, R. J. & See, L. M. (2007), *Neural network modelling of non-linear hydrological relationships*. Hydrology and Earth System Sciences. 11, 1563–1579. DOI: 10.5194/hess-11-1563-2007

[7] Solomatine, D. P. & Ostfeld, A. (2008), *Data-driven modelling: Some past experiences and new approaches*. Journal of Hydroinformatics. 10(1), 3–22. DOI: 10.2166/hydro.2008.015

[8] Haykin, S. (1998), *Neural Networks – A comprehensive foundation*. Upper Saddle River, New Jersey, USA: Prentice-Hall.

[9] Hagan, M. T., Demuth, H. B., Beale, M. H. & De Jesus, O. (2014). *Neural Network Design (2nd ed.)*. From http://hagan.okstate.edu/NNDesign.pdf.

[10] Levenberg, K. (1944). *A Method for the Solution of Certain Non-Linear Problems in Least Squares*. Quarterly of Applied Mathematics. 2: 164–168.

[11]] Marquardt, D. (1963). A*n Algorithm for Least-Squares Estimation of Nonlinear Parameters*. *SIAM Journal on Applied Mathematics. 11 (2): 431–441. DOI:10.1137/0111030*

[12] WMO (2008), *Guide to Hydrological Practices Volume I: Hydrology – From Measurement to Hydrological Information (6th ed.),* World Meteorological Organization, No. 168

[13] Nash, J. E. & Sutcliffe, J. V. (1970), *River flow forecasting through conceptual models I: A discussion of principles* Journal of Hydrology. 10, 282–290. DOI: 10.1016/0022-1694(70)90255-6

[14] Coulibaly, P. & Baldwin, C. K. (2005), *Nonstationary hydrological time series forecasting using nonlinear dynamic methods*. J. Hydrol. 307, 164–174.

[15] Solomatine, D. P. & Dulal, K. N. (2003), *Model trees as an alternative to neural networks in rainfall–runoff modelling*. Hydrological Sciences Journal. 48, 399–411. DOI: 10.1623/hysj.48.3.399.45291