Journal of Hydroscience and Hydraulic Engineering
Vol.12, No.2, November, 1994, 131-141

Comparisons Between Fuzzy Reasoning and Neural Network Methods to Forecast Runoff Discharge

By
M.-L. Zhu

Dept. of Civil Eng., Hokkaido University, Sapporo, 060, Japan

and

M. Fujita

Dept. of Civil Eng., Hokkaido University, Sapporo, 060, Japan

SYNOPSIS

In this paper, two different approaches to forecast runoff discharge using the fuzzy inference method and the neural network method are proposed. Their applicabilities are assessed and compared through making 1-hr, 2-hr and 3-hr lead-time forecasts of runoff discharge in Butternut Creek, NY. The forecast results indicate that both proposed methods are effective for runoff forecasting.

INTRODUCTION

The basic concepts of fuzzy theory were introduced by Zadeh, L.A. (1965). The most unique characteristic of this theory in contrast to classical mathematics is its operation on variables' membership functions instead of on the crisp real values of variables. This characteristic permits fuzzy theory to be a powerful tool whenever it handles imprecise data or ambiguous relationships between variables. On the other hand, the architecture of neural networks is motivated by models of our own brains. It is difficult to specify the founder of neural network since many researchers have taken part in its development. The basic building block of the brain and nervous system is the neuron. In artificial neural networks, the model neuron is a simple non-linear processing unit. A neural network consists of many of this kind of processing unit. The characteristic of neural networks is that they are not programmed but trained by examples. Through the learning of training data sets which consist of pairs of inputs and target outputs, neural networks gradually adjust internal parameters to the point where the networks can produce a meaningful answer in response to each input. After the learning procedure is completed, information about relationship between inputs and outputs, which may be non-linear and extremely complicated, is encoded in the networks.

The foundation of these two kinds of theories are completely different; however, they have a common performance, that is, to be able to handle the problems whose mathematical definition is difficult. As we know, the relationship between rainfall and runoff is not definite due to many related factors such as field moisture capacity, evaporation rate and etc. It is difficult for conventional mathematical methods to solve this kind of problem, thus it would be attractive to try fuzzy reasoning and neural network approaches which accommodate this kind of problem.

DERIVATIONS

Fuzzy Reasoning Method to Forecast Runoff

Runoff forecasting can be classified into several cases based on accessibility of hydrological data. In this paper, we consider the case of real-time runoff forecasting in which runoff information at every moment by the current time for the present flood is available. The general equation of runoff system can be expressed as:

$$\Delta Q(t)=f\ \{R(t-1)...R(t-m),\Delta Q(t-1)...\Delta Q(t-n)\} \quad (1)$$

where R, Q, t denote rainfall, runoff discharge and time respectively; and $\Delta Q(t)=Q(t)-Q(t-1)$; parameters m, n can be chosen by taking account of the hydrological characteristics of the basin.

In this paper, we used hydrological data observed in Butternut Creek, New York for analysis. Our intend task was to forecast runoff discharge up to a lead-time of 3-hr. For this purpose, we first made 1-hr, and then 2-hr lead-time forecasts of runoff discharge $\Delta Q'(t+1)$, $\Delta Q'(t+2)$ at the present time t. That is to say, to forecast the 3-hr lead-time runoff discharge $\Delta Q'(t+3)$, we may utilized the values of $\Delta Q'(t+1)$, $\Delta Q'(t+2)$ forecasted in advance. Thus the equation for forecasting $\Delta Q'(t+3)$ can be expressed as:

$$\Delta Q'(t+3)=f\{R(t),R(t-1)...R(t-m),\Delta Q'(t+2),\Delta Q'(t+1),\Delta Q(t)...\Delta Q(t-n)\} \quad (2)$$

where the denotation "$'$" is attached to the forecasted value to distinguish it from the observed value. To avoid calculation complexity, we simplify Eq. 2 as follows:

$$\Delta Q'(t+3)=f\{R(t),\Delta Q'(t+2),\Delta Q'(t+1)\} \quad (3)$$

The basis for this simplification is that information of the ignored variables $R(t-1)...R(t-m)$, $\Delta Q(t)...\Delta Q(t-n)$ is already contained in the variables of $\Delta Q'(t+1)$, $\Delta Q'(t+2)$ somehow. The runoff system equation corresponding to the forecast equation of Eq.3 is:

$$\Delta Q(t)=f\{R(t-3),\Delta Q(t-1),\Delta Q(t-2)\} \quad (4)$$

Fuzzifying variables $R, \Delta Q$ and representing them by their membership functions M_R, $M_{\Delta Q}$ respectively, we may transform Eq. 4 into a so-called fuzzy conditional proposition as follows:

$$\textit{if } R(t-3) \textit{ is } M_{R(t-3)} \textit{ and } \Delta Q(t-1) \textit{ is } M_{\Delta Q(t-1)} \textit{ and } \Delta Q(t-2) \textit{ is } M_{\Delta Q(t-2)}$$
$$\textit{then } \Delta Q \textit{ is } M_{\Delta Q(t)} \quad (5)$$

The fuzzy reasoning process to forecast runoff may be concluded in five steps:
(1) To translate the above proposition into a fuzzy relation P_t:

$$P_t = M_{R(t-3)} \wedge M_{\Delta Q(t-1)} \wedge M_{\Delta Q(t-2)} \wedge M_{\Delta Q(t)} \tag{6}$$

where $\wedge$ denotes the minimum operator. This P_t represents the fuzzy relation between rainfall and runoff increment at time t. Since such a fuzzy relation may be obtained at every moment, by the current time t we may have got a series of this kind of fuzzy relations $P_1, P_2, \ldots, P_t$.
(2) To combine the obtained fuzzy relations $P_1, P_2, \ldots, P_t$ to produce the whole fuzzy relation Π_t through a conjunction operator:

$$\Pi_t = P_1 \vee P_2 \vee \ldots \vee P_t \tag{7}$$

where $\vee$ denotes the maximum operator. If there are previous flood data, for examples, the flood data of last year, the Π_t obtained from these previous data is denoted as Π_{te}. As the past flood information, this Π_{te} can be utilized effectively but simply by being regarded as an initial value:

$$\Pi_t^* = \Pi_{te} \vee P_1 \vee P_2 \vee \ldots \vee P_t \tag{8}$$

(3) To infer the membership function of the 1-hr lead-time runoff increment based on the above Π_t^*:

$$M_{\Delta Q'(t+1)} = \Pi_t^* \odot M_{R(t-2)} \odot M_{\Delta Q(t-1)} \odot M_{\Delta Q(t)} \tag{9}$$

where $\odot$ denotes the max-min operator. Similarly, the membership functions of 2-hr and 3-hr lead-time runoff increment can be inferred as follows:

$$M_{\Delta Q'(t+2)} = \Pi_t^* \odot M_{R(t-1)} \odot M_{\Delta Q(t)} \odot M_{\Delta Q'(t+1)} \tag{10}$$

$$M_{\Delta Q'(t+3)} = \Pi_t^* \odot M_{R(t)} \odot M_{\Delta Q'(t+1)} \odot M_{\Delta Q'(t+2)} \tag{11}$$

(4) To calculate the crisp real values of $\Delta Q'(t+1)$, $\Delta Q'(t+2)$, $\Delta Q'(t+3)$ through a defuzzy procedure which adopts the centers of gravity of the predicted membership functions.
(5) To calculate the 1-hr, 2-hr and 3-hr lead-time runoff discharge by the following equations:

$$Q'(t+1) = Q(t) + \Delta Q'(t+1) \tag{12}$$

$$Q'(t+2) = Q'(t+1) + \Delta Q'(t+2) \tag{13}$$

$$Q'(t+3) = Q'(t+2) + \Delta Q'(t+3) \tag{14}$$

The above method to forecast runoff discharge was applied to Butternut Creek, NY. (drainage area: $154.6 Km^2$). The membership functions of rainfall R(t) and runoff increment $\Delta Q(t)$ in this basin were adopted as two simple triangle functions as shown in Fig. 1 and Fig. 2, where DR and DQ represent the vaguenesses of the values of rainfall intensity and runoff increment. Fig. 1 indicates that the rainfall intensity at time t, instead of a crisp real value, may be located in an interval [R(t)−DR, R(t)+DR] with different membership grades from 1.0 to 0. Fig. 2 can be explained similarly. In the calculation, the parameters DR and DQ were selected at 2 (mm/hr) and 0.08 (mm/hr) respectively. There are data of five flood events in Butternut Creek. We used the first two floods, shown in Fig. 3, to produce Π_{te}, then made 1−hr, 2−hr and 3−hr lead−time forecasts of runoff discharge for the third flood event. Next, we regarded the Π_t obtained at the end of the third flood event as Π_{te} to forecast the fourth flood event. The same process was applied to forecast the fifth flood event.

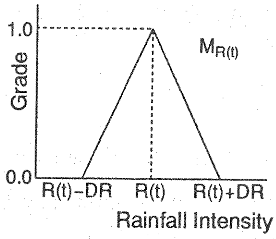

Fig. 1 Membership function of rainfall R(t)

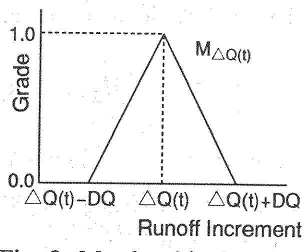

Fig. 2 Membership function of runoff increment △Q(t)

The results of 1−hr, 2−hr and 3−hr lead−time forecasts for the third flood are shown in Fig. 4(a), (b) and (c), where the solid lines denote the observed hydrographs and the black squares denote the forecasted values. From these figures, we can see the prediction error gradually enlarges from the 1−hr to the 3−hr lead−time forecasts. The forecast results for the fourth and fifth flood events are not shown here, however they have the same level of accuracy as that of the third. The forecasts may be considered accurate enough for application.

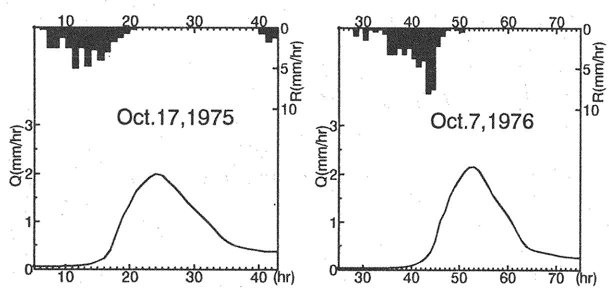

Fig. 3 The first two floods used to produce Π_{te}

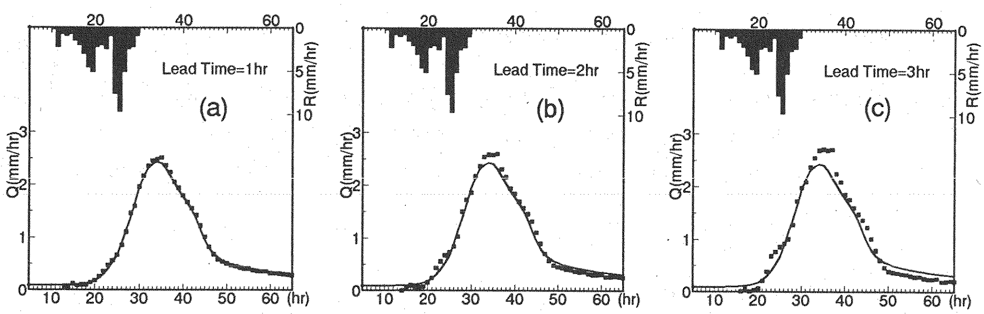

Fig. 4 Forecast results by the fuzzy reasoning method

Neural Network Method to Forecast Runoff

To forecast runoff discharge for the present storm, neural network (NN) needs first to learn about previous storm events. After the learning procedure is completed, information about the relationship between rainfall and runoff is encoded in the network and thus runoff discharge can be forecasted by this trained NN under the new rainfall inputs. The learning procedure is conducted using the back-propagation learning algorithm involving a forward-propagation step followed by a backward-propagation step. Fig. 5 illustrates a fully interconnected three-layer network, where the processing units in the input layer, hidden layer and the output layer are m, n and 1 respectively. The following describes the details of forward and backward propagation steps.

(1) Forward-propagation step: This step calculates the output from each processing unit of the neural network starting from the input layer and propagating forward through the hidden layer to the output layer. Each processing unit except the units in the input layer takes a weighted sum of the coming inputs and applies a sigmoid function to compute its output, while the processing units in the input layer, as a special case, just send the input values as they are along all the output interconnections to the units in the hidden layer.

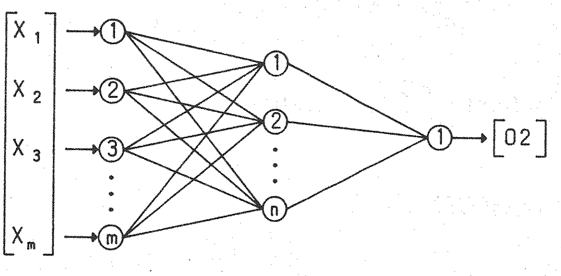

Fig. 5 A fully interconnected, three-layer neural network

Specifically, given an input vector $[x_1, x_2,...,x_m]$, the outputs from each layer are calculated in this way:

input layer: $o_i = x_i \qquad i=1,2,...,m$ (15)

hidden layer: $s1_j = \sum_{i=1}^{m} o_i w1_{ji} + \theta 1_j \qquad j=1,2,...,n$ (16)

$o1_j = f(s1_j)$

output layer: $s2 = \sum_{j=1}^{n} o1_j w2_j + \theta 2$ (15)

$o2 = f(s2)$

where o_i, $o1_j$, $o2$ are the outputs from unit i in the input layer, unit j in the hidden layer and the output unit in the output layer; $w1_{ji}$ is the interconnection weight between the i-th unit in the input layer and the j-th unit in the hidden layer; $w2_j$ is the interconnection weight between the j-th unit and the output unit in the output layer; $\theta 1_j$, $\theta 2$ are the biases of unit j in the hidden layer and the output unit in the output layer; and f is a sigmoid function.

(2) Backward-propagation step: This step is an error-correction step which takes place after the forward-propagation step is completed. The calculation begins at the output layer and progresses backward through the hidden layer to the input layer. The internal parameters of the neural network

including the interconnection weights between layers and biases in each processing unit are corrected in a way to minimize the following cost function:

$$E=\frac{(tg-o2)^2}{2} \tag{18}$$

where tg is the target output. Specifically, the correction is conducted as follows:

$$w2_j(new)=w2_j(old)+\eta(-\frac{\partial E}{\partial w2_j})=w2_j(old)+\eta\delta o1_j \tag{19}$$

$$\delta=-\frac{\partial E}{\partial s2}=(tg-o2)*f'(s2) \tag{20}$$

$$\theta2(new)=\theta2(old)+\eta(-\frac{\partial E}{\partial\theta2})=\theta2(old)+\eta\delta \tag{21}$$

$$w1_{ji}(new)=w1_{ji}(old)+\eta(-\frac{\partial E}{\partial w1_{ji}})=w1_{ji}(old)+\eta\delta_j o_i \tag{22}$$

$$\delta_j=-\frac{\partial E}{\partial s1_j}=(\delta*w2_j)*f'(s1_j) \tag{23}$$

$$\theta1_j(new)=\theta1_j(old)+\eta\delta_j \tag{24}$$

where f' denotes the derivative of the sigmoid function, and η denotes the learning rate.

During the learning procedure, the forward-propagation and backward-propagation steps are executed iteratively for the training data set, and the internal parameters of neural network are adjusted continuously. This learning procedure is ended when the produced output from neural network is hardly improved further. If the learning procedure is completed with success, namely, if the final produced output is close enough to the target output, the information of the relationship between the inputs and outputs is considered to be encoded in the network. Applying this kind of neural network to forecast runoff discharge, we may employ the following two possible techniques:

(1) Only the previous flood data is utilized as the training data set. The internal parameters of the neural network are tuned in the learning procedure and remain fixed to carry out the forecast task. That is to say, by the current time t, the known information about the present flood is not utilized to tune the neural network.

(2) Besides the previous flood data, the hourly accessible information about the present flood is also utilized to train the neural network. That is to say, the neural network forecasts floods in an adaptive forecasting environment where the internal parameters of the neural network are updated at every moment when new data on the present flood is received.

Again, we applied this method to Butternut Creek. The same runoff system equation expressed in Eq. 4 was adopted. The neural network developed in response to this runoff system equation employed is shown in Fig. 6. It should be noted that the structure of this developed neural network is partly interconnected. This is because the natures of rainfall inputs and runoff inputs are distinct and thus full interconnection is unsuitable.

First, we applied technique (1) to the forecast. The first two flood events (in Oct. 17, 1975 and Oct. 7, 1976) shown in Fig. 3 were adopted as the training data set. The internal parameters of the NN were initialized by assigning random numbers in the interval (−1, +1) and tuned by presenting the above training data set to the NN with 10,000 iterations. Since the output from the network is the increment of runoff discharge which may take positive or negative values, in our calculation, the sigmoid function described above were defined as:

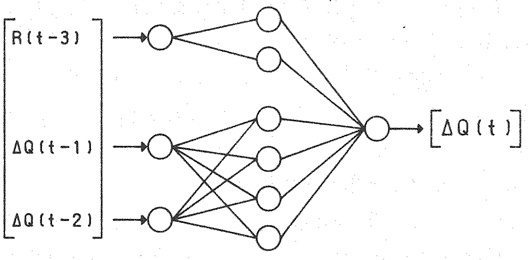

Fig. 6 Neural network developed in response to Eq. 2

$$f(x)=\frac{2}{1+e^{-x}}-1 \tag{25}$$

where the range of x is defined from $-\infty$ to $+\infty$ and the function values result in the interval (−1, 1). After the NN was tuned, its internal parameters remained fixed to forecast the last three flood events. Fig. 7 shows the forecasting schema of the NN. It can be seen from Fig. 7 that the forecasted values were used for the further lead–time forecasts just as the fuzzy reasoning method. Fig. 8(a), (b) and (c) show the results of 1–hr, 2–hr and 3–hr lead–time forecasts for the third flood event (in Oct. 20, 1976). It can be seen that the forecasts agree with the observed data fairly well from these figures. The forecast results for the fourth and fifth flood events are not shown here but they are of the same level of accuracy as the third.

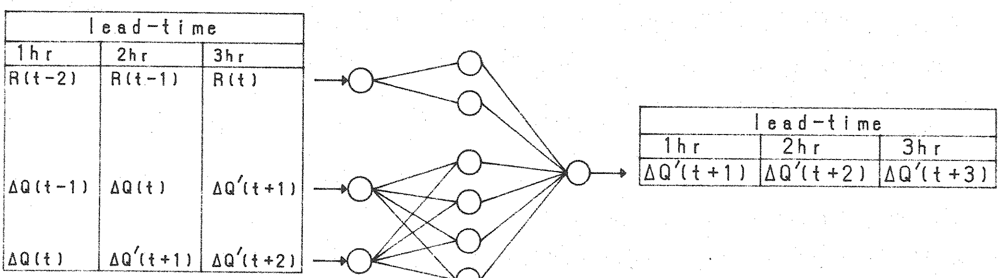

Fig. 7 Forecasting schema of neural network

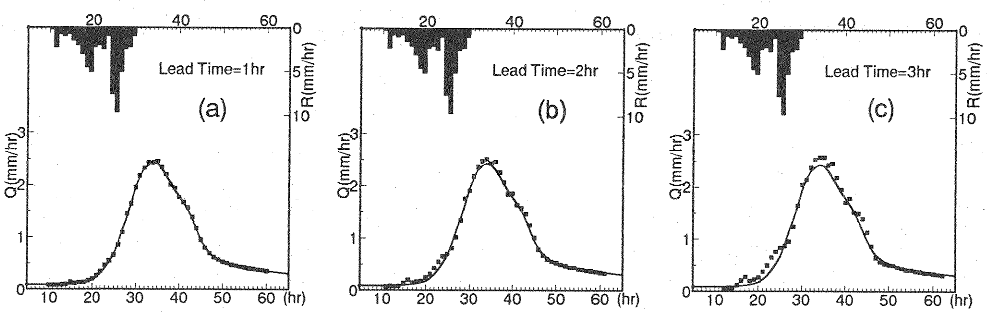

Fig. 8 Forecast results by technique (1) of the NN method

Second, we applied technique (2) to the forecast. The internal parameters of the NN were initialized by assigning the values obtained after the learning procedure in technique (1), rather than random values. To forecast the third flood, the NN was re–tuned at every moment by using a training data set which included not only the first two floods but also the hourly accessible data of the third. That is to say, the data involved in the training data set increased hourly. From the viewpoint of data utilization, technique (2) is the same as the fuzzy reasoning method since both of them utilize all available data including the present flood event. Similarly, when the NN was applied to forecast the fourth flood event, the training data included the first three events and the hourly accessible data of the fourth, and so on for forecasting the fifth flood event. To save calculation time, iteration of the training data set was limited to 100 when the NN was re–trained at every moment. The forecasting algorithm was the same as shown in Fig. 7. The forecast results for the third flood are shown in Fig. 9(a), (b) and (c).

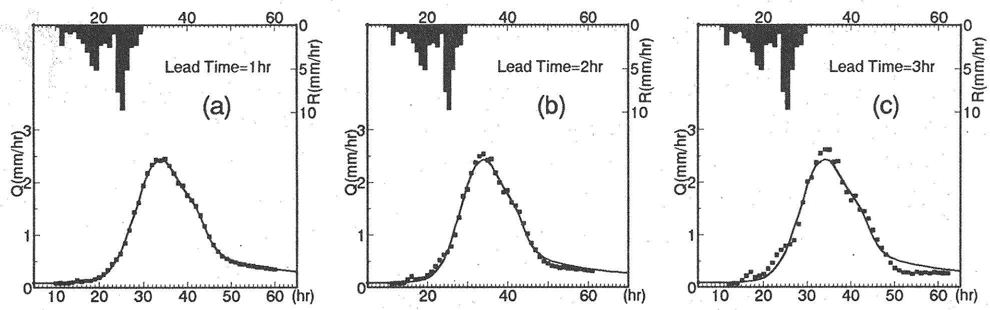

Fig. 9 Forecast results by technique (2) of the NN method

Qualitative comparisons between the results forecasted by fuzzy reasoning and two NN methods were made based on the relative error of peak flow $(Q_p^f - Q_p^o)/Q_p^o$, the time difference of peak flow $(t_p^f - t_p^o)$ and variance $\sum_{i=1}^{n}(Q^f - Q^o)_i^2/n$ (where Q_p^f, t_p^f, Q^f are the forecasted peak flow, the time to the forecasted peak flow and the forecasted flow respectively; Q_p^o, t_p^o, Q^o are the observed values and n is the number of samples in the forecasted flood). As an example, the comparison result for the third flood is shown in Table. 1. The mean calculation times for each 3–hr lead–time forecast executed at PC and WS are compared in Table. 2 (The fuzzy reasoning method was not executed at PC due to insufficient computer capacity). As concerns the calculation times for tuning the internal parameters of the NN and producing Π_{te}, they are not important factors in our forecast since these two time–consuming jobs can be executed in advance, and therefore, a comparison of this calculation time is omitted here.

Conclusions

Comparing technique (1) with technique (2) in the neural network method, the latter utilizes all flood data including the present flood data up to the current time, and thus is theoretically superior; however, this superiority was not shown in the forecast results. Two possible reasons for this unexpected result are considered as: (1) The training iteration of 100 times in the latter technique was not sufficient; and (2) the training data set used at every moment in the latter technique is somehow

Table. 1 Comparisons between the results forecasted by three different methods

		relative error of peak flow	time difference to peak flow	variance
fuzzy reasoning method	1-hr lead-time forecast	0.0319	1hr	0.00160
	2-hr lead-time forecast	0.0693	2hr	0.00914
	3-hr lead-time forecast	0.1147	1hr	0.02795
technique (1) of the neural network method	1-hr lead-time forecast	0.0092	1hr	0.00068
	2-hr lead-time forecast	0.0331	0hr	0.00406
	3-hr lead-time forecast	0.0568	0hr	0.01273
technique (2) of the neural network method	1-hr lead-time forecast	0.0069	1hr	0.00071
	2-hr lead-time forecast	0.0427	0hr	0.00461
	3-hr lead-time forecast	0.0755	0hr	0.01627

Table. 2 Comparison of mean calculation time for forecasting

	Toshiba WS AS4080	NEC PC-9801ES
fuzzy reasoning method	17.2Sec.	unknown
technique (1) of the neural network method	0.0027Sec.	0.018Sec.
technique (2) of the neural network method	1.25Sec.	51.9Sec.

biased. For example, if the present flood is still at the increasing stage by the current time, then the training data set, which includes the data about whole stage of the previous flood events and the data about only the increasing stage of the present flood event, will be of a biased tendency to the increasing stage of flood. Besides, the training procedure is carried out while forecasting for technique (2) and thus its forecast takes more time than technique (1). This can be seen from Table. 2. Therefore, it may be concluded that technique (1) is more applicable than technique (2) to runoff forecasting.

Comparing the fuzzy reasoning method with technique (1) of the neural network method, the latter shows slightly better performance than the former according to the forecast results. Moreover, the latter has the advantage of shorter calculation time for forecasting and much less computer capacity occupied. However, for the basin whose area is about 150km^2, forecast lead-time is limited to within approximately 3 hours; if a longer lead-time forecast is attempted, the rainfall forecast information is necessary. Recently, several studies on rainfall forecasting based on information provided by weather radar have been carried out, and research in this field is progressing. However, with prolonging the forecast lead-time, it will be difficult to make quantitative forecasts for future rainfall, but only qualitative forecast provided in a way as week, medium and strong intensity can be made. To utilize this kind of non–quantitative information for forecasting runoff, the fuzzy reasoning method appears very useful. The authors have attempted to make long lead–time forecasts of runoff discharge by applying the fuzzy reasoning method recently, the results show that this application is rather promising (see M.–L. Zhu and M. Fujita (4)).

REFERENCES

1. Dayhoff, J. E. : Neural Network Architectures, Van Nostrand Reinhold, New York, U.S.A., 1990
2. Fujita, M. and M.–L. Zhu: An application of fuzzy theory to runoff prediction, Proceedings of the Sixth IAHR International Symposium on Stochastic Hydraulics, Taiwan, pp. 727–734, 1992.
3. Rumelhart, D. E., J. L. McClelland, and the PDP Research Group : Parallel Distribution Processing, Vol.1, the MIT Press, Cambridge, U.S.A., 1986
4. Zhu, M.–L. and M. Fujita : Long lead time forecast of runoff using fuzzy reasoning method, Journal of Japan Society of Hydrology & Water Resources, Vol.7, No.2, 1994

APPENDIX – NOTATION

The following symbols are used in this paper:

DR, DQ = vaguenesses of the values of rainfall intensity and runoff;

E = cost function;

f, f' = sigmoid function and its derivative;

$M_R, M_{\Delta Q}$ = membership functions of rainfall and runoff increment;

NN = neural network

o_i = output from i–th processing unit in the input layer;

$o1_j$ = output from j–th processing unit in the hidden layer;

$o2_k$ = output from k–th processing unit in the output layer;

P_t = fuzzy relation at time t;

Q = runoff;

ΔQ = runoff increment;

R = rainfall;

$s1_j$ = weighted summation of inputs to j–th unit in the hidden layer;

$s2$ = weighted summation of inputs to output unit in the output layer;

t = time;

tg = target output;

$w1_{ji}$ = interconnection weight between i–th unit in the input layer and j–th unit in the hidden layer;

$w2_j$ = interconnection weight between j–th unit and the output unit in the output layer;

x_i = input to i–th processing unit in the input layer;

$\wedge, \vee$ = minimum and maximum operations;

Π_t = fuzzy relation by the current time t obtained from the present flood event;

Π_{te} = fuzzy relation obtained from the previous flood events;

Π_t' = fuzzy relation combining Π_t and Π_{te};

$\odot$ = composition operator, namely, max–min operator;

$\theta 1_j$, $\theta 2$ = biases of unit j in the hidden layer and the output unit in the output layer;

η = learning rate; and

δ, δ_j = error values for output unit in the output layer and j–th unit in the hidden layer.