# SHORT-TERM RAINFALL FORECASTING BY RADAR DATA

Xianyun CHENG* and Masato NOGUCHI**

**Abstract**: Rainfall is the most important and direct agent that causes flood disaster. It is therefore the flood forecasting essentially depends on forecasting of rainfall. With radar rainfall data remotely observed by the Foundation of River & Basin Integrated Communications, i.e. FRICS, a new methodology is developed for accomplishing short-term rainfall forecasting, wherein two main components are included: (1) settlement of rainfall vector movement with modified correlation method and Fuzzy rule, and (2) determination of spatial and temporal distribution of rainfall intensity using neural network (NN) approach. Reasonable results have been derived with a high accuracy through rainfall data on a real time basis.

**Keywords:** Short-term rainfall forecasting, Radar rainfall data, Neural networks, Fuzzy rule

## 1 INTRODUCTION

One of the major responsibilities in flood hydrology is to make efficient forecasts of the occurrence of flood events, so that flood emergency procedures can be implemented effectively. For achieving this goal practically, besides well understanding the transformation of rainfall to runoff, i.e. rainfall-runoff modelling, rainfall forecasting with an interest lead time becomes significant and urgent owing to the reason that rainfall is the most important and direct agent to cause flood. Particularly in recent years, as the continuous growth of exploiting and applying physically-based model, rainfall forecasting in both spatial and temporal distribution within the catchment comes to be obligatory. Consequently, a lot of rainfall models have been derived up to now. However, rainfall is an extremely complex and difficult problem involving many variables which are interconnected in a very complicated way, even for physically-based numerical models, not only are super-computer required for processing huge amount of data, but also the accuracy is restricted by available computational resources. Hence, a simple but practical method for rainfall prediction based only on the numerical rainfall data from FRICS is desired.

Nagasaki is located at the western part of Japan, wherein rainstorms frequently attack this region such as the Isahaya Flood in 1957 and the Nagasaki Flood in 1982. An early warning system becomes important to prevent and mitigate the damages of flood. The utilization of the Radar rain gauge provides a credible practice for acting this intention. However, because the rainfall information from FRICS does not contain the predicted one, it is necessary to develop a procedure for estimating rainfall.

The NN has great potential to handle complex and nonlinear phenomena in nature (Cheng and Noguchi, 1996), while uncertain questions can be treated well with the Fuzzy theory. The procedures for rainfall forecasting carried out here are accordingly made up two phases: the computation of rainfall movement with modified correlation method and Fuzzy rule, and the determination of rainfall intensity distribution by the neural network approach.

* Ph.D. Student, Graduate School of Marine Science & Engineering, Nagasaki University ,Nagasaki 852
** Fellow, Professor, Department of Civil Engineering, Nagasaki University, Nagasaki, 852

## 2 RADAR RAIN GAUGE DATA FROM FRICS

The radar rain data with a 30-minute interval received from a FRICS terminal, dated on June 30, 1995, have been used for conducting research in this paper. Fig.1 shows an example of a rain field, where the domain covers whole area of Nagasaki prefecture as well as the nearby sea. The area is divided into 58×30 grids, where each grid size is 6 km in longitudinal direction and 9 km in latitudinal direction. The rainfall intensity in each grid is categorized into 1 to 9 degrees represented by: 1: 1~5, 2: 5~10, 3: 10~20, 4: 20~30, 5: 30~40, 6: 40~50, 7: 50~70, 8: 70~100 and 9: 100~ (unit: mm/hr), respectively.

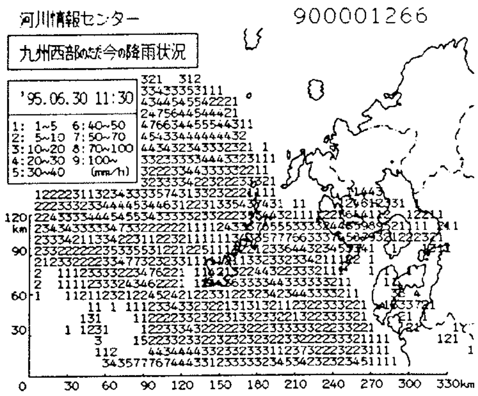

Fig. 1 Rainfall field example served by FRICS

## 3 DETERMINATION OF RAINFALL MOVEMENT

### 3.1 Movement Determination of Known Rainfall Fields

Components of a movement consist of two aspects: velocity and direction. It is very difficult to determine these two elements accurately based only on the information of known rainfall data owing to numerous influential factors such as wind speed and air pressure. Consequently, a modified correlation method is introduced here, which is mainly based on the consideration that heavy rainfall will play a dominant role in flood formation, thus requiring more attention. Mathematically, a series rainfall value-depended weights are served when applying the normal correlation method.

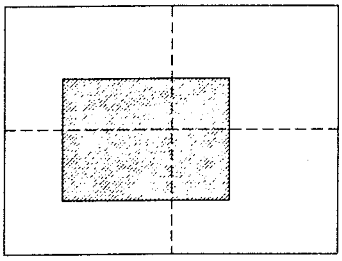
Fig.2 Geometry definition of overlapped grids

A single grid size is 6 km by 9 km, so for more precise determination of the relative movement of two adjacent rain fields, for example $A$ and $B$, at time $T1$ and $T2$, successive match comparison is operated in two steps through the computation of Eq.(1): (1) in a grid scale, to find out the maximum $R^*$, and then the distance corresponded. (2) based on the first step, the grid is further sub-divided into 10×10 grids, and a similar procedure is utilized in this sub-scale to determine the sub-distance. The final adjusted position will then be the rainfall movement from $T1$ to $T2$.

$$R^* = \frac{\sum_{i=1}^{N}(RA_i \times W_{RA_i} - \overline{RA})(RB_i - \overline{RB})}{\sqrt{\sum_{i=1}^{N}(RA_i \times W_{RA_i} - \overline{RA})^2 \sum_{i=1}^{N}(RB_i \times W_{RB_i} - \overline{RB})^2}} \quad \cdots\cdots\cdots\cdots\cdots (1) \qquad \text{and:} \quad \overline{RA} = \frac{1}{N}\sum_{i=1}^{N} RA_i \times W_{RA_i}, \quad \overline{RB} = \frac{1}{N}\sum_{i=1}^{N} RB_i \times W_{RB_i}$$

Where $RA_i$ and $RB_i$ are rainfalls in grid $i$ of vectors $A$ and $B$, respectively; $R^*$ *is* the modified correlation coefficient, different from the normal $R$; $N$ is the grid number and $W$ is the rainfall-depended weights, given by *0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8* and *0.9* for rainfall degrees of *1* to *9*, respectively.

The rainfall in all overlapped grids at time $T2$ can be re-computed referring to Fig.2. In this situation, the new rainfall value is represented by the shaded-area surrounded by four neighbored grids of the original rain field, which can be easily calculated.

### 3.2 Movement Prediction of Rainfall Fields

The variations of rainfall over a specific area can be systematized into two classes: one is external change in the horizontal direction, mainly governed by rainfall movement, and the other is comparatively called internal change in the vertical direction, corresponding to the four stages of rainfall formation: production, growth, decrease and disappearance. The first class can be decided with the procedure described in the previous section, while the succeeding connections within the second class can be simulated through the performance of a NN since its special attributes. If the NN is taken as a "black box", then a pair of adjacent rainfall vectors (as one pattern) is its input and output, respectively. A number of such patterns are used to train the NN for identifying their internal nonlinear relationships. However, it is clear that the further the rain field are apart, the weaker the connections will be and vice versa. Thus, it is unnecessary to take too many number of patterns as training examples of the NN. Additionally, the more the patterns, the longer the training time and the smaller the common parts (overlapped areas). Accordingly, for study here, three successive rainfall vectors from FRICS are used to predict the fourth vector, over the common parts. This ensures that no matter how severe the rainfall moves, the common area must include the whole Nagasaki county. Otherwise, this prediction will be meaningless. The procedures associated with a simple Fuzzy rule are developed as shown in Fig.3, where rain fields *M1*, *M2*, *M3* and *M4* are used to predict the *M5* rain field.

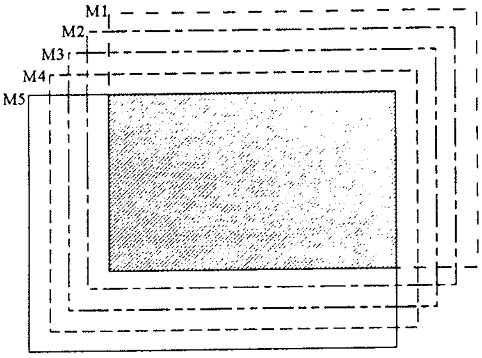

Fig.3 Illustration of overlapped rain fields

If $\overrightarrow{V1}$, $\overrightarrow{V2}$ and $\overrightarrow{V3}$ represent movement between two successive stages among *M1*, *M2*, *M3* and *M4*, which can be decided with the method described in the former section, then the predicted $\overrightarrow{V4}$ (from *M4* to *M5*) is calculated by Eq.(2), where $\varphi$, $\beta$ and $\lambda$ are the parameter weights, which can be judged using a simple Fuzzy rule. Firstly, based on Relative Humming Distance, the close degrees from *M1* and *M2* to *M3* are computed with Eqs.(3) and (4) respectively, where: $d_{13}=|\overrightarrow{V3}-\overrightarrow{V1}|$ and $d_{23}=|\overrightarrow{V3}-\overrightarrow{V2}|$ , $\mu_{12}$ and $\mu_{21}$ are close degrees from *M1* and *M2* to *M3*, respectively.

$$\overrightarrow{V4}=\begin{pmatrix}\varphi & \beta & \lambda\end{pmatrix}\begin{pmatrix}\overrightarrow{V1}\\ \overrightarrow{V2}\\ \overrightarrow{V3}\end{pmatrix}\cdots\cdots\cdots(2)$$

and

$$\mu_{12}=\frac{d_{23}}{d_{13}+d_{23}}\cdots\cdots\cdots(3)$$

$$\mu_{21}=1-\mu_{12}\quad\cdots\cdots\cdots(4)$$

Secondly, if the relative close degrees from *M1*, *M2* and *M3* to *M4* are taken as $\xi 1$, $\xi 2$ and $\xi 3$, respectively, then parameter $\varphi$ is defined as:

$$\varphi=\frac{\mu_{12}\times\xi_1}{\mu_{12}\times\xi_1+\mu_{21}\times\xi_2+\xi_3}\cdots\cdots(5)$$

Similar:

$$\beta=\frac{\mu_{21}\times\xi_2}{\mu_{12}\times\xi_1+\mu_{21}\times\xi_2+\xi_3}\cdots\cdots(6)$$

$$\lambda=\frac{\xi_3}{\mu_{12}\times\xi_1+\mu_{21}\times\xi_2+\xi_3}\cdots\cdots(7)$$

From experience, parameters of $\xi 1$, $\xi 2$ and $\xi 3$ are valued as 0.6, 0.8 and 1.0, respectively. The comparison between calculated results using this method and observed one (if we consider the results from the modified correlation method as the "observed one") is shown in Fig.4. The outcome is quite acceptable.

## 4 DETERMINATION OF RAIN INTENSITY

After the movement position and subsequently the common parts are settled, the NN can be utilized for forecasting rainfall intensity for each grid.

### 4.1 The Neural Network

A typical three-layer network has always an input layer, an output layer and a hidden layer as illustrated in Fig.5. Each layer is made up of several nodes (neurons), and layers are interconnected by sets of weights. The nodes receive input either from outside the model (the initial inputs) or from the interconnections. The main equations and relative notations of the NN are listed in Table1. Referring to Fig.6, for a neuron $k$, the operations of a node transformation are summarized as two stages: (1) integrate all incoming information signals using a propagation rule given by Eq.(8), and (2) update the level of its activation using a sigmoid function defined by Eq.(9).

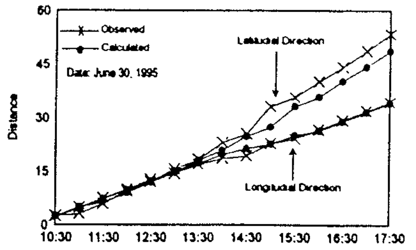


Fig.4. Comparisons Between Observed and Calculated Movement

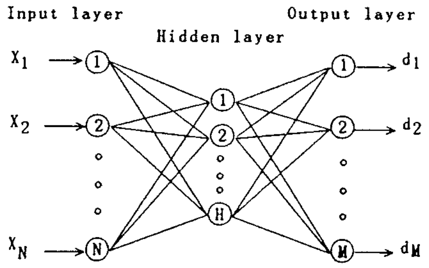


Fig.5 Typical three-layer neural network

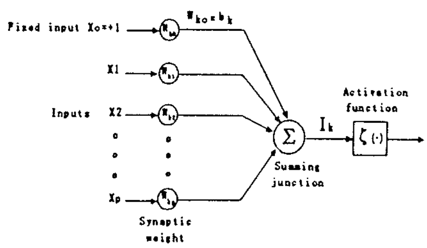


Fig.6 Non-linear models of a neuron

The training process or learning is a process by which the free parameters of a neural network are adapted through a continuing process of simulation by the environment in which the network is embedded. The type of learning is determined by the manner in which the parameter changes take place (Haykin, 1994). In this study, the error back-propagation process is introduced here as a training strategy which basically consists of two phases: (1) a forward phase, in which an input vector is propagated through the network layer by layer. Finally, the output information is produced as the response of the network. During this phase, the synaptic weights are all fixed. (2) a backward phase, in this stage, the synaptic weights are all adjusted in accordance with the difference in the calculated and desired information signal at the output unit. The training continues until either the desired criteria are achieved or some maximum number of iterations is completed.

The algorithm cycles as follows: (1) initialization: set all the synaptic weights and bias levels of the network to small random numbers, usually between *0* to *1* or *-1* to *1*. (2) forward computation: the input vector propagates forward layer by layer with Eq.(8) and Eq.(9) to produce $q(h)$ and $y(m)$. If neuron $j$ is in the output layer, hence, compute the error signal with Eq.(10). (3) backward computation: compute the 's of the network by proceeding backward layer by layer, for neuron $j$ in output layer, using Eq.(11), while for neuron $i$ in hidden layer, using Eq.(12). Hence, if take the loop of adjusting the synaptic weights of the network as a time series with index $t$, according to the generalized delta rule shown in Eq.(13), Eq.(14) is adopted from the neuron $p$ in input layer to the neuron $i$ in hidden layer, while Eq.(15) is used from the neuron $i$ in hidden layer to the neuron $j$ in output layer. (4) iteration: iterate the computation by presenting new epochs of training examples to the network until the free parameters of the network stabilize their values and the acceptable criteria are reached.

Table1 Useful equations and relative notations of the NN

| Equations | Notations |
|---|---|
| $I_k = \sum_{j=1}^{p} w_{kj} x_j$ ……(8) | $I_k$: net linear combined output |
| | $w_{kj}$: the synaptic weight of neural $k$ |
| $y_k = \zeta(\mu_k) = \frac{1}{1+\exp(\mu_k)}$ ……(9) | $y_k$: the output signal of the neural $k$ |
| | $\zeta(\cdot)$: the activation function. $u_k = I_k - b_k$, $b_k$ is the bias value |
| $e_j(m) = d_j(m) - y_j'(m)$ ……(10) | $x(n)$, $d(m)$: the input and output vector of the NN ($n=1, 2,...N$ and $m=1, 2,...M$) |
| $\delta_j(m) = e_j(m) y_j'(m)$ ……(11) | $q(h)$, $y(m)$ : the output vectors in hidden layer and output layer respectively ($h=1,2,...H$ and $m=1,2,...,M$) |
| $\delta_i(h) = q_i'(h) \sum_{j=1}^{M} \delta_j(m) w_{ij}$ ……(12) | $d_j(m)$: the jth element of the desired response vector $d(m)$ |
| $w(t+1) = w(t) + \Delta w(t)$ ……(13) | $y'(\cdot)$, $q'(\cdot)$: the derivations of the activation in hidden layer and output layer, respectively |
| $\Delta w(t) = \alpha[w_{ip}(t) - w_{ip}(t-1)] + \eta \delta_i(t) x_p(t)$…(14) | $\eta$,$\alpha$: the learning-rate and the momentum constant |
| $\Delta w(t) = \alpha[w_{ji}(t) - w_{ji}(t-1)] + \eta \delta_j(t) q_i(t)$…(15) | |

### 4.2 Rain Intensity Settlement

Based on the procedure carried out in section 3.2, three patterns (input-output) are formed as training examples of the NN, shown in Fig.7. Before conducting the training process, the number of hidden node, learning rate $\eta$ and the momentum $\alpha$ must be decided. There is no well-defined algorithm for determining the optimal number of hidden node (French, et al.,1992). Taking too high values of the hidden neural number will greatly increase the training time but without significant improvement on training results; conversely, too low values will not lead to convergence of training process. So the number adopted here is 10. The parameter $\eta$ and $\alpha$ were set at *0.1* and *0.9* at the beginning of training and decreased to 0.01 and 0.1 as the procedure goes on. This ensures to assist convergence toward the optimal solution.

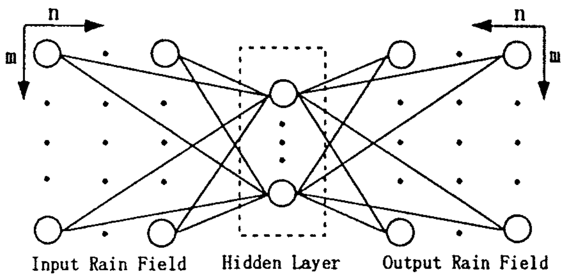


Fig.7 Neural Network for rainfall forecasting

Due to the different movement of rainfall field, the grid number within the common parts of the five successive rainfall fields are dissimilar. The criterion for judging the training result is the normal correlation coefficient $R$. The training procedure will be stopped when the $R$ values of all training examples reach *0.99*. When all synaptic weights of the NN are settled, the predicted one (outputs of the NN) can be calculated with inputs of known rainfall information. The normal correlation coefficient $R$ and *CSI* (Critical Success Index) are applied to evaluate the results of rainfall prediction. The *CSI* is defined as:

$$CSI = \frac{a}{a+b+c} \times 100\% \cdots\cdots(16)$$

where: $a$, $b$ and $c$ are total numbers connected to the grids with the situations: (1) correct prediction with rainfall, (2) wrong prediction with rainfall and (3) wrong prediction without rainfall, respectively.

The results are graphically demonstrated in Fig.8. The lead time of forecasting is 30 minutes. It can clearly be seen that *CSI* varied as the different threshold value selected. This is due to a fact that its value decreases if spatial distribution of rainfall intensity is not exactly anticipated, and also that a proportionality of rainfall field to a whole area interested is not so large. This things suggest that appropriate parameter should be programmed for estimating an accuracy of the goodness of prediction. Considering above-mentioned matter, present short-term rainfall forecasting seems to be done fairly well. A calculated rainfall field and its original one from FRICS are displayed in Fig.9 and Fig.10, respectively.

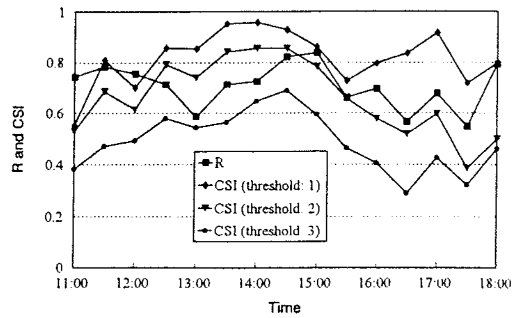

Fig 8. Temporal variations of R and CSI

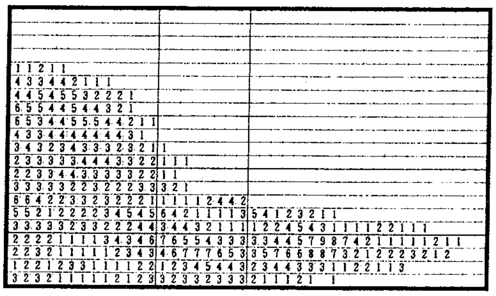
Fig.9 Observed rain field at 11:30

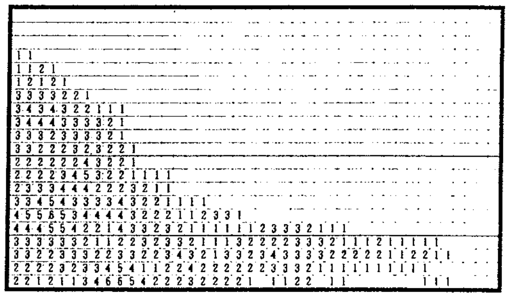
Fig.10 Calculated rain field at 11:30

## 5 CONCLUSIONS

Since rainfall is one of the most difficult elements to be determined, the outcomes obtained up to now show potential for short-term rainfall prediction. If the four stages of rainfall formation (production, growth, decrease and disappearance) can be distinguished when applying the NN in the second phase of the study, better yields can be expected.

## 6 ACKNOWLEDGEMENTS

The authors would like to express their gratitude to the Ministry of Construction and the Foundation of River & Basin Integrated Communications for use of the terminal. This study has been supported by a fund of the Grant-in-aid for Scientific Research (No. 07558059) from the Ministry of Education, Science and Culture of Japan.

## REFERENCES

Cheng, X. and M. Noguchi (1996): Rainfall-runoff Modelling by Neural Network Approach, Proceeding of International Conference on Water Resources & Environmental Research: Towards the 21st Century, Vol. 2, pp.143-150, Kyoto, Japan.

French, M. N., W. F. Krajewski and R. R. Cuykendall (1992): Rainfall Forecasting in Space and Time Using a Neural Network, J. Hydrology, 137, pp. 1-31.

Haykin, S. (1994): Neural Network - A Comprehensive Foundation, Macmillan College Publishing Company, New York.