

# 回饋式類神經網路於河川流量推估之應用

## Recurrent Neural Network for Streamflow Estimation

國立台灣大學農業工程研究所  
教授

張斐章

Fi-John Chang

國立台灣大學農業工程研究所  
碩士

黃浩倫

Hau-Lung Huang

國立台灣大學農業工程研究所  
博士班研究生

張麗秋

Li-Chiu Chang

### 摘 要

類神經網路可區分為前饋式(Feedforward)及回饋式(Recurrent)，前饋式的各類演算法中又以 Rumelhart 所提出的「倒傳遞類神經網路」(Back-propagation Neural Network, BPN)最具代表性，並有許多成功應用的實例。然而，在預測水文時間序列時，BPN 通常將時間序列以區塊式輸入(相連時間的輸入串成一輸入向量)，以建立輸入—輸出間的對應關係，預測時則由「樣本比對」方式進行，因此在不同時間點預測相同的輸入資料時，BPN 會產生相同的輸出，無法有效抓住水文序列的時變特性。

回饋式類神經網路(Recurrent Neural Network, RNN)藉由動態神經元(dynamic neurons)，有效學習時間序列的前後關係，並儲存早期的資訊留到以後使用。即時學習演算法(Real Time Recurrent Learning)的特性是不需要大量的歷史資料作為訓練範例，能隨真實環境降雨逕流特性的改變作有效而迅速的學習。本研究結合回饋式類神經網路與即時學習演算法以發展一水文推估模式，並運用於大甲溪上游流量推估，其試驗測試效果良好，顯示即時回饋式類神經網路的優越能力。

**關鍵詞：**回饋式類神經網路，動態神經元，即時學習演算法，降雨—逕流歷程。

### ABSTRACT

Based on network connection, two types of neural networks architecture are defined: feedforward and recurrent. For feedforward, the Back-propagation Neural Network (BPN) proposed by Rumelhart is probably the most famous algorithm with many satisfied applications. However, the BPN is based on the batch learning process that update action takes place only after the whole training data set has been presented. Apparently, it could not grasp the time variate characteristics of hydrological time series.

This research presents an alternative approach of the Artificial Neural Network (ANN) model to estimate streamflow. The architecture of Recurrent Neural Network (RNN) that we used provides a representation of dynamic internal feedback loops in the system to store information for later use. The Real-Time Recurrent Learning (RTRL) algorithm is implemented to enhance the learning efficiency. The main feature of the RTRL is that it doesn't need a lot of historical examples for training. Combining the RNN and RTRL to model watershed rainfall-runoff processes will complement traditional techniques in the streamflow estimation.

**Keywords:** Recurrent neural network, Dynamic neurons, Real-time recurrent learning, Rainfall-runoff processes.

## 一、前言

流量推估與預測在水資源工程的開發或營運中是非常重要的工作；流域內降雨—逕流間的因果關係，受氣象、水文、地文及人為開發等各種錯綜複雜之因素所影響，致使其水文特性兼具非線性及時變性，因此，建立合適之水文模式以正確模擬水文現象，是水資源工作者的重要目標。傳統推求河川流量的方法有單位歷線 (Sherman, 1932; Nash, 1960)、線性水庫模式 (Maidment, 1993)等，但這些傳統物理模式都有相當多的假設條件，過度簡化自然環境，因此實際應用於真實複雜的自然環境中，易產生前後不一致或難以掌握預測結果精確度之缺點。

人工智慧(Artificial Intelligence, AI)源自於生物學，模擬人類思考行為模式與人類大腦神經架構，藉由資料分類與儲存，將龐大的歷史資料資訊化，以解決傳統模式面臨的瓶頸。其主要的理論方法包含：模糊理論、類神經網路、遺傳演算法、專家系統與混沌理論等，此一科技目前已廣用於各項科學技術；如訊號處理、影像判識、自動控制等等。水文模式的建立亦可視為一種系統識別。系統識別就是由系統的輸入—輸出對應關係決定此系統的數學模型，是科學應用的重要基礎。系統識別包含兩大步驟：決定模型結構和推估模型參數。模型結構需要事先對系統的機制有充分的瞭解，但對於複雜的水文系統，這樣的知識未必隨手可得。在這種情形下我們選擇使用

神經網路來解決問題，因為使用神經網路進行系統識別只需要用到系統的輸入—輸出對應而已。類神經網路於水文上的推估即是基於這樣的想法。其優點具有高度處理信息的能力、優越的非線性映射能力以進行非線性的運算及高度的容錯能力以避免雜訊的干擾。現已有多種不同形式的類神經網路以解決不同的問題，常見的有倒傳遞類神經網路 (BP)、反傳遞類神經網路 (CPN)、Hopfield 類神經網路、SOM 類神經網路及其他結合了群聚分析、模糊理論等的多種類神經網路。

使用神經網路來進行系統識別的成功關鍵在於所使用的神經網路最好和所要識別的系統具有相似的結構，所以對於具有回饋的非線性系統，最適合的識別工具就是回饋神經網路 (Recurrent Neural Network, RNN)。回饋式神經網路的任何一層輸出都可以回饋至前一層或前數層之神經元，其靈感是來自於人腦中神經元的相互回饋。

本研究將建立一具有「人工智慧」的「即時學習之回饋式類神經網路」(Real Time Learning Recurrent Neural Network)模式，該模式藉由不斷地學習與修正，可適確的模擬推估集水區的降雨—逕流關係。

## 二、文獻回顧

神經網路的研究史可追溯到 1943 年，由 McCulloch 與 Pitts 提出神經元數學模型；1957

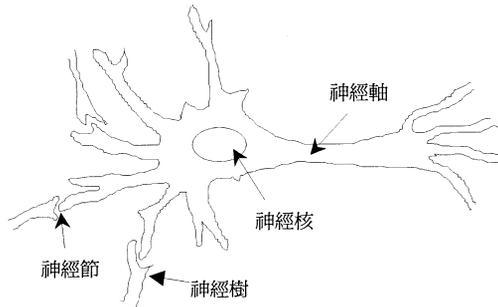


圖 1 生物神經元示意圖

年 Rosenblatt 提出認知機(Perceptron)模式，是最早的類神經網路模式；1969 年 Minsky 與 Papert 出版「認知機」一書，指出若沒有隱藏層存在的類神經網路功能有限，成為神經網路殺手，至此類神經網路的發展進入了黑暗時期；1980 年 Kohonen 提出自組織映射圖(SOM)模式；1982 年 Hopfield 提出 Hopfield 神經網路模式；1985 年 Rumelhart 等人提出倒傳遞網路(BPN)；1988 年國際神經網路學會創立。利用類神經網路於水文資訊系統於近幾年亦有相當多研究成果如：Chang & Huang (1999)、Chang et. al. (2001)、Chang & Chang (2001)、Chang & Chen (2001)。

國內近年來亦有相當多的研究成果應用於水資源相關領域，如將類神經網路應用在水資源系統上並發表於國內外期刊。1993 年張氏等人以自組性演算法應用於降雨--逕流模式，係為類神經網路應用於水文預測模式之開端；1997 年則有張與孫將類神經網路應用於降雨—逕流過程之研究；1998 年張氏等人結合類神經網路與模糊理論架構反傳遞模糊類神經網路應用於流量推估；2000 年陳與黃發表倒傳遞與反傳遞類神經網路於洪流量預測之比較。

### 三、理論概述

#### (一) 生物神經網路

神經元是中央神經系統(Central Nervous System, CNS)的基本單元。在人類大腦中約有 10 億個神經元，人類所有的活動或行為的產生都可追溯至這些神經元。每一個神經元都與許多的神

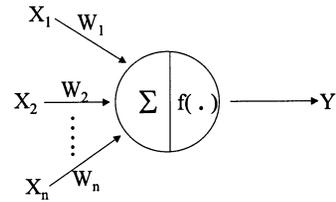


圖 2 人工神經元示意圖

經元連結，接受前面神經元傳來的訊息，經過處理再傳給後面的神經元，而形成神經網路(Neural network)。這些大量的連結提供了優異的巨量計算能力與記憶。

由訊號傳遞(Information-processing)的觀點來看，一個神經元(如圖 1 所示)可分為三部分：

- (1)樹突(dendrite): 負責接收其它神經元傳來的訊息。
- (2)神經核(soma): 負責處理其它神經元傳來的訊息。
- (3)軸突(axon): 負責傳遞訊息給其它的神經元。

一個神經元只有一條軸突，但可以有一條或多條樹突。而軸突與另一神經元的樹突的接點稱為突觸(synapse)。目前對於人類記憶過去所累積經驗與知識的機制並不非常清楚，但許多科學家認為突觸是一個關鍵的位置。在類神經網路(Artificial Neural Network)中則直接假設這個地方為存放記憶的所在。

#### (二) 人工神經網路

人工神經元又稱為人工神經細胞或處理單元(processing element)，用以模擬生物神經元，如圖 2 所示。一般可以輸入訊號的加權乘積之和的函數來表示：

$$Y = f\left(\sum_i W_i X_i\right) \dots\dots\dots(3.0)$$

其中

- $Y$  = 人工神經元輸出訊號。
- $f$  = 人工神經元的轉換函數 (transfer function)，是輸出值的數學公式。
- $W_i$  = 人工神經元的突觸強度，又稱連結加權

值。

$X_i$  = 人工神經元的輸入訊號。

### (三) 網路架構

回饋式神經網路(Recurrent Neural Network, RNN)是屬於動態神經網路(Dynamic Neural Network)，有別傳統的靜態(static)或前授(feedforward)式的網路架構。靜態式的神經網路在過去有許多成功的應用實例，但對於複雜且具回饋的非線性系統，動態神經網路會有較好的表現。原因在於動態神經網路對於人工神經元之間的相互連接較沒有限制，允許神經元間的相互回饋。如此不但更接近生物神經系統，也更加強網路的運算能力。因此，動態神經網路可以模擬出一些過去靜態神經網較難模擬的系統，例如具有時變性的系統(Time-vary system) (Connor & Martin 1994)。

本研究的網路架構為一「多層感知單元」(Multilayer Perceptron)，共可分為「連結輸入層」(concatenated input-output layer)、「回饋處理層」(processing layer)、「輸出層」(output layer)等三層。

考慮一有  $i$  個輸入、 $k$  個輸出的三層 RNN 架構，每一層都包含數個「人工神經元」(Artificial neuron)，如圖 3 所示。 $X(t)$  為在  $t$  時刻  $i \times 1$  維的輸入向量，相對應在  $t+1$  時刻  $k \times 1$  維的輸出向量為  $Z(t+1)$ 。 $Y(t+1)$  為「回饋處理層」的輸出，除了傳入「輸出層」，也在單位時間延遲後回饋給「連結輸入層」與  $X(t+1)$  作連結形成  $(i+j) \times 1$  維的輸入向量  $U(t+1)$ 。 $U(t)$  的第  $i$  個單元可表示為  $u_i(t)$ ，若  $i$  屬  $A$  集合表示  $u_i(t)$  是實際輸入值，若  $i$  屬於  $B$  集合表示  $u_i(t)$  是回饋輸入值：

$$u_i(t) = \begin{cases} x_i(t) & \text{if } i \in A \\ y_i(t) & \text{if } i \in B \end{cases} \quad (3.1)$$

網路的「連結輸入層」與「回饋處理層」內的神經元是完全地互相連結，因此有  $i \times j$  個向前傳遞的權重，有  $j \times j$  個回饋權重。這兩層間的連結權重可以維度是  $j \times (i+j)$  的矩陣  $W$  表示。

「回饋處理層」與「輸出層」內的神經元也

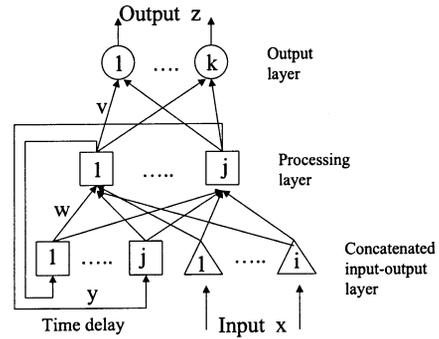


圖 3 回饋式神經網路架構圖

是完全地互相連結，因此這兩層間的連結權重可以維度是  $j \times k$  的矩陣  $V$  表示。

網路向前傳遞的方式是將輸入值  $x_i(t)$  與對應權重  $w_{ji}$  相乘後作累加得到  $net_j(t)$ ，再將  $net_j(t)$  經由一非線性函數  $f(\cdot)$  轉換，得到「回饋處理層」的輸出  $y_j(t)$ 。再將  $y_j(t)$  與相對應的權重  $v_{kj}$  相乘後作累加得到  $net_k(t)$ ，再將  $net_k(t)$  經由一非線性函數  $f(\cdot)$  轉換，得到「輸出層」的輸出  $z_k(t)$ 。

$$net_j(t) = \sum_{i \in A \cup B} w_{ji}(t-1)u_i(t-1) \quad (3.2)$$

$$y_j(t) = f(net_j(t)) \quad (3.3)$$

$$net_k(t) = \sum v_{kj}(t)y_j(t) \quad (3.4)$$

$$z_k(t) = f(net_k(t)) \quad (3.5)$$

### (四) 即時回饋學習演算法(Real Time Recurrent Learning Algorithm)：

假設  $d_k(t)$  是輸出層第  $k$  個神經元在  $t$  時刻的目標輸出值。定義在  $t$  時刻維度  $K \times 1$  的誤差向量為  $\underline{e}(t)$ ，它的第  $k$  個單元可表示如下：

$$e_k(t) = d_k(t) - z_k(t) \quad (3.6)$$

定義在  $t$  時刻的瞬時誤差函數  $E(t)$  為：

$$E(t) = \frac{1}{2} \sum_{k=1}^K e_k^2(t) \quad (3.7)$$

由「最陡坡降法」(Steepest descent method) 可得對特定權重  $v_{kj}(t)$  每次的修正量為：

$$\Delta v_{kj}(t) = -\eta \frac{\partial E(t)}{\partial v_{kj}(t)} \quad \dots\dots\dots(3.8)$$

$\eta$  為一正的常數稱為學習速率 (learning-rate)。而誤差函數  $E(t)$  對權重  $v_{kj}(t)$  的偏微分可由微積分的連鎖律(chain rule)來求得：

$$\frac{\partial E(t)}{\partial v_{kj}(t)} = -e_k(t) f'(net_k(t)) y_j(t) \quad \dots\dots\dots(3.9)$$

同樣由最陡坡降法得知對特定權重  $w_{mn}(t)$  每次的修正量為：

$$\Delta w_{mn}(t) = -\eta \frac{\partial E(t)}{\partial w_{mn}(t)} \quad \dots\dots\dots(3.10)$$

修正  $w_{mn}(t)$  所用的  $\eta$  與修正  $v_{kj}(t)$  所用的  $\eta$  可以為不同值。而誤差函數  $E(t)$  對權重  $w_{ji}(t)$  的偏微分同樣可由連鎖律來求得：

$$\frac{\partial E(t)}{\partial w_{mn}(t)} = \left[ \sum_1^K -e_k(t) f'(net_k(t)) v_{kj} \right] \frac{\partial y_j(t)}{\partial w_{mn}} \quad (3.11)$$

$$\Rightarrow \frac{\partial y_j(t)}{\partial w_{mn}} = f'(net_j(t)) \frac{\partial net_j(t)}{\partial w_{mn}} \quad \dots\dots\dots(3.11a)$$

$$\Rightarrow \frac{\partial net_j(t)}{\partial w_{mn}} = \sum_{i \in A \cup B} \frac{\partial (w_{ji}(t-1) u_i(t-1))}{\partial w_{mn}} \quad \dots\dots\dots(3.11b)$$

$$\Rightarrow \frac{\partial net_j(t)}{\partial w_{mn}} = \sum_{i \in A \cup B} \left[ w_{ji}(t-1) \frac{\partial u_i(t-1)}{\partial w_{mn}} + \frac{\partial w_{ji}(t-1)}{\partial w_{mn}} u_i(t-1) \right] \quad \dots\dots\dots(3.12)$$

當  $j=m$  且  $i=n$  時,  $w_{ji}$  對  $w_{mn}$  的偏微分為 1, 否則為 0。故上式可改寫為：

其中  $\delta_{mj}$  為 Kronecker delta function, 當  $m = j$  時  $\delta_{mj}$  為 1, 否則為 0。另外, 由  $u_i(t)$  定義可得知：

$$\frac{\partial net_j(t)}{\partial w_{mn}} = \sum_{i \in A \cup B} w_{ji}(t-1) \frac{\partial u_i(t-1)}{\partial w_{mn}} + \delta_{mj} u_i(t-1) \quad \dots\dots\dots(3.13)$$

因此  $y_j$  對權重  $w_{mn}$  的偏微分可改寫為：

$$\frac{\partial u_i(t-1)}{\partial w_{mn}} = \begin{cases} 0 & \text{if } i \in A \\ \frac{\partial y_j(t-1)}{\partial w_{mn}} & \text{if } i \in B \end{cases} \quad \dots\dots\dots(3.14)$$

對於遞迴式的初始條件是假設在  $t = 0$  的時候  $\mathbf{Y}$  與  $\mathbf{W}$  是互相獨立。因此：

$$\frac{\partial y_j(t)}{\partial w_{mn}} = f'(net_j(t)) \left[ \sum_{i \in B} w_{ji}(t-1) \frac{\partial y_i(t-1)}{\partial w_{mn}} + \delta_{mn} u_n(t-1) \right] \quad \dots\dots\dots(3.15)$$

$$\frac{\partial y_j(0)}{\partial w_{mn}} = 0 \quad \dots\dots\dots(3.16)$$

我們定義三維的新變數  $\pi_{mn}^j(t)$ , 稱為「動態變數」(dynamic variable)。

$$\pi_{mn}^j(t) = \frac{\partial y_j(t)}{\partial w_{mn}} \quad \text{for all } j \in B, m \in B, n \in A \cup B \quad \dots\dots\dots(3.17)$$

重新改寫遞迴式為：

$$\pi_{mn}^j(t+1) = f'(net_j(t)) \left[ \sum_{m \in B} w_{ji}(t) \pi_{mn}^j(t) + \delta_{mn} u_n(t) \right] \quad \dots\dots\dots(3.18)$$

初始條件為  $\pi_{mn}^j(0) = 0$

故權重  $w_{mn}(t)$  的修正量最後可寫為：

$$\Delta w_{mn}(t) = \eta \left[ \sum_1^K e_k(t) f'(net_k(t)) v_{kj} \right] \pi_{mn}^j(t) \quad \dots\dots\dots(3.19)$$

最後歸納 RTRL 的步驟如下：

- (1) 將第一個時間的  $\mathbf{X}$  代入網路計算出  $\mathbf{Z}$ , 並將  $\mathbf{Y}$  經過一單位時間延遲後回饋到「連結輸入層」。
- (2) 計算  $\Delta \mathbf{Y}$ 、 $\Delta \mathbf{W}$ , 並更新權重。
- (3) 再代入下個時間的  $\mathbf{X}$ 。並重複以上的步驟。

#### 四、回饋式神經網路於即時流量的推估

德基水庫為一多目標水庫系統, 位於大甲溪水庫串聯系統之首, 所以德基水庫的操作有其重要性, 如果可以得到準確的即時流量預測, 則對於水庫的操作將有很大的貢獻。在本節中將以即時修正參數之回饋式類神經網路預測德基水庫上游的松茂即時流量站的時流量資料。

松茂流量站上游有四個雨量站分別是志佳陽、平岩山、捫山、松峰等, 其地理位置圖 4 所示。



圖 4 德基水庫上游集水區水文站分佈圖

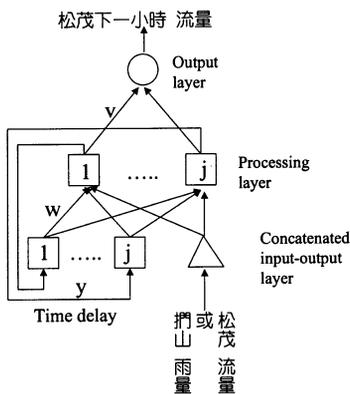


圖 5 單一輸入網路圖

首先建立一個網路只有一個輸入一個輸出(如圖 5 所示)。分別以攔山雨量站這一小時的雨量及松茂流量站這一小時的流量當作網路的輸入，松茂流量站下一小時的流量當作輸出。

接著建立一個網路有四個輸入一個輸出(如圖 6 所示)。輸入為這一小時志佳陽、平岩山、攔山、松峰四個雨量站的雨量，輸出為下一小時松茂流量站的流量

最後建立一個網路有五個輸入一個輸出(如圖 7 所示)。輸入為這一小時志佳陽、平岩山、攔山、松峰四個雨量站的雨量及松茂流量站的流量，輸出為下一小時松茂流量站的流量。

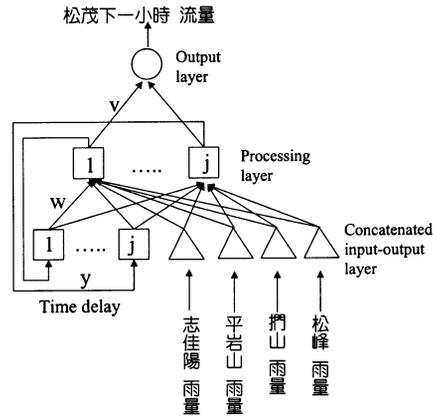


圖 6 四個輸入網路圖

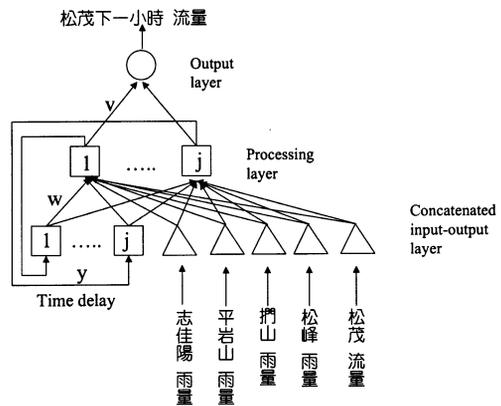


圖 7 五個輸入網路圖

以 1995 年的資料來檢定網路結構，選定處理層的神經元個數為 2，來預測 1996 年一整年的流量。比較以上六種不同的輸入對預測即時流量的影響。除了 MAE(Mean absolute error)值，另外再加入正確率 RMAE(Relative mean absolute error)值作為比較的標準。

$$MAE = \frac{\sum_{i=1}^n |\hat{Q}_i - Q_i|}{n}$$

$$RMAE = \frac{MAE}{\bar{Q}}$$

其中  $\hat{Q}_i$  為第  $i$  期預測入流量， $Q_i$  為實際入流量， $\bar{Q}$  為平均入流量

表 1 1996 年流量預測結果比較

比較時期 輸入項	1996 一整年		1996 一整年 (扣除初期不準確預測值)	
	MAE	RMAE	MAE	RMAE
松茂流量	0.835	0.035	0.762	0.032
四個雨量站的雨量	1.235	0.052	1.146	0.048
四雨量站加松茂	0.771	0.032	0.734	0.031

RTRL 的特色在於 on-line 操作及即時訓練的能力，一開始因資料訊息的不足，推估會相當不準確，但很快就可以達到穩定且準確的推估。以推估 1996 年松茂站流量為例，大約經過 20 到 30 次輸入輸出後就可以達到較準確的推估。

由表可以看出，在推估 1996 年一整年的流量時，沒有扣除初期不準確推估值與扣除後(約扣除 25 個)的 MAE 值及 RMAE 值有差異。在扣除初期不準確推估值後的 MAE 值與 RMAE 值都有下降，這樣的現象表示初期的推估並不理想。需要經過約一天 24 筆的資料，可以開始有比較準確的推估。在所推估的一整年八千多筆資料，24 筆算是少數，所以 RNN 可以在歷史資料不長的情況下很快就進入較準確的預測。

另外看出在三種不同的輸入組合中以圖 7 的 RNN 架構有最準確的推估值，也就是 RNN 的輸入一共有 5 個包含 4 個雨量站雨量及松茂本身的流量。

接著我們以圖 7 的網路架構來推估 1992 到 1995 年的時流量，輸入有 5 個分別為志佳陽、平岩山、捫山、松峰四個雨量站的雨量及松茂流量站的流量，輸出為下一小時松茂流量站的流量。

在資料缺失部分的處理是，直接刪除該筆資料不進行補遺，因此每年的資料量都不盡相同；1992 年有 5592 筆、1993 年有 8758 筆、1994 年有 7519 筆、1995 年有 8758 筆、1996 年有 8783 筆。

由表 2 可以看出 RNN 對各年即時流量的推估非常準確，MAE 值皆小於 2 cms，而 RMAE

表 2 RNN 推估各年即時流量比較表

	實際流量平 均值(cms)	推估流量平 均值(cms)	MAE	RMAE
1992	45.87	45.91	1.13	0.025
1993	15.85	15.85	0.27	0.017
1994	34.66	34.72	1.68	0.048
1995	19.01	19.02	0.31	0.016
1996	24.04	24.08	0.73	0.030

值亦多小於 5%，而實際流量與推估流量的平均值的差異亦極為有限。

## 五、結論

系統識別係從動態系統所觀測的資訊中建立系統的數學模式，系統識別主要由兩大步驟組成：模型結構的決定以及模型參數的推估。模型結構的決定需先對系統的結構特性有所瞭解，然而現實的系統常因過於複雜或由於雜訊的因素使我們無法事先決定其模型結構，此時非傳統的識別工具—神經網路便是最佳的選擇。我們所要模擬的水文系統為具有回饋的非線性系統，因此選用同樣具有回饋及非線性特性的「回饋式神經網路(RNN)」。而在模型參數的推估上，我們使用的是「即時回饋學習演算法(RTRL Algorithm)」。

本研究利用大甲流域德基水庫上游松茂水文站的流量推估，證實 RNN 搭配 RTRL Algorithm 是一種簡單而有效模擬水文系統的方式。針對歷年之降雨—逕流資料，進行流量推估，其平均絕對誤差(MAE)小於 2 cms，而相對平均誤差(RMAE)則非常接近零(小於 5%)，顯示此一方法可有效且精確的推估與預測松茂水文站下一期的流量。

對於缺乏歷史資料的地區，因為無法提供足夠的訓練範例，故其它的類神經網路將無法作有效的模擬，也不容易以統計方法作出準確的推估。在本研究中可以證實 RTRL 演算法具有優異的 On-Line 線上即時訓練能力，經過數十個數距的疊代運算後，系統就可趨於穩定，故可適用於缺乏歷史資料的地區。RNN 的網路架構有別於

傳統多層前向式(MLP)的網路架構，其特點除了更接近生物神經網路外，在本研究中不同的時間有相同的降雨可以有不同的流量輸出，比傳統的網路只要輸入相同輸出就會相同，RNN 更接近真實的自然情況。

## 六、參考文獻

1. Chang, F.-J. and Y.-Y. Huang, "A Self-organization algorithm for real-time flood forecast", *Hydrological Processes*, vol.13, n.2, pp.123-138, 1999.
2. Chang L.-C. and F.-J. Chang, "Intelligent control of modeling of real time reservoir operation", *Hydrological Processes*, 2001, (inpress.)
3. Chang F.-J. and Y.-C. Chen, "A counter-propagation fuzzy-neural network modelling approach to real-time streamflow prediction", *J. of Hydrological* vol.245, pp. 153-164, 2001.
4. Chang F.-J., H.-F. Hu, and Y.-C. Chen, "Counterpropagation fuzzy-neural network for streamflow reconstructing", *Hydrological Processes*, vol.15, n.2, pp.219-232, 2001.
5. Connor, J. T. and R. D. Martin, "Recurrent neural networks and robust time series prediction", *IEEE Trans. on Neural Network*, vol. 5, n.2, pp.240-254, 1994.
6. Hopfield, J. J., "Neuron with graded response have collective computational properties like those of two-state neurons", *Proc. Nat. Sci.*, vol.81, pp.3088-3092, 1984.
7. Hsu, K. -L., H. V. Gupta, and S. Sorooshian, "Artificial Neural Network Modeling of the Rainfall-Runoff process", *Water Resour. Res.*, vol.31, pp.2517-2530, 1995.
8. Ivakhnenko, A. G., "Heuristic Self-organization in Problem of Engineer Cybernetics", *Automatica*, vol.6, pp.207-219, 1970.
9. Kohonen, T, *Self-organization and associative memory*, 2nd edition, Berlin, Germany: Springer-Verlag, 1998.
10. Maidment, D. R., *Handbook of Hydrology*, McGraw-Hill inc., 1993.
11. Minsky, M. L. and S. A. Papert, *Perceptrons*, Cambridge, MA: MIT Press, 1969.
12. Nash, J. E., "A Unit Hydrograph study with particular reference to British Catchments", *Proc. Inst. Civ. Engrs.*, vol. 17, pp.249-282, 1960.
13. Nelder, A. J. and R. Mead, "A Simplex Method for Function Minimization", *Comput. J.*, vol.7, n.4, pp.308-313, 1965.
14. Nie, J., "Nonlinear time-series forecasting: A fuzzy-neural approach", *Neurocomputing*, ELSEVIER, vol.16 pp.63-76, 1997.
15. Nie, J. and D.A. Linkens, "Fast self-learning multivariable fuzzy controllers constructed from a modified CPN network", *Intl. J. of control*, vol.60, n.3, pp.369-393, 1994.
16. Rosenblatt, F., "The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain", *Psych. Rev.*, vol.65, pp.386-408, 1958.
17. Rumelhart, D. E., G. E. Hinton, and R. J. Williams, "Learning Internal Representation by Error Propagation", *Parallel Distributed Processing*, vol.1, pp. 318-362, 1986.
18. Sherman, L. K., "Streamflow from rainfall by the unit-graph method", *Eng. News record*, vol.108, pp.501-505, 1932.
19. 張斐章、黃源義、徐國麟，1992，"自組性演算法及其應用於降雨--逕流模式建立之研究"，臺灣大學農學院研究報告，Vol. 32, No. 4, pp.293-316
20. 張斐章、黃源義、梁晉銘，模糊推論模式之建立及其應用於水文系統之研究"，中國農業工程學報，第 39 卷，第 1 期，pp.71-83，1993。
21. 張斐章、孫建平，"類神經網路及其應用於降雨—逕流過程之研究"，中國農工學報，第 43 卷，第 1 期，pp.9-25，1997。
22. 張斐章、胡湘帆、黃源義，"反傳遞模糊類神經網路於流量推估之應用"，中國農工學報，第 44 卷，第 2 期，pp.26-38，1998。
23. 陳昶憲、黃尹龍，"倒傳遞與反傳遞類神經網路於洪流量預測之比較"，台灣水利，第 48 卷，第 3 期，pp.60-68。

收稿日期：民國 89 年 11 月 19 日

接受日期：民國 90 年 2 月 16 日