

回饋式類神經網路於二階段即時流量預測

Devising a Two-Step Real-Time Recurrent Learning Algorithm for Streamflow Forecasting

江 衍 銘

YEN-MING CHIANG

國立台灣大學

生物環境系統工程研究所

碩士班研究生

張 麗 秋

LI-CHIUE CHANG

國立台灣大學

生物環境系統工程研究所

博士後研究

張 斐 章

FI-JOHN CHANG

國立台灣大學

生物環境系統工程研究所

教授兼水工所研究員

摘 要

回饋式類神經網路的神經元間可相互回饋，故可有效的學習時間序列的前後關係，並儲存早期的資料留到未來使用。即時回饋學習演算法的主要特性是不需要大量的歷史資料作為訓練範例，可以直接將網路在線上一邊訓練一邊操作，能隨真實環境的變化做有效率的學習。以往的水文預測模式，不論是傳統的物理模式或前向式的類神經網路，對於河川流量的預測大都著重於預測下一小時的流量，惟因台灣地形特殊，洪峰往往在 2~3 個小時就到達，故若能將預測的時距往後延長，將有助於水資源規劃操作者更多的決策時間。因此針對此一特性，本研究旨在發展二階段回饋式類神經網路，以推估下二小時之流量；為證明模式的實用性與可行性，我們以大甲溪上游流量的推估結果，並與 ARMAX 模式作比較，證實回饋式類神經網路具有良好的學習能力及優越的推估能力。

關鍵詞：回饋式類神經網路，即時回饋學習演算法，線上學習。

ABSTRACT

The architecture of Recurrent Neural Network (RNN) provides an efficient representation of dynamic internal feedback loops in the system to store information for later use. The Real-Time Recurrent Learning (RTRL) algorithm deriving for RNN is an on-line learning. The main feature of the RTRL is that it doesn't need a great deal of historical data for training, and it can perform on-line learning with training and operating at the same time. Most of the traditional physical models or feed-forward neural networks for streamflow forecasting emphasize the one-step ahead forecasting. Due to the high mountain and steep channel all over the Taiwan island, the peak flow usually approach downstream about 2 or 3 hours after heavy rainfall on the watershed. Consequently, it is desired and will be very beneficial if the model can provide multi-step ahead forecasting. The main aim of this study is to develop a two-step ahead algorithm based RTRL. For the purpose of comparison, the predictive ability of RTRL and ARMAX model are performed by using the rainfall-runoff data of the Da-Chia River. Our results demonstrate that the RTRL has a learning capacity with high efficiency and is an adequate model for time series prediction.

Keywords: Recurrent Neural Network, Real-Time Recurrent Learning algorithm, On-line learning.

一、前言

台灣地區因地處亞熱帶與熱帶氣候的交接處，又屬於海島型國家，年降雨量相當豐沛，加上夏、秋兩季颱風頻繁，且山高坡陡，只要豪雨一來，上游的洪水在短暫的時間內很快的便到達下游平原，往往因為流量太大而造成河道潰堤，以致洪災屢有所聞，影響沿岸與下游居民生命與財產的安全。而在台灣這種特殊的降雨—逕流特徵，即時流量的預測成為一件困難的工作；傳統上推估河川流量的模式，大都著重於物理模式、概念模式為主，如單位歷線 (Sherman, 1932; Nash, 1960)、線性水庫 (Maidment, 1993) 而這些模式都有相當多的假設條件，造成過度簡化實際問題，產生了難以掌握預測結果的精確度；然而在傳統理論與物理模式無法突破瓶頸之處，也漸漸由目前倍受重視，廣為發展的類神經網路代之為解決許多不同研究領域上的困境。

類神經網路可以說是一種計算系統，它使用大量且簡單的人工神經元來模仿生物神經網路的能力。人工神經元是生物神經元的簡單模擬，它從外界環境或其它神經元取得資訊，經過簡單的運算程序後，輸出其結果到外界或其它神經元。而回饋式類神經網路屬於動態神經網路，對人工神經元之間的相互連接限制較小，允許神經元之間的相互回饋，以儲存序列的前後關係。類神經網路應用的成功關鍵在於網路模式最好和模擬系統具有相似的結構與特性，因此對於具有非線性時序特性的水文系統，回饋式類神經網路便是最適合的識別工具。

然而不論是傳統的物理模式或前向式的類神經網路，對於河川流量的預測皆著重於下一小時的預測，但台灣地區因地形特殊、河川坡陡湍急，洪峰往往在很短的時間就到達，因此若能提供較長時間的預測，便能在洪水事件來臨前，提供防洪預警準備工作與決策者較充裕的時間。本研究以即時回饋學習演算法 (RTRL) 為架構，推演下二小時預估模式之演算法，並應用於大甲溪流域德基水庫入流量預測。

二、文獻回顧

目前國內外已有相當多利用類神經網路於河川

流量之預測上，國內有張、孫 (1997) 將類神經網路應用於集水區內之降雨—逕流過程，其網路架構是採用 MLP，並以 EBP 為學習演算法，應用於推估暴雨事件的逕流量；張、胡 (1998) 將 R. Hecht-Nielsen 提出的 CPN 神經網路應用於推估暴雨事件的流量，其所得之結果也相當令人滿意；陳等 (1996) 以類神經網路對烏溪流域之洪流預報做研究；國外則有 Atiya and Shaheen (1999) 比較在各種不同情況下之類神經網路對於尼羅河流量預測之研究；Chang and Chen (2001) 則是應用反傳遞模糊類神經網路推估颱風時期之流量；Chang 等 (2001) 以輻狀基底函數網路建立集水區內降雨—逕流模式。而上述的模式所採用的類神經網路架構大多屬於前向式多層感知器 (Feed-forward Multilayer Perceptron, MLP)，但是前向式神經網路卻無法針對具有時序特性的資料，做時間關聯性的學習。然而回饋式類神經網路卻可藉由動態神經元有效的學習時間序列的前後關係，並儲存早期的資料留到以後使用。且回饋式類神經網路的特點在於學習速度快、網路具穩定性與可塑性、執行時間短、網路收斂快等，而即時回饋學習演算法屬於線上學習，其特性是不需要大量的歷史資料作為訓練範例，而是將資料直接送上網路，一邊訓練一邊預測，故能隨真實的自然環境變化做有效且迅速的學習，本研究旨在沿續之前研究 (張等 2001、Chang 2002) 發展二階段回饋式類神經網路的演算法，並應用於預測二時刻後的河川流量。

三、網路架構

即時回饋學習演算法 (RTRL: Real-Time Recurrent Learning Algorithm) 的網路架構是三層的回饋式類神經網路 (RNN: Recurrent Neural Network)，其網路傳遞的方式為輸入層 (t) 傳至隱藏層 ($t+1$)，隱藏層 ($t+1$) 之輸出在同一時間傳至輸出層 ($t+1$) 及回饋至輸入層 ($t+1$)，此架構是以推估下一時刻之流量為目的；而本研究所推導的二階段 RTRL 則以原一階段的 RTRL 架構為基礎，不同之處則是改變了隱藏層到輸出層間的傳輸，將原本從隱藏層 ($t+1$) 到輸出層 ($t+1$) 的方式改變，即從隱藏層 ($t+1$) 到輸出層 ($t+2$) 的轉換方式；其詳細演算法之推演如后。

二階段 RTRL 的網路架構為一多層感知單元

(Multi layer Perceptron)，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。考慮一具有1~M個輸入神經元之連結輸入層、1~K個輸出神經元之輸出層、1~N個回饋處理神經元之回饋處理層的三層回饋式神經網路架構，如圖1所示。

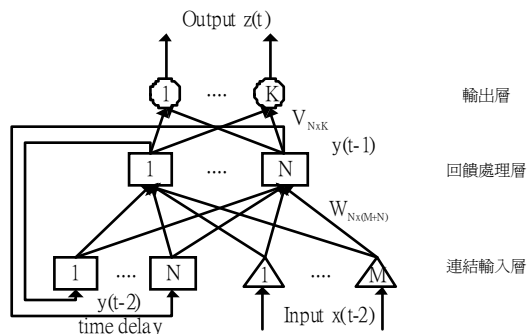


圖1 完全相互連結之二階段回饋式神經網路架構圖

$x(t-2)$ 為在 $t-2$ 時刻 $M \times 1$ 維的輸入向量，而相對應在 t 時刻 $K \times 1$ 維的輸出向量為 $z(t)$ 。 $y(t-1)$ 為 $t-1$ 時刻 $N \times 1$ 維之「回饋處理層」的輸出，除了傳入「輸出層」，也在同一時間全部回饋到「連結輸入層」與 $x(t-1)$ 作連結，形成 $(M+N) \times 1$ 維的輸入向量 $U(t)$ 。 $U(t)$ 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：

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

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 $M \times N$ 個向前傳遞的權重，有 $N \times N$ 個回饋權重。這兩層間的連結權重可以維度是 $N \times (M+N)$ 的矩陣 W 表示。而「回饋處理層」與「輸出層」內的神經元也是完全地互相連結，因此這兩層間的連結權重可以維度是 $N \times K$ 的矩陣 V 表示。

網路向前傳遞的方式是將輸入值 $x_i(t-2)$ 與對應權重 w_{ji} 相乘後做累加得到 $net_j(t-1)$ ：

$$net_j(t-1) = \sum_{i \in A \cup B} w_{ji}(t-2)u_i(t-2)$$

再將 $net_j(t-1)$ 經由一非線性函數 $f(\cdot)$ 轉換，得到「回饋處理層」的輸出 y_j ：

$$y_j(t-1) = f(net_j(t-1))$$

再將 y_j 與相對應的權重 v_{kj} 相乘後作累加得到 $net_k(t)$ ：

$$net_k(t) = \sum v_{kj}(t-1)y_j(t-1)$$

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 。

$$z_k(t) = f(net_k(t))$$

四、即時回饋學習演算法 (RTRL)

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 $e(t)$ ，它的第 k 個單元可表示如下：

$$e_k(t) = d_k(t) - z_k(t)$$

定義在 t 時刻的瞬時誤差函數 $E(t)$ 為：

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

定義在 t 時刻網路的總誤差 E_{total} 為：

$$E_{total} = \sum_{t=1}^T E(t)$$

而為了使總誤差為最小，應用「陡降法」(Gradient descent method) 可得知對特定權重 $v_{kj}(t)$ 每次的修正量為：

$$\Delta v_{kj}(t-1) = -\eta_1 \frac{\partial E(t)}{\partial v_{kj}(t-1)}$$

η_1 為一大於零的常數稱為學習速率 (learning-rate)。而誤差函數 $E(t)$ 對權重 $v_{kj}(t-1)$ 的偏微分可由微積分的連鎖律 (chain rule) 來求得：

$$\frac{\partial E(t)}{\partial v_{kj}(t-1)} = \frac{\partial [\frac{1}{2} \sum_{k=1}^K e_k^2(t)]}{\partial v_{kj}(t-1)} = \sum_{k=1}^K e_k(t) \frac{\partial [e_k(t)]}{\partial v_{kj}(t-1)}$$

$$= \sum_{k=1}^K e_k(t) \frac{\partial [d_k(t) - z_k(t)]}{\partial v_{kj}(t-1)} = \sum_{k=1}^K e_k(t) \frac{\partial [-f(net_k(t))]}{\partial v_{kj}(t-1)}$$

$$= - \sum_{k=1}^K e_k(t) f'(net_k(t)) y_j(t-1)$$

故

$$\Delta v_{kj}(t-1) = \eta_1 \left[\sum_{k=1}^K e_k(t) f'(net_k(t)) y_j(t-1) \right]$$

同樣由陡降法得知對特定權重 $w_{mn}(t-2)$ 每次的修正量為：

$$\Delta w_{mn}(t-2) = -\eta_2 \frac{\partial E(t)}{\partial w_{mn}(t-2)}$$

修正 $w_{mn}(t-2)$ 所用的 η_2 為另一大於零的學習速率。而誤差函數 $E(t)$ 對權重 $w_{mn}(t-2)$ 的偏微分同樣可由連鎖律來求得：

$$\frac{\partial E(t)}{\partial w_{mn}(t-2)} = \left[\sum_{j=1}^J -e_j(t) f'(net_j(t)) v_{mj}(t-1) \right] \frac{\partial y_j(t-1)}{\partial w_{mn}(t-2)}$$

$$\Rightarrow \frac{\partial y_j(t-1)}{\partial w_{mn}(t-2)} = f'(net_j(t-1)) \frac{\partial net_j(t-1)}{\partial w_{mn}(t-2)}$$

$$\Rightarrow \frac{\partial net_j(t-1)}{\partial w_{mn}(t-2)} = \sum_{i \in A \cup B} \frac{\partial (w_{ji}(t-2) u_i(t-2))}{\partial w_{mn}(t-2)}$$

$$\Rightarrow \frac{\partial net_j(t-1)}{\partial w_{mn}(t-2)} = \sum_{i \in A \cup B} \left[w_{ji}(t-2) \frac{\partial u_i(t-2)}{\partial w_{mn}(t-2)} + \frac{\partial w_{ji}(t-2)}{\partial w_{mn}(t-2)} u_i(t-2) \right]$$

當 $j=m$ 且 $i=n$ 時， $w_{ji}(t-2)$ 對 $w_{mn}(t-2)$ 的偏微分為 1，否則為 0。故上式可改寫為：

$$\frac{\partial net_j(t-1)}{\partial w_{mn}(t-2)} = \sum_{i \in A \cup B} w_{ji}(t-2) \frac{\partial u_i(t-2)}{\partial w_{mn}(t-2)} + \delta_{mj} u_n(t-2)$$

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

$$\frac{\partial u_i(t-2)}{\partial w_{mn}(t-2)} = \begin{cases} 0 & \text{if } i \in A \\ \frac{\partial y_j(t-2)}{\partial w_{mn}(t-2)} & \text{if } i \in B \end{cases}$$

因此 y_j 對權重 $w_{mn}(t-2)$ 的偏微分可改寫為：

$$\frac{\partial y_j(t-1)}{\partial w_{mn}(t-2)} = f'(net_j(t-1)) \left[\sum_{i \in B} w_{ji}(t-2) \frac{\partial y_j(t-2)}{\partial w_{mn}(t-2)} + \delta_{mj} u_n(t-2) \right]$$

對於這個遞迴式的初始條件是假設在 $t=0$ 的時候 Y 與 W 是互相獨立。因此：

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

接著我們假設

$$\frac{\partial y_j(t-1)}{\partial w_{mn}(t-1)} \approx \frac{\partial y_j(t-1)}{\partial w_{mn}(t-2)}$$

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

$$\pi_{mn}^j(t-1) = \frac{\partial y_j(t-1)}{\partial w_{mn}(t-1)} \quad \text{for all } j \in B, m \in B,$$

$$n \in A \cup B$$

由上我們可重新改寫遞迴式為：

$$\pi_{mn}^j(t-1) = f'(net_j(t-1)) \left[\sum_{i \in B} w_{ji}(t-2) \pi_{mi}^j(t-2) + \delta_{mj} u_n(t-2) \right]$$

又初始條件為 $\pi_{mn}^j(0) = 0$ ；故權重 $w_{mn}(t-2)$ 的修正量最後可寫為：

$$\Delta w_{mn}(t) = \eta_2 \left[\sum_{j=1}^J e_k(t) f'(net_k(t)) v_{kj}(t-1) \pi_{mn}^j(t-1) \right]$$

二階段 RTRL 乃是針對預測下二時刻所推導之演算法，與一階段 RTRL，主要的差別在於網路的傳遞上，一階段 RTRL 在輸出層的輸出僅做一非線性的轉換，而在隱藏層傳入輸出層時，維持相同的時刻，這中間亦屬於一線性之過程；本研究發展的二階段 RTRL 網路架構，不論是輸入層至隱藏層亦或是隱藏層至輸出層皆在時間上作進一步的轉換，整個架構仍符合非線性之時變歷程，藉此期能進一步預測下二小時之流量。

五、二階段回饋式類神經網路於即時流量之推估

大甲溪為台灣本島主要河川之一，位於台中縣，發源於中央山脈的雪山及南湖大山等群嶺之間，其間的溪谷有寬有窄，溪流有緩有急，變化多端。而流程一百多公里，流域面積為一二四四平方公里，年平均流量約三十一立方公尺/秒，河川坡

度陡急，平均坡降為百分之二點六，由於落差極大，水量均勻且豐沛致使其單位河長蘊藏水力高居全省河川首位，為台灣水質源最豐富的河川，亦為水力發電的重心。而德基水庫為一多目標水庫系統，又位於大甲溪水庫串聯系統之首，因此，準確的即時流量預測將對水庫操作、防洪與發電有很大的貢獻。本節中將以二階段即時回饋類神經網路預測德基水庫上游松茂流量站之時流量，同時與 ARMAX 模式做一比較，研究資料包括 1994~1997 年佳陽、平岩山、松峰、松茂四個雨量站之時雨量與松茂流量站之時流量，其相關地理位置之分佈圖如圖 2 所示。



圖 2 德基水庫上游集水區水文站分佈圖

本模式所建構網路有五個輸入值及一個輸出值；並以 1994 年各站 8760 筆的時雨量及時流量資料來檢定網路結構，選定處理層的神經元個數為 5，學習速率介於 10~30 之間；輸入值為這一小時志佳陽、平岩山、松茂、松峰四個雨量站的雨量及松茂流量站的流量，輸出值為下二小時松茂流量站的流量，其架構圖如圖 3 所示。

在 ARMAX 模式階次的選擇上，則以 AIC 值最小為準則，本研究將 1994 年前 1000 筆資料做為 ARMAX 的訓練階段；在不同的階次組合測試中，以 ARMAX(2, 2; 2) 之 AIC 值最小，其模式如下式：

$$\begin{aligned} y(t) = & a_1 y(t-1) + a_2 y(t-2) + b_1 u_1(t-2) \\ & + b_2 u_2(t-2) + b_3 u_3(t-2) + b_4 u_4(t-2) \\ & + c_1 e(t) + c_2 e(t-1) + c_3 e(t-2) \end{aligned}$$

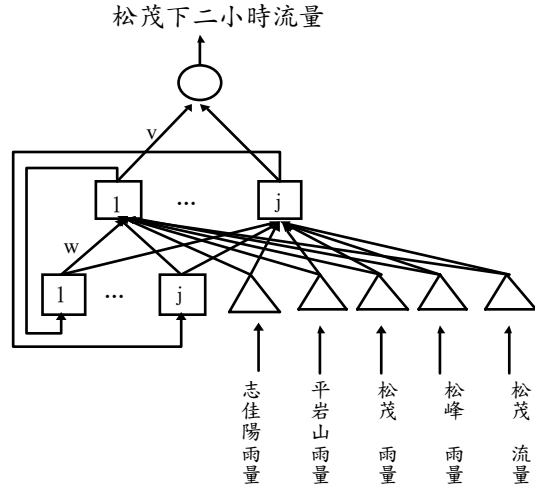


圖 3 一個輸出-五個輸入之回饋式類神經網路架構

比較以上兩種模式對預測下二小時即時流量的結果，除了以 MAE (Mean absolute error) 值為比較的標準外，另外加入正確率 RMAE (Relative mean absolute error) 值與洪峰流量作為比較。

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

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

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， Q_p 為相對於真實洪峰流量位置之推估值 (意即在每年最大時流量發生時刻之推估值)。

由表 1 可看出二階段 RTRL 模式對於 1994 年~1997 年下二時刻之時流量的推估非常準確；MAE 值比之於 ARMAX 皆有較佳的模擬；明顯地，ARMAX 模式的推估結果都較二階段 RTRL 模式為差。再以 Q_p 值為例，ARMAX 模式對 1994~1997 年的洪峰流量推估都有低估的現象，而二階段 RTRL 模式對 1994~1997 年洪峰流量推估誤差較小。由圖 4 及圖 5 可看出二階段 RTRL 推估之流量與真實流量皆非常接近。另外比較兩種模式在洪峰發生處附近產生時間延遲的現象，在 1996 及 1997 年間分別由圖 6、圖 7 表示，其中每年之洪峰發生時間標示於圖下方，由圖中可看出 ARMAX 模式推估結果產生明顯的二小時延遲的現象，也就導致其洪峰流量推估值比

表 1 ARMAX(2,2;2)與二階段 RTRL 推估松茂站 1994~1997 年之時流量比較

年份	平均 流量	尖峰 流量	ARMAX(2,2;2)			二階段 RTRL		
			MAE	RMAE	Q _p	MAE	RMAE	Q _p
1994	34.66	1120	1.464	0.047	695	1.298	0.037	734
1995	19.01	105	0.434	0.024	96	0.388	0.020	96
1996	24.04	1090	0.850	0.035	987	0.679	0.028	1073
1997	24.04	798	0.873	0.036	687	0.755	0.032	738

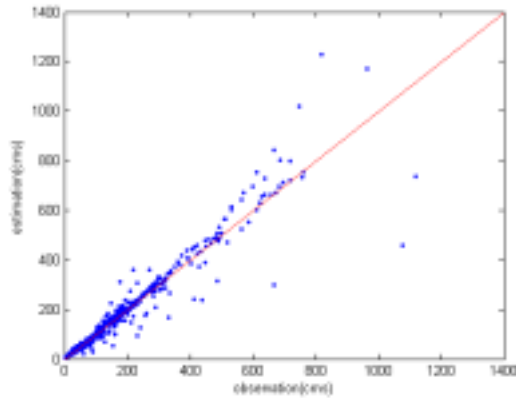


圖 4 1994 年松茂站洪峰時期推估流量與真實流量之比較圖

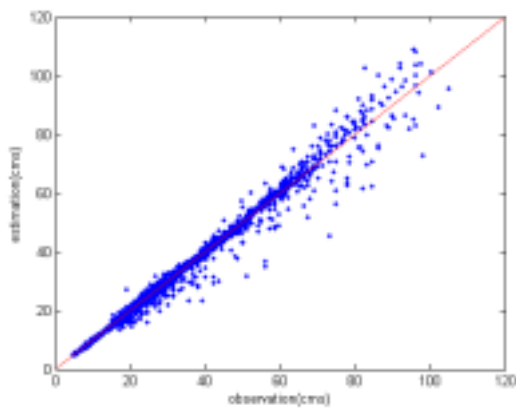


圖 5 1995 年松茂站洪峰時期推估流量與真實流量之比較圖

真值低估甚多，而二階段 RTRL 模式則沒有明顯的時間延遲的現象。因此，綜合以上，可以得知二階段 RTRL 在預測下二小時之即時流量的能力，明顯的優於 ARMAX，而且準確度亦相當高。

六、結 論

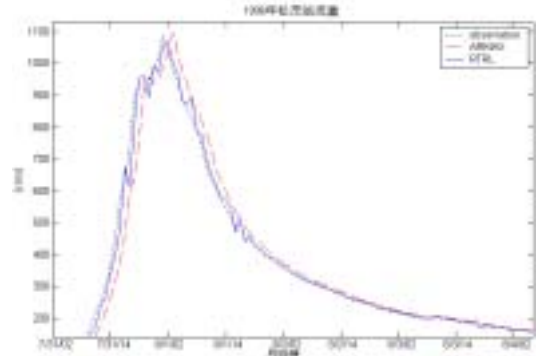


圖 6 1996 年松茂站洪峰時期推估流量與真實流量之比較圖

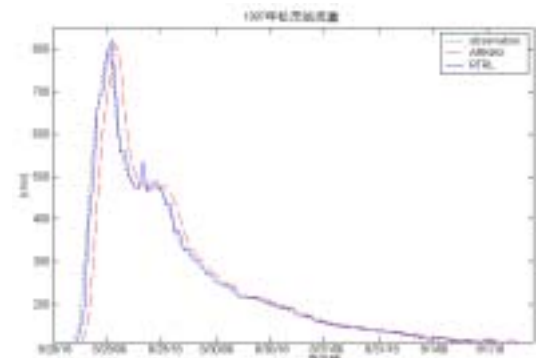


圖 7 1997 年松茂站洪峰時期推估流量與真實流量之比較圖

本研究所推演之二階段 RTRL 演算法除了保有原 RTRL 模式可以不斷的更新網路參數來學習不同的情況，適於水文系統中的時變性特徵外，更可以進一步地往前推估，提供更有效且精確的推估，對於未來防洪預警的工作有更進一步的實用性。

本研究利用大甲溪流域德基水庫上游松茂水文站的下二小時的流量推估，證實二階段 RTRL 模式是有效的演算法來模擬水文系統。針對歷年之降雨—

逕流資料，進行下二小時的流量推估，其平均絕對誤差(MAE)小於 0.8 cms，而相對平均誤差(RMAE)則非常接近零(小於 2.5%)，相較於 ARMAX 模式，顯示此模式可有效且精確的推估與預測松茂水文站下二期的流量。

RNN 的網路架構有別於傳統多層前向式(MLP)的網路架構，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，比起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。在結果中亦可以證實二階段 RTRL 模式演算法具有優異的 On-Line 線上即時訓練能力，可適用於缺乏歷史資料的地區。但 RNN 只有在處理層作回饋的連結，來儲存過去資訊。未來的研究可將輸出層亦做回饋的連結，如此可以儲存更多過去的訊息，期能有較佳的結果。

參 考 文 獻

1. 陳昶憲、楊朝仲、王益文，1996，「類神經網路於烏溪流域洪流預報之應用」，中華水土保持學報，vol.27，no.4。
2. 張斐章、胡湘帆，1998，「反傳遞模糊類神經網路於流量推估之應用」，中國農工學報，vol.44，No.2，pp.26-38。
3. 張斐章、孫建平，1997，「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，vol.43，No.1，pp.9-25。
4. 張斐章、黃浩倫、張麗秋，2001，「回饋式類神經網路於河川流量推估之應用」，農業工程學報，vol.47，No.2，pp.32-39。
5. 葉怡成，1993，類神經網路模式應用與實作，儒林圖書有限公司，台北。
6. Atiya, A.F., and S.I. Shaheen, 1999, "A Comparison Between Neural-Network Forecasting Techniques-Case Study: River Flow Forecasting," IEEE Trans. on NN, vol.10, pp.402-409.
7. Catfolis, T., 1993, "A Method for Improving the Real-Time Recurrent Learning Algorithm," Neural Network, vol.6, no.6, pp.807-821.
8. Chang, F.J., J.M. Liang, and Y.C. Chen, 2001, "Flood Forecasting Using Radial Basis Function Neural Network," IEEE Trans. On System, Man and Cybernetics Part C: Appl. & Rev.
9. Chang, F.J., L.C. Chang, and H.L. Huang, 2001, "Real Time Recurrent Neural Network for Stream flow Forecasting," Hydrological Processes, vol.15.
10. Chang, F.J., and Y.C. Chen, 2001, "A counterpropagation fuzzy-neural network modeling approach to real time stream flow prediction," Journal of Hydrology, vol.245, pp.153-164.
11. Elman, J.L., and D. Zipser, 1988, "Learning the Hidden Structure of Speech," Journal of the Acoustical Society of America, 83(4), pp.1615-1626.
12. Haykin, S. 1999, Neural Network: A comprehensive foundation. 2nd ed. Prentice Hall.
13. Hopfield, J.J., 1984, "Neurons with graded response have collective computational properties like those of two-state neurons," Proc. Nat. Sci. vol.81, pp.3088-3092.
14. Lee, C. G., G.M. Kuhn, and R.J. Williams, 1994, "Dynamic recurrent neural network: theory and applications," IEEE Trans. on NN, vol.5, pp.153-155.
15. McCulloch, W.S., and W. Pitts, 1943, "A logical calculus of the ideas immanent in nervous activity," Bulletin of Mathematical Biophysics, vol.5, pp.115-133.
16. Williams, R.J., and D. Zipser, 1989, "A learning algorithm for continually running fully recurrent neural network," Neural Computation, vol.1, pp.270-280.

收稿日期：民國 91 年 4 月 9 日

接受日期：民國 91 年 5 月 24 日