

การพยากรณ์สภาพน้ำท่าในลุ่มน้ำลำภาชีโดยระบบโครงข่ายประสาทประดิษฐ์

Runoff Forecasting in Lum Phachi River Basin by Artificial Neural Networks

ทรงศักดิ์ ภัทราวุธนชัย และวารวุธ วุฒิวนิชย์¹

(Songsak Puttrawutichai and Varawoot Vudhivanich)¹

บทคัดย่อ

ระบบโครงข่ายประสาทประดิษฐ์ สามารถนำมาใช้ในการทำนายปริมาณน้ำท่าโดยไม่จำเป็นต้องทราบวิธีทางกายภาพและอุทกวิทยาของการเกิดน้ำท่า ในการศึกษาครั้งนี้ระบบโครงข่ายประสาทประดิษฐ์ซึ่งใช้ back propagation algorithm ในการเรียนรู้มาทำนายปริมาณน้ำท่าในลุ่มน้ำลำภาชี ซึ่งมีพื้นที่รับน้ำประมาณ 2590 ตารางกิโลเมตร อยู่ทางภาคตะวันตกของประเทศ โดยจะทำนายปริมาณน้ำท่าในวันถัดไปของสถานี K.59 (จ.กาญจนบุรี) ความแม่นยำในการทำนายจะประเมินประสิทธิภาพใน 2 ด้านคือ ด้านคุณภาพโดยการเปรียบเทียบกราฟน้ำท่าที่ทำกรวัดได้กับกราฟน้ำท่า ที่ได้จากการทำนายโดยแบบจำลอง ด้านปริมาณโดยดัชนีทางสถิติ ผลที่ได้จากการทดลองโดยใช้ข้อมูลรายวันระหว่าง เม.ย.2000-มี.ค.2001 พบว่าหากใช้ข้อมูลสุ่ม 60% มาใช้ฝึกอบรมและทดสอบโครงข่ายและใช้ข้อมูล 40% ที่เหลือสำหรับทดสอบการทำนาย แบบจำลอง 10-10-10-1 จะให้ความถูกต้องในการทำนายมากที่สุด แต่หากใช้ข้อมูลช่วงเดือนมิ.ย.2000 - เดือนมี.ค.2001 มาฝึกอบรมและทดสอบโครงข่าย แบบจำลอง 5-10-10-1 จะทำนายปริมาณน้ำท่าในช่วงเดือนเม.ย.-พ.ค. 2000 ได้ดีที่สุด

Abstract

Runoff forecasting was performed by using artificial neural networks (ANNs). Artificial neural networks can be used for flow prediction without need of physical and hydrological knowledge related to runoff occurrence. In this study, artificial neural networks with back propagation algorithm was used for runoff forecasting in the Lum Phachi River Basin, which has drainage area of 2590 km² and located in the western part of Thailand. Artificial neural networks were applied to one-day ahead forecasting of daily discharges at station K.59 (Kanchanaburi). The accuracy of runoff forecasting was evaluated by using statistical performance index and the comparison of forecasted hydrograph with the observed hydrograph. The result showed that when the 60% random set of data was used in training and testing the model, and used the remaining 40% of data for forecasting, the 10-10-10-1 model gave the highest accuracy. However when the continuing set of data during June 2000-March 2001 was used for training and testing, the 5-10-10-1 model gave the highest accuracy for runoff forecasting during April-May 2000.

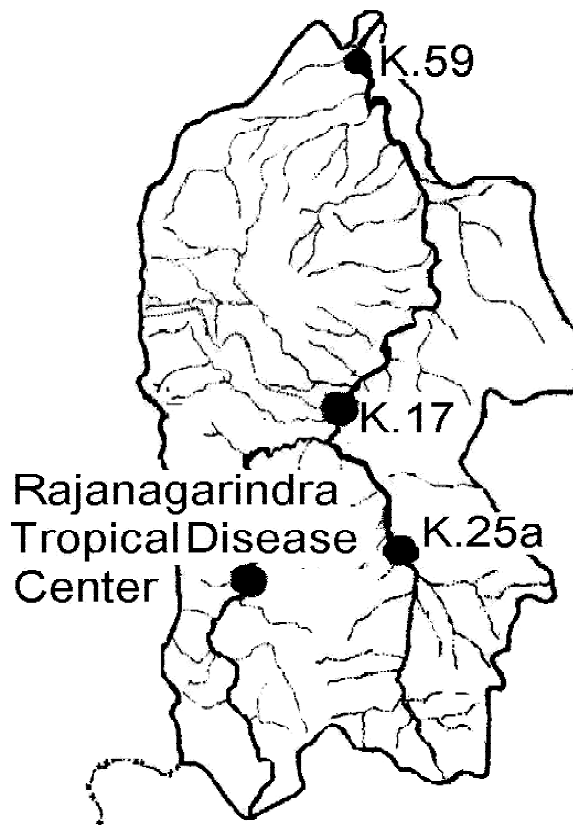
¹ ภาควิชาวิศวกรรมชลประทาน คณะวิศวกรรมศาสตร์ ม.เกษตรศาสตร์ วิทยาเขตกำแพงแสน นครปฐม 73140

Department of Irrigation Engineering, Faculty of Engineering, Kasetsart University, Kamphaengsean Campus, Nakornpathom. 73140

คำนำ

การศึกษาการพยากรณ์สภาพน้ำท่ามีความจำเป็น เนื่องจากทำให้ทราบสถานการณ์ของการเกิดน้ำท่าเมื่อเกิดมีฝนตกในปริมาณต่างๆ โดยจะนำไปสู่การวิเคราะห์หาปริมาณน้ำที่จะไหลสู่ลำน้ำหรือไหลลงสู่อ่างเก็บน้ำ สามารถวางแผนสนับสนุนกิจกรรมทางวิศวกรรม เพื่อเป็นประโยชน์ในการจัดการทรัพยากรน้ำ

ลุ่มน้ำลำภาชีเป็นลุ่มน้ำย่อยของลุ่มน้ำแม่กลองใหญ่ ตั้งอยู่ในเขต อำเภอสวนผึ้ง อำเภोजอมบึง จังหวัดราชบุรี และอำเภอด่านมะขามเตี้ย จังหวัดกาญจนบุรี มีพื้นที่ประมาณ 2,590 ตารางกิโลเมตร มีลำน้ำสายหลักคือ แม่น้ำภาชี ซึ่งมีต้นกำเนิดอยู่บริเวณเทือกเขาตะนาวศรี ในเขตอำเภอสวนผึ้ง ไหลย้อนขึ้นมาทางเหนือ ไปบรรจบกับแม่น้ำแควน้อย ที่บ้านท่าเสา จังหวัดกาญจนบุรี รวมความยาวทั้งสิ้นประมาณ 154 กิโลเมตรมีลำน้ำสาขาที่สำคัญคือ ห้วยลำขลุง ห้วยสีเสียด ห้วยอ้ายหลิว ห้วยบ่อคลึง ห้วยห้วยน้อย ห้วยท่าเคย และห้วยนายจอง



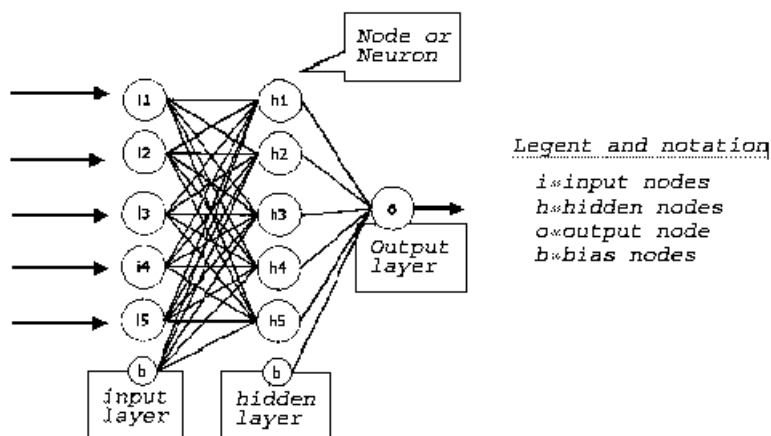
ภาพที่ 1 ลุ่มน้ำลำภาชี

การพยากรณ์สภาพน้ำท่าในลุ่มน้ำลำภาชี ที่จะทำการศึกษารั้งนี้ จะเลือกใช้แบบจำลองระบบโครงข่ายประสาทประดิษฐ์ มาจำลองสภาพการไหลของน้ำในบริเวณลุ่มน้ำลำภาชี

ระบบโครงข่ายประสาทประดิษฐ์

ระบบโครงข่ายประสาทประดิษฐ์ (Artificial Neural Networks, ANNs) ถูกนำมาใช้คาดการณ์และทำนายทางแหล่งน้ำกันอย่างมากโดยระบบโครงข่ายประสาทประดิษฐ์ ถูกจัดอยู่ในกลุ่มของ black box model ซึ่งไม่จำเป็นต้องให้ความสำคัญกับกระบวนการทางอุทกวิทยา และทางชลศาสตร์ ก็สามารถทำนายปริมาณน้ำได้อย่างแม่นยำ ต่างจากแบบจำลองกลุ่มอื่นที่ต้องอาศัยข้อมูลและความสัมพันธ์ของข้อมูลจำนวนมากมาใช้ (เสรี, 2544) ทำให้แบบจำลองในกลุ่ม black box model ถูกนำมาใช้กันมากขึ้น ในการทำนายการเกิดน้ำท่า

กระบวนการของระบบโครงข่ายประสาทประดิษฐ์ มีอยู่หลายวิธี แต่วิธี back propagation จะถูกนำมาใช้ให้เห็นในการใช้งานต่าง ๆ เสมอ back propagation (BP) ถูกพัฒนาโดย Rumelhart และ McClelland (1986 และ 1988) สำหรับเรียนรู้ความสัมพันธ์ระหว่าง input และ output ที่ใช้ layer มากกว่าชั้นเดียว ดังภาพที่ 2 โครงข่าย back propagation จะมีการเรียนรู้แบบต้องการครูช่วยสอน โดยเปรียบเทียบผลลัพธ์จากแบบจำลองที่กำลังเรียนรู้กับค่าที่ได้จากการวัดในสนาม และนำความแตกต่างที่ได้ไปปรับค่าถ่วงน้ำหนักย้อนกลับในแบบจำลองอีกที ข้อมูล input เดิมจะถูกป้อนสู่แบบจำลองอีกครั้งหนึ่ง และจะทำซ้ำ ๆ ทำให้ผลลัพธ์ที่ได้ใกล้เคียงกับความเป็นจริงมากที่สุด



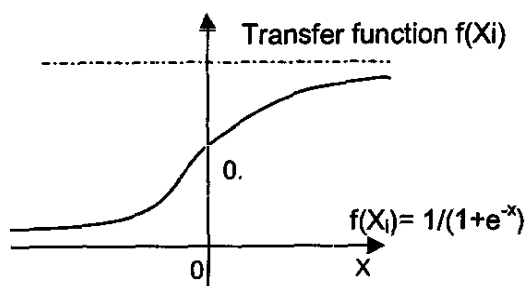
ภาพที่ 2 ตัวอย่างแบบจำลองโครงข่ายประสาทประดิษฐ์ 3 ชั้นแบบ 5-5-1

การทำงานของระบบโครงข่ายประสาทประดิษฐ์

node หรือเซลล์ประสาทในชั้น hidden layer จะถูกกำหนดคุณสมบัติ โดยการกระทำหน้าที่ 2 อย่างคือ

- Activation การรวมเอา input ที่ถูกถ่วงน้ำหนักแล้วมารวมให้เป็นผลลัพธ์
- Transfer การส่งข้อมูลให้เป็น output โดยใช้ฟังก์ชันที่เหมาะสม เช่น logistic, เชิงเส้น และแบบขั้นบันได

ความต้องการขั้นพื้นฐานของ BP training คือ การมีฟังก์ชันที่ใช้ส่งข้อมูลได้ต่อเนื่อง และแยกแยะได้อย่างชัดเจน ดังนั้น ฟังก์ชันแบบไม่เป็นเชิงเส้น sigmoid logistic จึงได้ถูกนำมาใช้ ฟังก์ชัน sigmoid จะให้ผลลัพธ์อยู่ระหว่าง 0 ถึง 1 ลักษณะของ Sigmoid function แสดงให้เห็นได้ดัง ภาพที่ 3



ภาพที่ 3 SIGMOID NON-LINEAR TRANSFER FUNCTION

กระบวนการ back propagation แบบมาตรฐานมีขั้นตอนดังนี้ (Tingsanchali,2544)

1. กำหนดค่า weights และ bias factors ขึ้นมาด้วยการสุ่มขึ้น
2. ป้อนค่า Input ($I_1, I_2, \dots, I_{n_0}$) และ Output ($t_1, t_2, \dots, t_{n_1}$) ที่ต้องการ
3. สำหรับ layer $m = 1, 2, \dots, l$ จะคำนวณค่า $N_{j,m}$ ของ neuron j ใน layer m ได้โดย

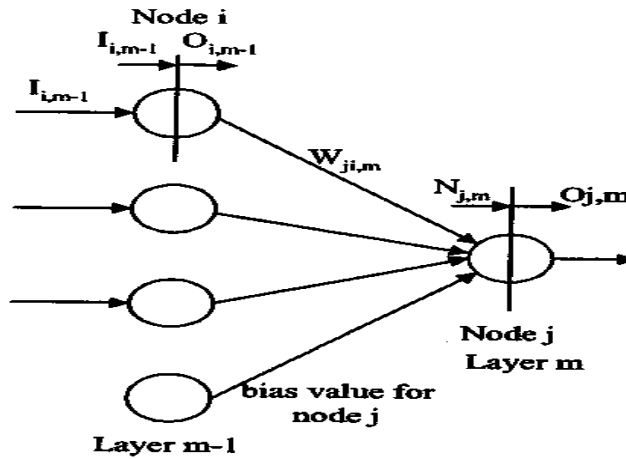
$$N_{j,m} = \sum_{i=1}^{n_{m-1}} W_{ji,m} O_{i,m-1} + \theta_{j,m} \quad (1)$$

เมื่อ $O_{i,0} = I_i$; t_j ค่าเป้าหมายของ neuron j ใน output layer; $O_{j,m}$ = output ของ neuron j ใน layer m ; $\theta_{j,m}$ = ค่า bias สำหรับ neuron j ใน layer m ; $W_{ji,m}$ = ค่า weight ระหว่าง node j ใน layer m และ node i ใน layer $m-1$; N_m = จำนวน neuron ใน layer m

คำนวณ output $O_{j,m}$ ของหน่วย j ใน layer m

$$O_{j,m} = \frac{1}{1 + e^{-N_{j,m}}}; j = 1, 2, \dots, n_m \quad (2)$$

4. คำนวณผลลัพธ์สุดท้าย ($O_{1,1}, O_{2,1}, \dots, O_{n_1,1}$) นำมาเปรียบเทียบกับผลลัพธ์ที่ต้องการ ($t_1, t_2, \dots, t_{n_1}$) ถ้าผลต่างอยู่ในเกณฑ์ที่ยอมรับได้ กระบวนการก็จะสิ้นสุด ระบบถือว่าได้เรียนรู้แล้ว แต่ถ้ายอมรับไม่ได้ก็ต้องทำต่อไปในขั้นต่อไป ถ้าทำแล้วหลายครั้งผลที่ได้ยังไม่อยู่ในเกณฑ์ที่ดีขึ้น ก็จะต้องมีการกำหนดค่าเริ่มต้นชุดใหม่ หรือมีการปรับปรุงโครงสร้างของโครงข่าย



ภาพที่ 4 การส่งผ่าน input จาก layer m-1 ไปเป็น output ของ layer m

5. การส่งย้อนกลับ สำหรับ layer $m = l, l-1, l-2, \dots, 1$ คำนวณได้จาก

สำหรับ output layer m

$$\delta_{j,m} = O_{j,m} (1 - O_{j,m}) (t_j - O_{j,m}) \quad (3a)$$

สำหรับ hidden layer m

$$\delta_{j,m} = O_{j,m} (1 - O_{j,m}) \sum_{k=1}^{n_{m+1}-1} W_{kj,m+1} \delta_{k,m+1} \quad (3b)$$

เมื่อ $\delta_{j,m}$ = ค่าของ δ สำหรับ neuron j ใน layer m

แล้วคำนวณค่าปรับแก้ของ weight

$$\Delta W_{ji,m}(n+1) = \eta \cdot \delta_{j,m} O_{i,m+1} + \alpha \cdot \Delta W_{ji,m}(n) \quad (4)$$

เมื่อ η = learning parameter; α = momentum constant; $\Delta W_{ji,m}(n)$ = ค่าปรับแก้ weight ระหว่าง node j ใน layer m และ node i ณ.รอบการเรียนรู้ที่ n; $\Delta W_{ji,m}(n+1)$ = ค่าปรับแก้ weight ระหว่าง node j ใน layer m และ node i ณ.รอบการเรียนรู้ที่ n+1; $O_{j,m} = l_{i,m} =$ รอบการเรียนรู้ (n = 1, 2, 3,...)

คำนวณค่าใหม่ของ weights โดย

$$W_{ji,m}(n+1) = W_{ji,m}(n) + \Delta W_{ji,m}(n+1) \quad (5)$$

เมื่อ $W_{ji,m}(n)$ = ค่า weight ระหว่าง node j ใน layer m และ node i ณ.รอบการเรียนรู้ที่ n

$W_{ji,m}(n+1)$ = ค่า weight ระหว่าง node j ใน layer m และ node i ณ.รอบการเรียนรู้ที่ n+1

6. ย้อนกลับไปทำซ้ำตั้งแต่ขั้นตอนที่ 2 จนกว่าโครงข่ายมีประสิทธิภาพเป็นที่พอใจ

ในโครงข่ายประสาทประดิษฐ์นี้ output node จะแสดงค่าของระดับน้ำที่ถูกทำนายขึ้น hidden nodes จะเป็นส่วนที่เรียนรู้ความสัมพันธ์แบบไม่เป็นเชิงเส้นของ input และ output พารามิเตอร์ที่เกิดขึ้นในการทำนาย จะขึ้นอยู่กับรูปแบบของฟังก์ชันที่เลือก Tingsanchali (2544) กล่าวว่าระยะเวลาการเรียนรู้จะขึ้นตรงกับขนาดของโครงข่าย หากเลือกโครงข่ายขนาดใหญ่ก็จะต้องใช้เวลาการเรียนรู้มากด้วย กระบวนการเรียนรู้ขึ้นกับ จำนวนของหน่วย input จำนวนของ hidden units และเงื่อนไขของการลู่เข้า การเลือก node ใน hidden layer มากไปก็จะทำให้การเรียนรู้เกิดความยุ่งยาก แต่ถ้าเลือกน้อยเกินไปก็จะทำให้การเรียนรู้ไม่ประสบความสำเร็จ

ค่า weights ที่ถูกนำสู่ output node และ hidden node จะถูกปรับแก้ย้อนกลับทำให้ความคลาดเคลื่อนลดลง เมื่อมีการกระทำซ้ำ ๆ ค่าตอบของระบบก็จะลู่เข้าหาเกณฑ์ที่ยอมรับได้

จากการที่ BPNN Model ได้เลือกใช้ฟังก์ชัน Logistic แบบ Sigmoid ซึ่งให้ผลการคำนวณอยู่ในช่วง $[0, 1]$ จึงต้องแปลงค่าข้อมูลที่มีให้อยู่ในช่วง $[0.05, 0.95]$ ก่อนที่จะป้อนสู่โครงข่าย การที่เลือกช่วง $[0.05, 0.95]$ เนื่องจากฟังก์ชัน Sigmoid มีค่าไม่สิ้นสุดดังภาพที่ 3 จึงไม่ควรใช้ช่วง $[0, 1]$ จึงต้องมีการแปลงข้อมูลโดยใช้สมการ

$$X_{t'} = \frac{0.9(X_t - x_{\min})}{x_{\max} - x_{\min}} + 0.05 \quad (6)$$

การแปลงค่าย้อนกลับเป็นค่าเดิมทำได้โดยใช้สมการ

$$X_t = \frac{(x_{\max} - x_{\min})(X_{t'} - 0.05)}{0.9} + x_{\min} \quad (7)$$

ประสิทธิภาพของแบบจำลอง

แบบจำลองโครงข่ายประสาทประดิษฐ์มีการประเมินประสิทธิภาพ 2 ด้านคือ

- ด้านคุณภาพ โดยการเปรียบเทียบไฮโดรกราฟของน้ำท่าจากการวัด และจากการทดลอง
- ด้านปริมาณ โดยใช้ดัชนีทางสถิติ

Nash และ Sutcliffe(1970) ได้เสนอวิธีการคำนวณประสิทธิภาพของ Nash (Nash Efficiency : Nash Eff) เพื่อเป็นดัชนีที่ประเมินประสิทธิภาพของแบบจำลอง ตามสมการ

$$\text{Nash Eff} = \frac{F_0^2 - F^2}{F_0^2} \quad (8)$$

เมื่อ $F_0^2 = \text{initial variance} \left(\sum (x_o - \overline{x_0})^2 \right)$; $F^2 = \text{residual variance} \left(\sum (x_o - x_c)^2 \right)$

อุปกรณ์และวิธีการ

ในการศึกษาครั้งนี้จะใช้แบบจำลองระบบโครงข่ายประสาทประดิษฐ์ชื่อว่า Qnet2000 (trial version) โดยมีวิธีการทดลองการจำลองเหตุการณ์โดยแบบจำลองดังนี้

1. การเตรียมข้อมูล ใช้ข้อมูลน้ำฝน-น้ำท่า สถานี K.17 K.25a และ K.59 ข้อมูลน้ำฝนจาก สถานีวิจัยโรคมมาเลเรียในเขตร้อน เดือนเมษายน 2000 ถึง เดือนมิถุนายน 2001 แบ่งข้อมูลออกเป็น 2 ชุด ด้วย 2 วิธีการคือ

วิธีที่ 1 โดยการสุ่มตัวอย่าง (Random) เพื่อใช้ข้อมูล 60% ของทั้งหมดเป็นชุดที่ 1 ในการฝึกสอนและสอบเทียบ และข้อมูลที่เหลืออีก 40 % เป็นชุดที่ 2 ในการทำนายเพื่อเปรียบเทียบกับข้อมูลที่วัดได้ในสนาม

วิธีที่ 2 โดยการตัดชุดข้อมูลเป็น 2 ช่วงเวลาคือ ใช้ข้อมูลช่วงเดือนมิถุนายน 2000 ถึง มีนาคม 2001 ในการฝึกสอนและสอบเทียบ และใช้ข้อมูลเดือนเมษายนถึงพฤษภาคม 2000 ในการทำนายเพื่อเปรียบเทียบกับข้อมูลที่วัดได้ในสนาม

2. ทดลองหา Learning parameter และ Momentum constant ที่เหมาะสมเพื่อใช้ในแบบจำลองโดยการสมมติ Learning parameter 3 ค่า คือ 0.01, 0.03 และ 0.05 Momentum constant 3 ค่า คือ 0.3, 0.5 และ 0.8 จับคู่ทดสอบทั้ง 9 กรณี กับระบบโครงข่าย 5-5-1 หาค่าที่เหมาะสมที่สุดเพื่อนำมาใช้

3. การหาโครงสร้างที่เหมาะสมของระบบโครงข่าย โดยเลือกจำนวน Input nodes 2 กรณี ตามตารางที่ 1 เพื่อทำนายปริมาณน้ำท่าของสถานี K.59 ในวันถัดไป ($Q_{59}(t+1)$) ประเมินประสิทธิภาพทั้งด้านคุณภาพและปริมาณ

ตารางที่ 1 ข้อมูลที่ใช้เป็น Input Nodes เพื่อทำนายปริมาณน้ำ

Input layer	Data for Input Nodes
5 Nodes	P25a(t), P17(t), Prtd(t), Q17(t), Q59(t)
10 Nodes	P25a(t), P25a(t-1), P17(t), P17(t-1), Prtd(t), Prtd(t-1), Q17(t), Q17(t-1), Q59(t), Q59(t-1)

*ตัวอย่างการเรียกชื่อข้อมูลเช่น

Prtd(t) = ปริมาณน้ำฝนที่วัดจาก Rajanagarindra tropical disease center ณ. วันที่ t

Q17(t) = ปริมาณน้ำท่าที่วัดจากสถานี K.17 ณ. วันที่ t

ผลการศึกษาและวิจารณ์

Learning parameter และ Momentum constant ที่เหมาะสม

จากตารางที่2 และ ตารางที่3 สรุปว่าทุกทุก Learning parameter และ Momentum constant ที่ใช้ให้ผลที่ผู้เข้าหาคำตอบจึงเลือกใช้ Learning parameter เท่ากับ 0.05 และ Momentum constant เท่ากับ 0.8 เพื่อระบบโครงข่ายใช้เวลาฝึกฝนน้อยที่สุด

ตารางที่2 Nash Efficiency จาก Learning parameter และ Momentum constant ต่างๆ

Learning parameter	Momentum		
	0.3	0.5	0.8
0.01	0.861	0.862	0.846
0.03	0.854	0.851	0.847
0.05	0.841	0.851	0.846

ตารางที่3 จำนวนรอบที่ใช้ฝึกฝน จาก Learning parameter และ Momentum constant ต่างๆ

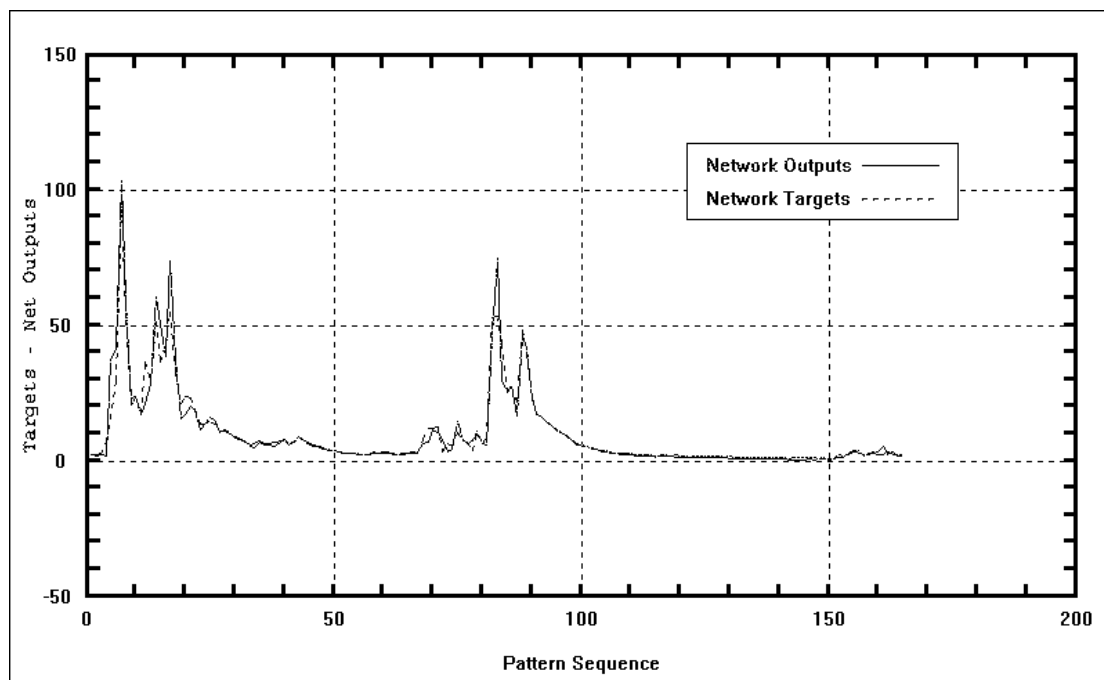
Learning parameter	Momentum		
	0.3	0.5	0.8
0.01	24310	19743	21046
0.03	8759	5427	7063
0.05	6249	6183	1565

โครงสร้างที่เหมาะสมของระบบโครงข่าย

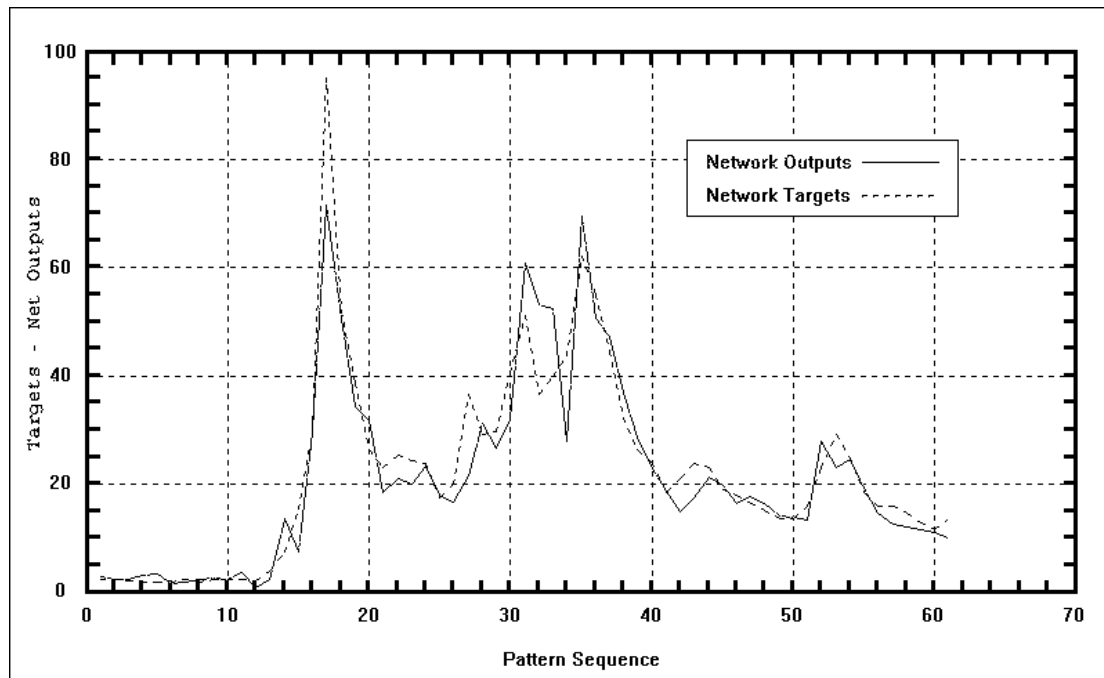
จากการทดลองพบว่าหากใช้ข้อมูลโดยการสุ่มตัวอย่าง (Random) ในการฝึกสอน สอบเทียบ และทำนายเพื่อเปรียบเทียบกับข้อมูลที่วัดได้ในสนาม ระบบโครงข่ายแบบ 10-10-10-1 ให้ประสิทธิภาพของ Nash (Nash Efficiency) ดีที่สุดและมีกราฟน้ำท่าที่ทำนายใกล้เคียงกับกราฟน้ำท่าที่ได้จากการวัดในสนามที่สุด โดยคำนวณค่า Nash Eff = 0.916 แต่หากใช้ข้อมูลเดือนมิ.ย.2000-มี.ค.2001มาทำนายปริมาณน้ำท่าเดือนเม.ย.-พ.ค.2000 โครงข่ายแบบ 5-10-10-1 ให้ประสิทธิภาพของ Nash ดีที่สุดและมีกราฟน้ำท่าที่ทำนายใกล้เคียงกับกราฟน้ำท่าที่ได้จากการวัดในสนามที่สุด โดยคำนวณค่า Nash Eff = 0.884

ตารางที่ 4 ประสิทธิภาพของ Nash (Nash Efficiency : Nash Eff) ที่คำนวณได้จากโครงสร้างของระบบ
โครงข่ายต่างๆ

Random Data		June 2000-March 2001 for runoff forecasting during April-May 2000.	
Neural structure	Nash Eff	Neural structure	Nash Eff
5-5-1	0.914	5-5-1	0.842
5-10-1	0.883	5-10-1	0.832
5-5-5-1	0.866	5-5-5-1	0.884
5-10-10-1	0.905	5-10-10-1	0.884**
5-5-5-5-1	0.888	5-5-5-5-1	0.825
5-10-10-10-1	0.895	5-10-10-10-1	0.786
10-5-1	0.884	10-5-1	0.749
10-10-1	0.902	10-10-1	0.877
10-5-5-1	0.913	10-5-5-1	0.713
10-10-10-1	0.916**	10-10-10-1	0.501
10-5-5-5-1	0.909	10-5-5-5-1	0.596
10-10-10-10-1	0.912	10-10-10-10-1	0.671



ภาพที่ 5 กราฟน้ำท่าเปรียบเทียบระหว่างกราฟน้ำท่าที่ทำนายจากโครงข่าย 10-10-10-1 กับกราฟน้ำท่าที่ได้จากการวัดในสนามเมื่อใช้ข้อมูลแบบสุ่ม



ภาพที่ 6 กราฟน้ำท่าเปรียบเทียบระหว่างกราฟน้ำท่าที่ทำนายจากโครงข่าย 5-10-10-1 กับกราฟน้ำท่าที่ได้จากการวัดในสนามเมื่อใช้ข้อมูลเดือนมิ.ย.2000-มี.ค.2001มาทำนายปริมาณน้ำท่าเดือนเม.ย.-พ.ค.2000

สรุปผลการทดลอง

ผลที่ได้จากการทดลองโดยใช้ข้อมูลรายวันระหว่าง เม.ย.2000-มี.ค.2001 พบว่าหากใช้ข้อมูลส่วน 60% มาใช้ฝึกอบรมและทดสอบโครงข่ายและใช้ข้อมูล 40% ที่เหลือสำหรับทดสอบการทำนายแบบจำลอง 10-10-10-1 จะให้ความถูกต้องในการทำนายมากที่สุด แต่หากใช้ข้อมูลช่วงเดือนมิ.ย.2000 - เดือนมี.ค.2001 มาฝึกอบรมและทดสอบโครงข่าย แบบจำลอง 5-10-10-1 จะทำนายปริมาณน้ำในช่วงเดือนเม.ย.-พ.ค.2000 ได้ดีที่สุด สามารถสรุปได้ว่าโครงข่ายประสาทประดิษฐ์ที่ใช้ back propagation algorithm สามารถนำมาใช้ทำนายปริมาณน้ำท่าในลุ่มน้ำลำภาชีได้อย่างแม่นยำ โดยไม่จำเป็นต้องใช้ข้อมูลทางกายภาพและกระบวนการทางอุทกวิทยา เหมาะที่จะนำมาใช้ทำนายปริมาณน้ำที่จะไหลสู่ลำน้ำหรือไหลลงสู่อ่างเก็บน้ำ สามารถวางแผนสนับสนุนกิจกรรมทางวิศวกรรม เพื่อเป็นประโยชน์ในการจัดการทรัพยากรน้ำและมีประโยชน์ในการเตรียมรับมือและเตือนภัยการเกิดน้ำท่วม

เอกสารอ้างอิง

เสรี สุภราทิตย์. 2544. คู่มือการพยากรณ์น้ำโดยโครงข่ายประสาทเทียม. คณะอนุกรรมการศึกษา

วิจัยด้าน บริหารจัดการน้ำ สถาบันพัฒนาการชลประทาน(IDI), นนทบุรี. 78น.

Nash, J.E. and Sutcliffe, J.V. 1970. River Flow Forecasting through Conceptual Models.

Journal of Hydrology. 10 : 282-290.

Tingsanchali, T. 2544. Back Propagation Network for Flood Forecasting. เอกสารการบรรยาย

พิเศษ "การประยุกต์ใช้ศาสตร์ทางปัญญาประดิษฐ์ในงานวิศวกรรมแหล่งน้ำ". วิศวกรรมสถานแห่งประเทศไทย, กรุงเทพฯ. 1-12