ELSEVIER

Journal of Hydrology 214 (1999) 32–48

Journal of Hydrology

Short term streamflow forecasting using artificial neural networks

Cameron M. Zealand, Donald H. Burn*, Slobodan P. Simonovic

Department of Civil and Geological Engineering, University of Manitoba, Winnipeg, MB, Canada R3T 5V6

Received 26 September 1997; received in revised form 26 May 1998; accepted 15 September 1998

Abstract

The research described in this article investigates the utility of Artificial Neural Networks (ANNs) for short term forecasting of streamflow. The work explores the capabilities of ANNs and compares the performance of this tool to conventional approaches used to forecast streamflow. Several issues associated with the use of an ANN are examined including the type of input data and the number, and the size of hidden layer(s) to be included in the network. Perceived strengths of ANNs are the capability for representing complex, non-linear relationships as well as being able to model interaction effects. The application of the ANN approach is to a portion of the Winnipeg River system in Northwest Ontario, Canada. Forecasting was conducted on a catchment area of approximately 20 000 km^2. using quarter monthly time intervals. The results were most promising. A very close fit was obtained during the calibration (training) phase and the ANNs developed consistently outperformed a conventional model during the verification (testing) phase for all of the four forecast lead-times. The average improvement in the root mean squared error (RMSE) for the 8 years of test data varied from 5 cms in the four time step ahead forecasts to 12.1 cms in the two time step ahead forecasts. © 1999 Elsevier Science B.V. All rights reserved.

Keywords: Forecasting; Streamflow; Artificial neural networks

1. Introduction

Many of the activities associated with the planning and operation of the components of a water resource system require forecasts of future events. For the hydrologic component, there is a need for both short term and long term forecasts of streamflow events in order to optimize the system or to plan for future expansion or reduction. Many of these systems are large in spatial extent and have a hydrometric data collection network that is very sparse. These conditions can result in considerable uncertainty in the hydrologic information that is available. Furthermore, the inherently non-linear relationships between input and output variables complicates attempts to forecast streamflow events. There is thus a need for improvement in forecasting techniques. Many of the techniques currently used in modelling hydrological time-series and generating synthetic streamflows assume linear relationships amongst the variables. The two main groups of techniques include physically based conceptual models and time-series models. Techniques in the first group are specifically designed to mathematically simulate the sub-processes and physical mechanisms that govern the hydrological cycle. These models usually incorporate simplified forms of physical laws and are generally non-linear, time-invariant, and deterministic, with parameters that are representative of watershed characteristics (Hsu et al., 1995) but ignore the spatially distributed, time-varying, and stochastic properties of the

* Corresponding author. Present address: Civil Engineering, University of Waterloo, Waterloo, ON, Canada N2L 3G1.
e-mail: dhburn@sunburn.uwaterloo.ca

0022-1694/99/$ - see front matter © 1999 Elsevier Science B.V. All rights reserved.
PII: S0022-1694(98)00242-X

rainfall–runoff (R–R) process. Kitanidis and Bras (1980a,b) state that conceptual watershed models are reliable in forecasting the most important features of the hydrograph. However, the implementation and calibration of such a model can typically present various difficulties (Duan et al., 1992), requiring sophisticated mathematical tools (Duan et al., 1992, 1994; Sorooshian et al., 1993), significant amounts of calibration data (Yapo et al., 1996), and some degree of expertise and experience with the model (Hsu et al., 1995). The problem with conceptual models is that empirical regularities or periodicities are not always evident and can often be masked by noise.

In time-series analysis, stochastic or time-series models are fitted to one or more of the time-series describing the system for purposes which include forecasting, generating synthetic sequences for use in simulation studies, and investigating and modelling the underlying characteristics of the system under study. Most of the time-series modelling procedures fall within the framework of multivariate autoregressive moving average (ARMA) models (Raman and Sunilkumar, 1995). Traditionally, the class of ARMA models have been the statistical method most widely used for modelling water resources time-series (Maier and Dandy, 1996). In streamflow forecasting, time-series models are used to describe the stochastic structure of the time sequence of streamflows and precipitation values measured over time. Time-series models are more practical than conceptual models because one is not required to understand the internal structure of the physical processes that are taking place in the system being modelled. The limitation of univariate time-series methods in streamflow forecasting is that the only information they incorporate is that which is present in past flows. Many of the available techniques are deficient in that they do not attempt to represent the non-linear dynamics inherent in the transformation of rainfall to runoff.

The main focus of this article is the development of Artificial Neural Network (ANN) models for short term streamflow forecasting, determining which characteristics of the model have the greatest impact on performance and deriving general methodologies for using these models for any catchment area. Comparisons are made between the performance of different ANN structures and a model based on a more traditional forecasting approach. Conventional models for streamflow forecasting typically involve a number of physical variables that function as inputs. A physical variable that is not very useful for forecasting on its own can often be useful when used in conjunction with other variables. Given the number of physical variables that could be considered as potentially relevant, it is apparent that a very large number of different combinations of both variables and mathematical relationships that link them together are available when developing a streamflow forecasting model. Determining an appropriate model structure by a trial-and-error process is therefore not always practical. In this context, the power of ANNs arises from the capability for constructing complicated indicators (non-linear models) for multivariate time-series.

ANNs have been successfully applied in a number of diverse fields including water resources. In order to optimally fit an ARMA-type model to a time-series, the data must be stationary and follow a normal distribution (Hipel, 1986). When developing ANN models, the statistical distribution of the data need not be known (Burke, 1991) and non-stationarities in the data, such as trends and seasonal variations, are implicitly accounted for by the internal structure of the ANNs (Maier and Dandy, 1996). ANNs differ from the traditional approaches in synthetic hydrology in the sense that they belong to a class of data-driven approaches. Data-driven approaches are suited to complex problems. They have the ability to determine which model inputs are critical, so that there is no need for prior knowledge about the relationships amongst the variables being modelled. ANNs are relatively insensitive to noisy data, unlike ARMA-type models, as they have the ability to determine the underlying relationship between model inputs and outputs, resulting in good generalization capability. Lorrai and Sechi (1995) verified the possibility of utilizing ANNs to predict R–R when only information about the variation of the basic input variables, namely rainfall and temperature, is available. Cheng and Noguchi (1996) obtained better results modelling the R–R process with ANNs using previous rainfall, soil moisture deficits, and runoff values as model inputs, when compared with that from a R–R model. Smith and Eli (1995) applied ANNs to convert remotely sensed, spatially distributed rainfall patterns into rainfall rates, and hence into runoff for a given

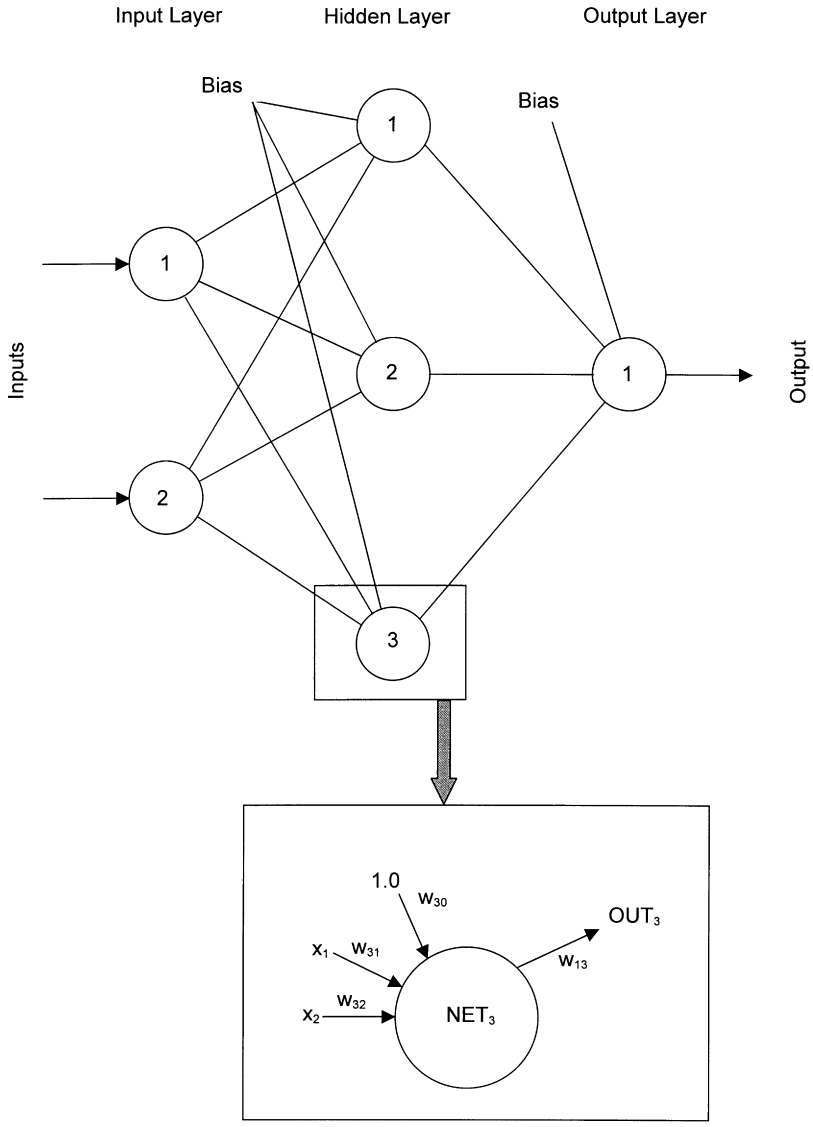

Fig. 1. Multi-layered feed-forward ANN and processing element architecture.

river basin. Hsu et al. (1995) showed that a non-linear ANN model provided a better representation of the R–R relationship of the medium-sized Leaf River basin near Collins, Mississippi, than the linear ARMAX (autoregressive moving average with exogenous inputs) time-series approach or the conceptual SAC-SMA (Sacramento soil moisture accounting) model. Karunanithi et al. (1994) used ANNs for river flow prediction and Raman and Sunilkumar (1995) for forecasting multivariate water resources time-series. ANNs have also been applied to areas such as deriving a general operating policy for reservoirs Raman and Chandramouli (1996), prediction of water quality parameters Maier and Dandy (1996) and real-time forecasting of water quality Dandy and Maier (1996).

In the next section, a brief review of ANNs is presented, including the differences between ANNs and more traditional forecasting methods. ANNs are then applied to forecasting streamflow into Namakan

Lake, located in Northwestern Ontario, Canada. The case study along with the development of the ANN model is described in Section 3. The results and conclusions of the study are given in Sections 4 and 5, respectively.

2. Artificial neural networks

2.1. ANN characteristics

ANN models attempt to achieve good performance through dense interconnections of simple computational elements. In this respect, the architectures of ANNs are based on the present understanding of the biological nervous systems. ANNs offer valuable characteristics unavailable together elsewhere. First, they infer solutions from data without prior knowledge of the regularities in the data; they extract the regularities empirically. Second, these networks learn the similarities among patterns directly from instances or examples of them. ANNs can modify their behavior in response to the environment (i.e. shown a set of inputs with corresponding desired outputs, they self-adjust to produce consistent responses). Third, ANNs can generalize from previous examples to new ones. Generalization is useful because real-world data are noisy, distorted, and often incomplete. Fourth, ANNs are also very good at the abstraction of essential characteristics from inputs containing irrelevant data. Fifth, they are non-linear, that is, they can solve some complex problems more accurately than linear techniques do. Finally, ANNs are highly parallel. They contain many identical, independent operations that can be executed simultaneously, often making them faster than alternative methods.

ANNs also have several drawbacks for some applications. Firstly, they may fail to produce a satisfactory solution, perhaps because there is no learnable function or because the data set is insufficient in size. Secondly, the optimum network geometry as well as the optimum internal network parameters are problem dependent and generally have to be found using a trial-and-error process. Finally, ANNs cannot cope with major changes in the system because they are trained (calibrated) on a historical data set and it is assumed that the relationship learned will be applicable in the future. If there were any major changes in the system, the neural network would have to be adjusted to the new process.

2.2. ANN architecture

Generally speaking, all ANNs are vector mapping functions. That is, they map one vector space to another. An input vector is applied to the network and in response, the network produces an output vector. Each vector consists of one or more components, each of which represents the value of some variable (e.g. precipitation, temperature, streamflow).

The architecture of a feed-forward ANN can have many layers where a layer represents a set of parallel nodes. A typical three-layer feed-forward ANN is shown in Fig. 1. It should be noted that the feed-forward ANN architecture is but one of many possible architectures. The feed-forward ANN is adapted herein due to its general applicability to a variety of different problems (Hsu et al., 1995). The first layer is called the input or passive, layer and consists of a set of processing elements (PEs) that connect with the input variable(s). The sole role of the input layer is to pass the input variables onto the subsequent layers of the network. The last layer connects to the output variable(s) and is called the output, or active, layer. The inputs to the neurons in hidden layer(s) and the output layer come exclusively from the outputs of neurons in previous layers, and outputs from these neurons pass exclusively to neurons in following layers. Layer(s) of PEs in-between the input and output layers are called hidden layers because they have no direct connection to the outside world, neither input nor output. Introducing this intermediate layer enhances the network's ability to model complex functions. Lippmann (1987) suggests that no more than three layers (one hidden layer) are required in feed-forward networks because a three-layer network can generate arbitrarily complex decision regions. The PEs in each layer are called nodes or units. The number of nodes in the input and output layers are dictated by the dimension of input and output vectors presented to the network for training. The number of hidden layers and hidden layer nodes is determined in the process of designing the network. Each connection has an associated adjustable parameter called a weight or connection strength. All of the hidden nodes receive all input data, but because each has a different

set of weights, the sets of values differ. The number of hidden nodes must be large enough to form a decision region that is as complex as is required by a given problem (Lippmann, 1987).

The architecture of a single PE or node is shown in Fig. 1. In general, a node can have n inputs, labeled from 1 through n. For the example node shown in Fig. 1, $n = 2$. In addition, each node has an input that is always equal to 1.0, called the *bias*. Each node j receives information from every node i in the previous layer. A weight (w_{ji}) is associated with each input (x_i) to node j. The effective incoming information (NET_j) to node j is the weighted sum of all incoming information, otherwise known as the net input

$$\mathrm{NET}_j = \sum_{i=0}^{n} w_{ji} x_i, \tag{1}$$

where x_0 and w_{j0} are called the bias term ($x_0 = 1.0$) and the bias weights respectively. Eq. (1) applies to the nodes in the output layer and in the hidden layer(s).

The transfer function, or squashing function, introduces a non-linearity that, when applied to the net input of a node, determines the output of that node. The effective input, NET_j, is passed through a transfer function to produce the outgoing value (OUT_j) of the node. Hsu et al. (1995) state that the most commonly used activation function is the steadily increasing S-shaped curve called a sigmoidal function. This function acts as a squashing function and compresses the range of NET so that OUT lies between 0 and 1. The sigmoidal function most often used for ANNs is the logistic function (Hsu et al., 1995), expressed mathematically as:

$$\mathrm{OUT}_j = \frac{1}{1 + \exp(-\mathrm{NET}_j)}. \tag{2}$$

The value of NET_j can vary between $\pm\infty$, but OUT_j is bounded between 0 and 1. Since the desired outputs of the network are not generally in the range from 0 to 1, the target data must be scaled appropriately prior to the training (calibration) process.

2.3. Network training

The main objective of training (calibrating) a network is to produce the desired set of outputs when a set of inputs is fed to the ANN. Training a network is a procedure during which an ANN processes a training set (input –output data pairs) repeatedly, changing the values of its weights, according to a predetermined algorithm, to improve its performance. It is important that the training set provide a full and accurate representation of the problem domain; otherwise the network will not meet expectations. Each pass through the training data is called an epoch and the ANN learns through the overall change in weights accumulated over many epochs. During training, the network weights gradually converge to values such that each input vector produces output values that are as close as possible to the desired output vector. Valluru and Hayagriva (1993) estimate that over 80% of all neural network projects in development use the backpropagation (BP) training algorithm. The BP algorithm gives a prescription for changing the weights, w_{ji}, in any feed-forward network to learn a training vector of input–output pairs. It is a supervised learning method in which an output error is fed back through the network, altering connection weights so as to minimize the error between the network output and the target output. During supervised training, the output predicted by the network is compared with the actual (desired, historical) output and the mean squared error (MSE) between the two is calculated. Further details on the BP algorithm can be found in Rumelhart et al. (1987).

Designing an ANN can be as simple as selecting a commercially available software package and configuring it to agree with the data or as complex as coding a fully custom network from scratch — an approach beyond the scope of this article. The intended research concentrates on the application of ANNs for streamflow forecasting. Braincel™ (Promised Land Technologies, 1993) is the software that was used to implement the ANNs in this research.

3. Application of the neural network

This section describes the study area and provides an overview of the conventional forecasting model that is used as a basis of comparison for the ANN model. The two forecasting approaches are compared first in an experiment where the ANN is provided with exactly the same inputs as are provided to the conventional forecasting model. Finally, a second experiment

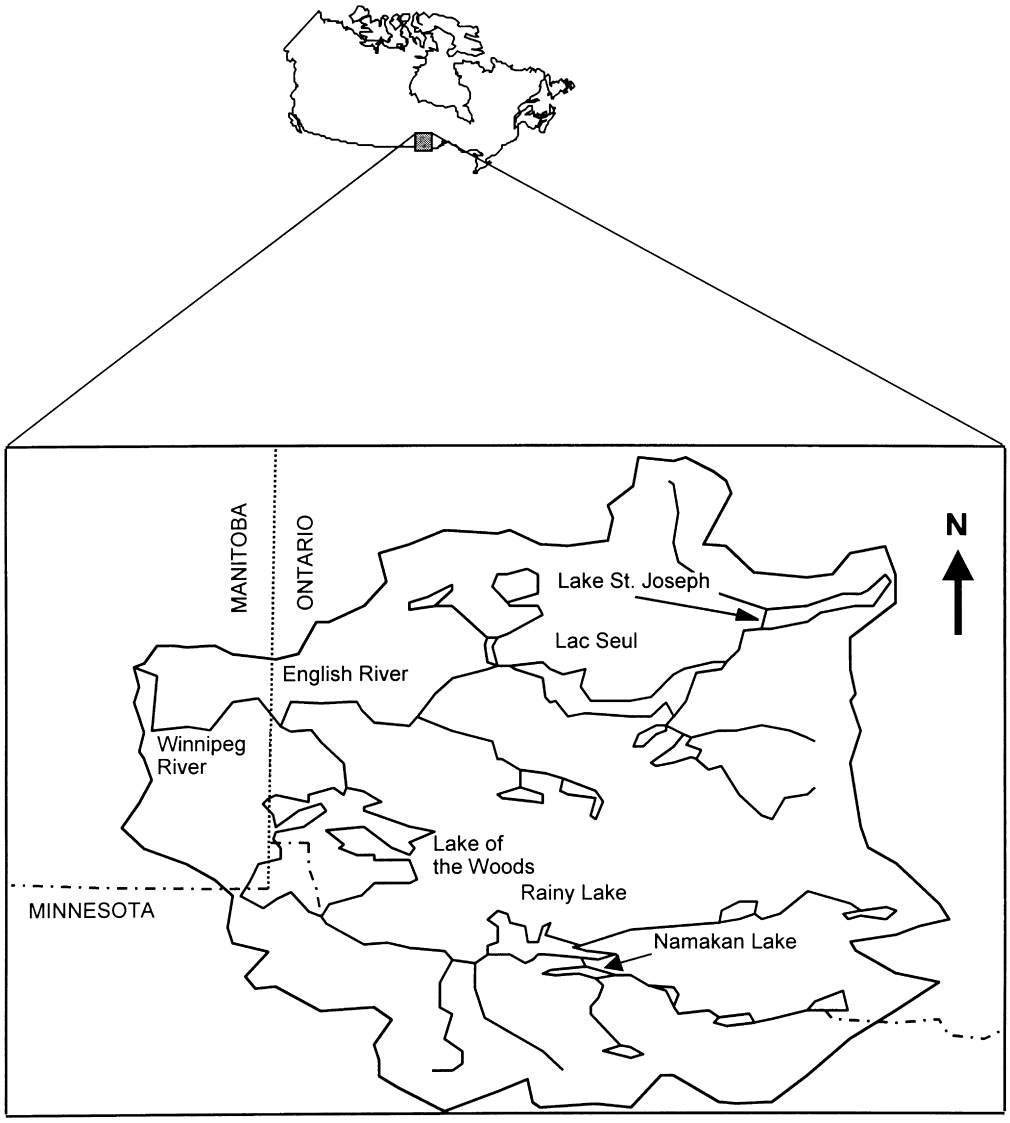

Fig. 2. Location map for the study area.

is presented in which an attempt is made to improve upon the performance of the ANN by altering the input information provided.

3.1. Description of study area

The majority of the Winnipeg River Basin is located in the Precambrian Shield region of Northwestern Ontario with parts of the watershed in Southeastern Manitoba and Minnesota (see Fig. 2). The Winnipeg River and its main tributary, the English River, constitute an enormous water resource, with a drainage area of approximately 150,000 km^2. (Acres International Ltd., 1993). There are five regulated lakes in the Winnipeg River Basin, namely Lake St. Joseph, Lac Seul, Namakan Lake, Rainy Lake and Lake of the Woods. The management of the Winnipeg River Basin system is particularly difficult due to interests in control of flooding, water based recreational activities, hydroelectric power generation,

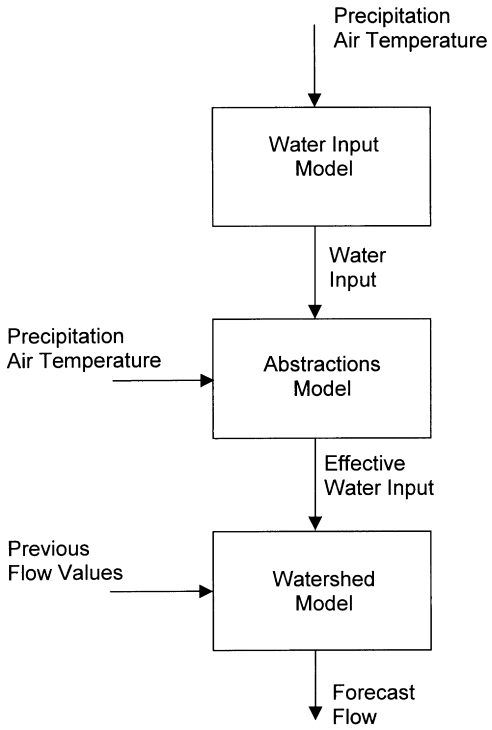

Fig. 3. Schematic diagram of the WIFFS model.

agriculture, and municipal and industrial water supply. Directing the operation of large lakes requires monitoring and forecasting basin conditions, planning regulation strategies, consulting with affected parties and providing public information. The most upstream lake in the watershed, Namakan Lake, was chosen as the test location in this case study. The Namakan Lake subwatershed has a drainage area of approximately 19 270 km^2. The catchment area receives an average precipitation of 78 cm, of which roughly 30% is snowfall, and the average annual runoff from the catchment is about 26 cm. There are 29 years of average daily precipitation and temperature data as well as average weekly streamflow values available from 1960 to 1988 for the Namakan Lake subwatershed.

3.2. Description of experiments

In the first experiment (experiment no.1), an ANN model was produced to give a fair comparison between the forecasting accuracy of the conventional model and that of the ANN technology. This was done by providing the ANN with the same inputs and information that were used in the development of the conventional model. The ANN model was then trained on the data from the period 1965 to 1985 that were used to calibrate the conventional model and tested on the data before 1965 and after 1985 that were set aside for model verification purposes LWCB (1994b). The intent with experiment no. 1 was to provide a fair and unbiased comparison between the two models. To the extent possible, all conditions for the two models were kept the same.

In the second experiment, an ANN was built that was not restricted to the input variables used in the conventional model. The purpose of the second experiment (experiment no. 2), was to build an 'optimum' ANN. This 'optimum' network involved a different set of inputs. This was done using some of the same inputs as in the conventional model and investigating the use of additional inputs. In this experiment, part of the data had to be eliminated from the data set. The reason for doing this was that it appeared that the actual measured data were missing. The missing data appeared to have been artificially filled-in for purposes of keeping the time-series continuous. Therefore, to obtain a proper comparison between the accuracy of the ANN model developed in experiment no. 1 and that developed in experiment no. 2, these data were eliminated and experiment no. 1 was repeated using the reduced data set. The results for the ANN model for experiment no. 2 are then compared with those from the ANN model for experiment no. 1 applied to the new data set.

3.3. Conventional forecasting model

The Winnipeg Flow Forecasting System (WIFFS) is a stochastic-deterministic watershed model developed by ACRES International of Niagara Falls, Ontario, to estimate the quarter-monthly natural flows into the major lakes in the system. The WIFFS model served as the comparison tool to the ANN method. This model contains three basic components: (i) the water input generation model; (ii) the abstraction or loss model; and (iii) the distribution or watershed model. The water input generation component includes a procedure to calculate rainfall and snowmelt water input to the watershed,

based on the meteorological data. This module uses a degree-day snowmelt model to determine the contributions from the snowpack. The abstractions model estimates the portion of the water input that does not infiltrate into the ground but rather becomes runoff. Abstractions are modelled using a variable runoff coefficient model. The watershed routing component of the model routes this effective water input through the watershed using a time-series model of the ARMAX type. The WIFFS model is presented in schematic form in Fig. 3. For a more detailed description of each of the three components of the WIFFS model, the reader is referred to Acres International Ltd. (1993).

The WIFFS model includes four time steps each month. The first time step includes 8 days, the second and third 7 days, and the fourth has the remaining days in the month ((LWCB, 1994a). This translates into 48 periods per year. For the remainder of the article, each time period will be referred to as a 'week', even though each quarter-month period is not comprised of seven days. The inputs to the WIFFS model include *Period* '*now*', $P(t)$, the period before the first forecast period, *Forecast lead-time*, the number of periods from *Period*'*now*' for which forecasts are calculated, *Historical Meteorological inputs* consisting of complete precipitation and temperature data for *Period* '*now*' and the previous six periods for Namakan Lake. The number of previous periods of meteorological data required may differ for each subwatershed. The final inputs are *Forecasted Meteorological inputs*, which provide an inference of future precipitation and temperature inputs. The outputs from the WIFFS model are *Streamflow* $(t + 1, 2, 3$ *or* $4)$, one, two, three or four week(s)-in-advance forecast of streamflow. For the one-time step ahead forecasts for the Namakan Lake watershed, WIFFS has a total of 17 parameters that must be determined through calibration. The objective function for the calibration of WIFFS is the minimization of the MSE between observed and forecasted flows. The same objective function is used in the training (calibrating) of the ANN models.

3.4. ANN model identification

In both experiments, a multivariate time-series model was derived for the output $Flow_{t+i}$; $i = 1, 2, 3$ and 4. The data used are total weekly precipitation, average weekly temperature and average weekly streamflow. In all cases, the networks were trained over a certain part of the data (1965–1985) and once training was complete, the networks were tested over the remaining data (1960–1964 and 1986–1988). Training and testing periods corresponded to the calibration and verification periods adopted for WIFFS.

3.4.1. Experiment no. 1

The input and output variables used in the ANN model were fixed to match those used in the WIFFS model. These inputs included precipitation, temperature and local inflow of the preceding weeks as well as rainfall and temperature of the examined week. For the one-week ahead forecast of flow, $Flow\ (t + 1)$, the model consisted of the following 18 input variables: past seven periods of weekly precipitation, $P(t)$, $P(t - 1)$, $\ldots P(t - 6)$, past seven periods of weekly temperature, $T(t)$, $T(t - 1)$, $\ldots$, $T(t - 6)$, past two periods of local inflow, $Flow(t)$ and $\mathrm{Flow}(t - 1)$, 1-week ahead inference of the average weekly precipitation, $P(t + 1) = 50$th percentile, and 1-week ahead inference of the average weekly temperature, $T(t + 1) = 50$th percentile. The 2-week ahead forecast included an additional three inputs [$P(t + 2) =$ 50th percentile, $T(\mathrm{t} + 2) = 50^{\mathrm{th}}$ percentile, and $Flow(\mathrm{t} + 1)$ calculated in the 1-week ahead forecast] for a total of 21 inputs and 1 output [$Flow(\mathrm{t} + 2)$]. The number of input/output variables for the three and four-week ahead forecasts followed the same sequence and included 24 inputs -1 output and 27 inputs -1 output, respectively.

Since the number of inputs (18) and outputs (1) for the one-week ahead forecast were fixed to match those of the WIFFS model, the input layer consisted of 18 input nodes with one node in the output layer. The only part of the network that remained to be configured was the hidden layer(s). Initial forecasting results indicated that the problem of forecasting streamflow in this river basin could be accomplished with only one hidden layer. Networks were initially configured with both one and two hidden layers but improvement in forecasting results were only marginal for the two hidden layer case with training times increasing from approximately ten minutes to approximately 2 h.

Table 1
ANN structures and input variables

Lead time	ANN structure[a]	Time periods for input variables		
		Precipitation	Temperature	Flow
Experiment no. 1				
1	(18, 17, 1)	$t + 1 \rightarrow t - 6$	$t + 1 \rightarrow t - 6$	$t \rightarrow t - 1$
2	(21, 9, 1)	$t + 2 \rightarrow t - 6$	$t + 2 \rightarrow t - 6$	$t + 1 \rightarrow t - 1$
3	(24, 24, 1)	$t + 3 \rightarrow t - 6$	$t + 3 \rightarrow t - 6$	$t + 2 \rightarrow t - 1$
4	(27, 18, 1)	$t + 4 \rightarrow t - 6$	$t + 4 \rightarrow t - 6$	$t + 3 \rightarrow t - 1$
Experiment no. 2				
1	(10, 17, 1)	$t \rightarrow t - 2$	t	$t \rightarrow t - 3$
2	(10, 5, 1)	$t + 1 \rightarrow t - 1$	$t + 1$	$t + 1 \rightarrow t - 2$
3	(10, 7, 1)	$t + 2 \rightarrow t$	$t + 2$	$t + 2 \rightarrow t - 1$
4	(10, 16, 1)	$t + 3 \rightarrow t + 1$	$t + 3$	$t + 3 \rightarrow t$

[a] Note: values represent (number of input nodes, number of hidden nodes, number of output nodes)

Therefore, the ANNs have been built using one-hidden layer. The problem then became that of how many nodes would make up this single hidden layer. Cheng and Noguchi (1996) indicate that selecting too many hidden neurons will increase the training time but without significant improvement on training results. Ranjithan et al. (1993) state that too many hidden neurons will encourage each hidden neuron to memorize one of the input patterns, and thereby diminish the interpolation (generalization) capabilities. Ranjithan et al. (1993) state that the general practice is to determine the number of intermediate units by trial-and-error based on a total error criterion. In this research the trial-and-error process was used.

The error of the training set decreases gradually with an increasing number of intermediate units. Further, the error predictions for the test set decrease initially but tend to increase after reaching a minimum. These observations indicate that although the training process becomes progressively easier, the generalization capabilities of the network reaches an optimum and does not improve indefinitely with an increasing number of intermediate nodes. The results for this case suggest that, for the one-week ahead forecast, a single hidden layer consisting of 17 hidden nodes is suitable for prediction. This model structure is represented by the notation ANN_{Lead} (n_i, n_h, n_o), where n_i is the number of nodes in the input layer, n_h is the number of nodes in the hidden layer, and n_o is the number of nodes in the output layer ($n_o = 1$ in our case). Therefore, the optimum ANN structure for the one-week ahead forecast is presented as ANN_1(18, 17, 1). The optimum ANN structure and the associated input variables, for all forecast lead times, are summarized in Table 1. It was expected that the number of hidden nodes would increase uniformly as the number of inputs to the model increased. As can be seen from Table 1, this was not the case in experiment no. 1. It is not clear why, for example, ANN_2 has roughly half the number of nodes that are used for ANN_1.

3.4.2. Experiment no. 2

After training the initial model in experiment no. 1 with the same inputs used in the WIFFS model, two new inputs were added to the data set. These additional inputs, referred to as 'identifying information', proved to be helpful in training the network. The first of these two inputs was the 'period of the year', *Period*($t + 1$), of the one-week ahead forecast. This input provided the network with information on the season in which the forecast was made. The second of the two inputs was the cumulative precipitation from 1st November to the time of the current period of the year, *Cprecip*(t), up to 1st April. This input represented a measure of the amount of snowpack that accumulates over the winter and adds to the spring runoff. This input was particularly helpful in accurately forecasting the rising limb of the hydrograph during the spring runoff periods. As in experiment no. 1, only one hidden layer was used in the network with the problem again of how many nodes to use in the hidden layer. In addition to determining the number of nodes in the hidden layer was the problem of how

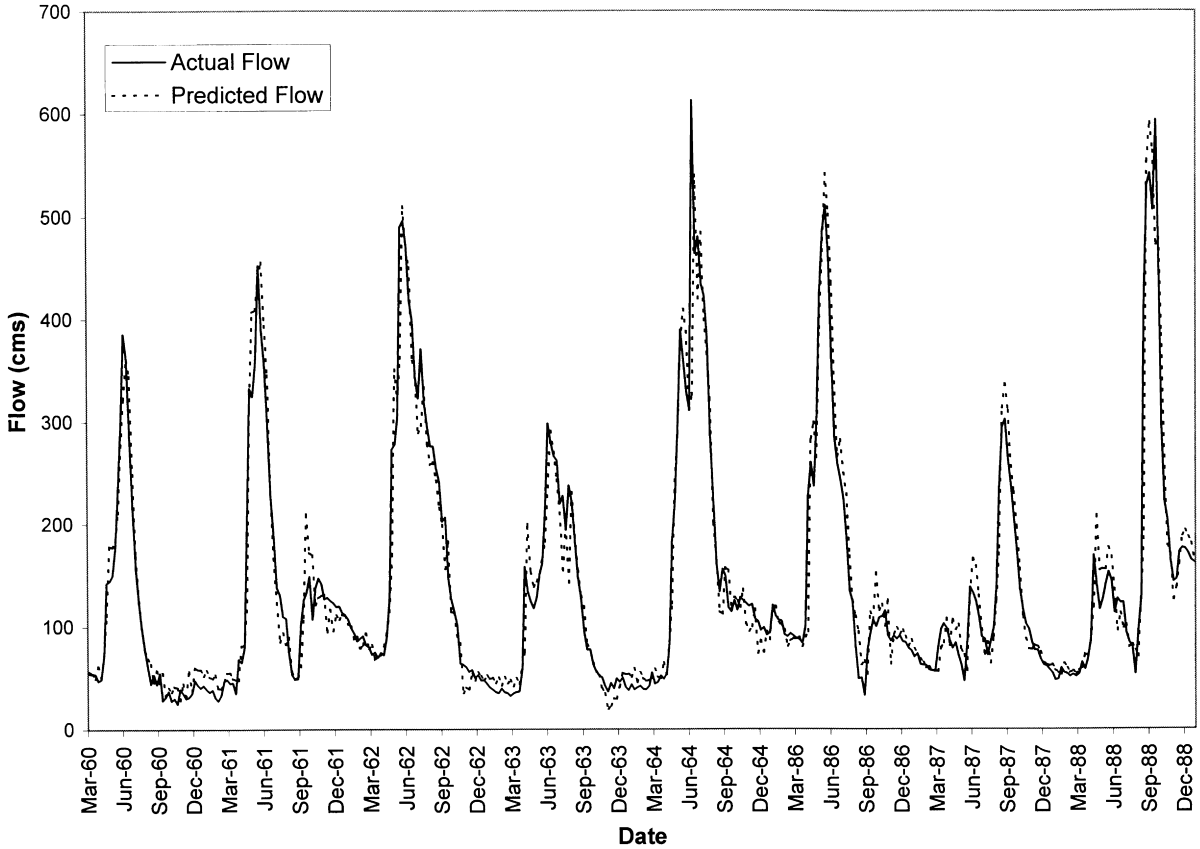

Fig. 4. One-week ahead forecast of flow into Namakan Lake for 1960–1964 and 1986–1988 using the experiment no. 1 ANN model.

many nodes should be used in the input layer. Inputs were not restricted to only those used in the WIFFS model and inputs that were not being utilized by the network to learn the underlying function being modelled were eliminated.

With the additional two inputs added to the data set, a sensitivity analysis of all of the input variables was carried out to determine the relative significance of each of the model inputs. The aim of the sensitivity analysis was to delete those inputs that do not have a significant effect on model performance. The sensitivities were used as a guide to decide which inputs to retain and which to delete by applying some degree of judgement. This was done using the 'leave out analysis' of the software that helps identify inputs that have a relatively small influence on the output.

For the one-week ahead forecast of local inflow, *Flow* $(t + 1)$, the ANN model consisted of the following 10 input variables: 'period of the year' of the 1-week ahead forecast, *Period*$(t + 1)$; past three periods of weekly precipitation, $P(t)$, $P(t - 1)$, $P(t - 2)$; cumulative precipitation since 1st November to the period of the forecast, *Cprecip*(t); past week of average temperature, $T(t)$; and past four periods of average weekly local inflow, *Flow*(t), *Flow*$(t - 1)$,…., *Flow*$(t - 3)$. The benefit of the optimum network configured in experiment no. 2 (for the 1-week ahead forecast) is that the forecaster does not need to make an inference on the future weekly precipitation and average weekly temperature as was needed in experiment no. 1. All data used in the network for the one-week ahead forecast is historical data. In 'Multi-lag' predictions, on the other hand, the 50th percentiles were used as predicted values of precipitation and temperature and previous forecasted streamflows from the neural networks were used as predicted

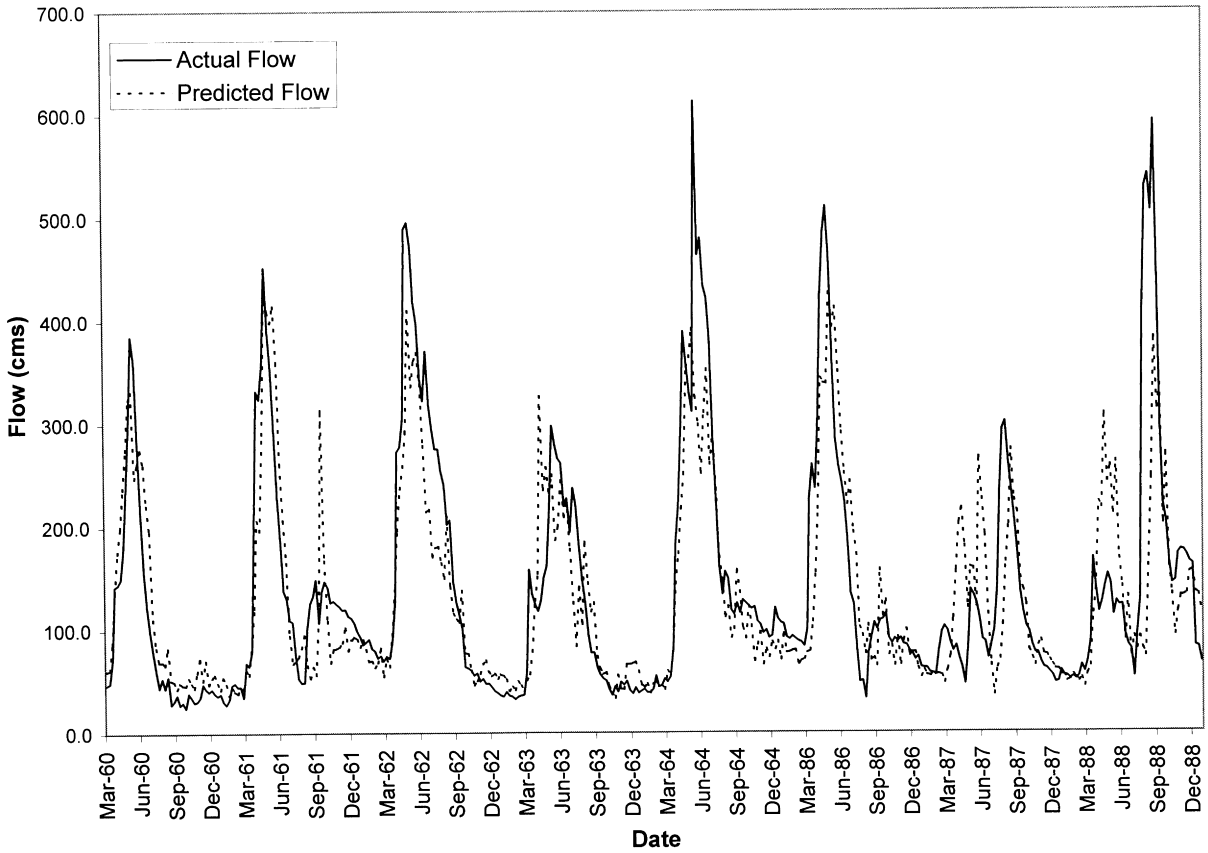

Fig. 5. 4-week ahead forecast of flow into Namakan Lake for 1960–1964 and 1986–1988 using the experiment no. 1 ANN model.

values for streamflow. These values were appended to the database and used to predict future streamflow values. The network for a one-week ahead forecast consisted of 17 hidden neurons and is represented as $ANN_1(10, 17, 1)$. Similar simulations were conducted to determine the hidden node structure for the 2, 3 and 4-week ahead forecasting networks. Table 1 summarizes the structure and input variables for the ANN for the each of the four lead times. Note that in addition to the input variables shown in Table 1 for experiment no. 2, each model also uses the period of the year and cumulative precipitation, as noted above.

4. Results

In order to quantify the relative merit of using ANNs for streamflow forecasting four evaluation measures were used to measure the effectiveness of each method. These performance measures were computed separately for the ANN forecasts of the training data and the independent events or testing data. These measures indicate how well the ANN learned the events it was trained to recognize, and the degree to which each ANN can generalize its training to forecast events not included in the training process. The primary performance measure that compared the WIFFS model to the ANN model was: (i) root mean squared error (RMSE). The three secondary performance measures that were used (only in experiment no. 2) to compare the best ANN model against historical values were; (ii) forecast of total volume; (iii) forecast of peak flow magnitude; and (iv) forecast of peak flow timing. Each of the four effectiveness measures were calculated for the

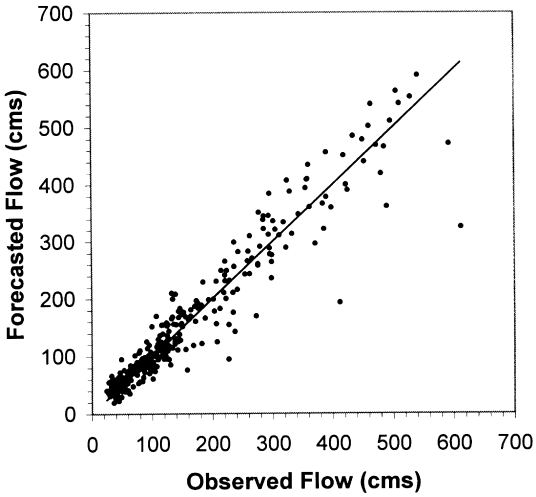

Fig. 6. Observed versus 1-week ahead Forecast of flow into Namakan Lake for 1960–1964 and 1986–1988 using experiment no. 1 ANN.

calibration set (1965–1985) and each of the two verification sets (1960–1964 and 1986–1988).

4.1. Experiment no. 1

The results of forecasting using the ANN model (with the same inputs used in the WIFFS model) on the test data are presented in Figs. 4 and 5 for forecast lead times of one and four weeks respectively. These graphs show the time of the year the forecast was made, along the *x*-axis with a comparison of both the recorded and forecasted streamflows on the *y*-axis. The ANN model forecasts the magnitude and timing of both the summer peaks and the smaller peaks which occur early and late in the year, quite well. The results obtained for a 1-week forecast of Namakan Lake local inflow were generally quite good except for part of October in 1961. The recorded precipitation and streamflow showed approximately 14.0 cm of precipitation in period 34 and approximately 150 cms of streamflow in period 36, both in the year of 1961. Based on historical data one might have expected a streamflow value in the range of 250 cms given a precipitation value of 14.0 cm. This forecasting error is probably due to an error in either the precipitation data or the streamflow data.

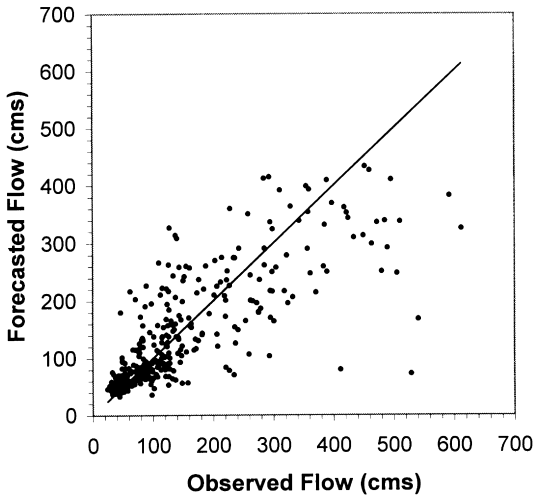

Fig. 7. Observed versus 4-week ahead Forecast of flow into Namakan Lake for 1960–1964 and 1986–1988 using experiment no. 1 ANN.

The ANN model forecasts the magnitude of the baseflow quite well, but encounters greater difficulty in forecasting the magnitude of the peak flows. Generally, as the forecast lead-time grows from 1 to 4 weeks so does the forecast error. This is expected since these forecasts are beginning to use inferences on the precipitation, temperature and forecasted flow from the previous week(s).

Scatter plots showing observed (historical) flows on the *x*-axis against the forecasted flows from the ANN on the *y*-axis, for lead times of one and four weeks, are displayed in Figs. 6 and 7 for the test set data. In each of the scatter plots a perfect forecast lies on the 45° line. The 1-week ahead forecast (Fig. 6) falls relatively close to the 45° line except for three points. These three points are most likely problems with the input data (i.e. streamflow or precipitation gauging errors). It is interesting to note that the flows below 200 cms tend to fall closer to the 45°. This illustrates that the ANN is accurate at forecasting baseflow values. As the flows increase above 200 cms, the forecasts tend to diverge from the 45° line showing the difficulty the ANN has with larger, peakflow values. As the lead-time increases from one through to four weeks, the forecasted flows tend to increasingly diverge from the 45° line, especially for flows greater than 200 cms. This can be seen by comparing the plots in Figs. 6 and 7. This model performance is qualitatively similar to that of the WIFFS model.

The ANN was able to train to a smaller RMSE than

Table 2
Root mean squared forecast error, experiment no. 1

Period	Forecast Lead (weeks)	RMSE (m^3/s)	
		WIFFS	ANN
Calibration (1965–1985)	1	36.5	35.5
	2	56.3	51.9
	3	69.7	68.6
	4	82.3	78.4
Verification (1960–1964)	1	36.9	33.9
	2	51.3	49.4
	3	60.9	54.6
	4	68.6	58.4
Verification (1986–1988)	1	50.4	34.1
	2	78.3	52.7
	3	80.1	71.7
	4	89.8	89.2

the WIFFS model in all of the four forecast leads. The real test of the ANN was the comparison to the WIFFS model for the testing periods from 1960 to 1964 and 1986 to 1988. The ANN was able to forecast to a smaller RMSE than the WIFFS model in all four forecast leads in all 8 years of test data. These results are summarized in Table 2, which lists the calculated RMSE for both models (WIFFS and ANN) and each of the three data sets (one calibration set and two verification sets).

4.2. Experiment no. 2

The results of experiment no. 2 using the ANN on the test data are presented in Figs. 8 and 9 for forecast lead-times of one and four weeks. Scatter plots showing observed (historical) flows on the *x*-axis against the forecasted flows from the ANN on the *y*-axis (for lead times of 1 and 4 weeks) are displayed in Figs. 10 and 11 respectively, for the test set data. The following section quantitatively evaluates the results in terms of four performance measures.

The change in the calculated root-mean-squared error (ΔRMSE) from experiment no. 1 to experiment no. 2 for each of the three data sets is given in Table 3. Negative values of ΔRMSE indicate that the model for experiment no. 2 achieved a lower RMSE than was the case for experiment no. 1. The 'optimum' ANN was able to train to a smaller RMSE than the ANN model trained in experiment no. 1, in all of the four forecast leads. The optimum ANN was able to forecast to a smaller RMSE than the ANN model trained using the inputs from experiment no. 1, in all but two of the forecast leads during the 8 years of test data. The results are summarized in Table 3.

The ANN model was generally effective at forecasting the volume of flow that could be expected in a given year. For a lead-time of one-week, the forecasted volumes ranged from an overprediction of 6.1% to an underprediction of 3.7%. The average overprediction was 0.1%. For a lead-time of two

Table 3
Change in root mean squared forecast error, experiment no. 2

Period	Forecast lead (weeks)	ΔRMSE (m^3/s)
Calibration (1965–1985)	1	−1.2
	2	−6.7
	3	−7.9
	4	−7.5
Verification (1960–1964)	1	−0.4
	2	1.8
	3	−1.0
	4	1.2
Verification (1986–1988)	1	−3.5
	2	−5.1
	3	−6.3
	4	−8.0

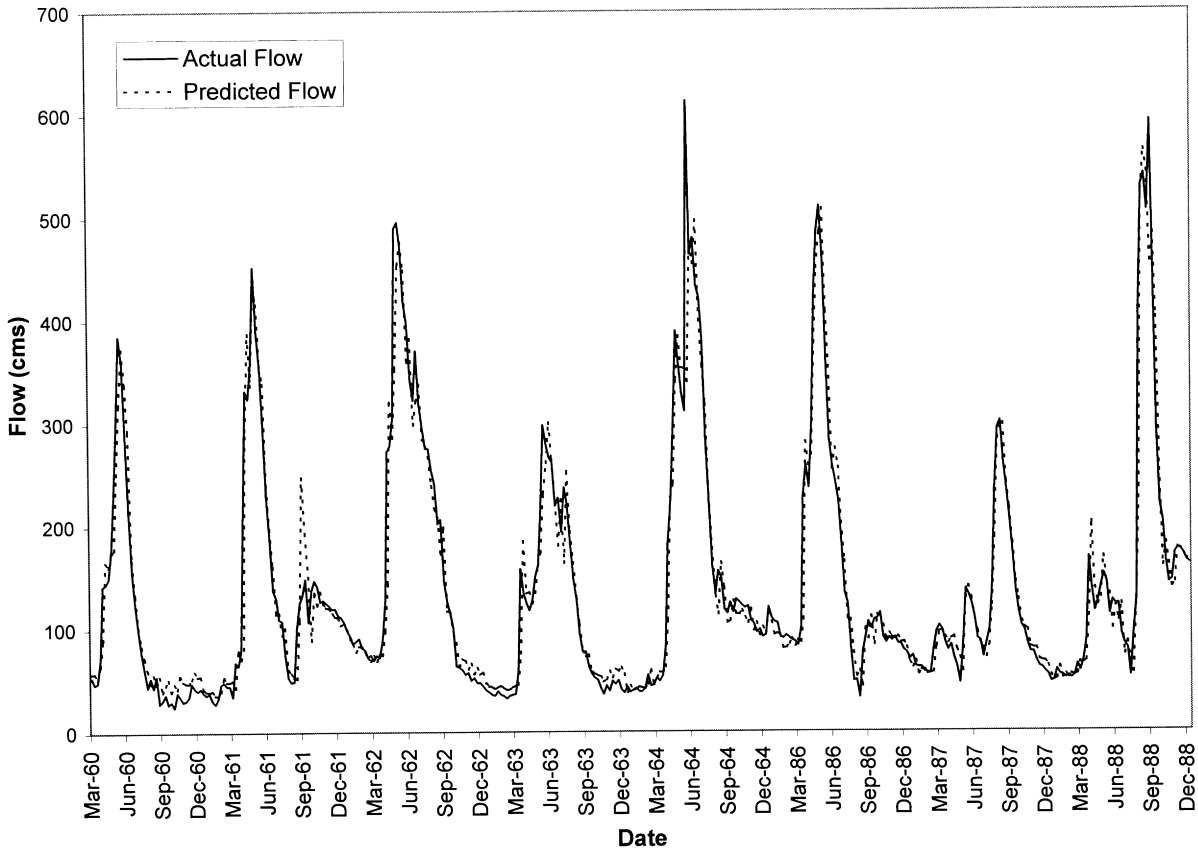

Fig. 8. One-week ahead forecast of flow into Namakan Lake for 1960–1964 and 1986–1988 using the experiment no. 2 ANN model.

weeks, the forecasted volumes ranged from a 4.9% overprediction to a 7.5% underprediction with an average underprediction of 2.1%. Three-week lead times produced forecasted volumes ranging from a 5.4% overprediction to a 15.1% underprediction with an average underprediction of 5.2%. Four-week lead times produced forecasted volumes ranging from a 13.4% overprediction to a 18.2% underprediction with an average underprediction of 3.4%. As expected, there was a considerable range of observed errors in forecasting the total yearly volume within both the 8 years of test data as well as the four different lead times. It was encouraging though, to see that the average total forecasted yearly volume only worsened by 5.3% from a 1-week ahead forecast to a 3-week ahead forecast with forecasts made four weeks ahead improving by 1.8% over the three-week ahead forecast. This latter improvement was unexpected since forecasts generally deteriorate with increasing lead times.

The ANN model performed well at forecasting both the magnitude and timing of the eight peak flows in the test set. However, there is a considerable range of observed errors in forecasting the magnitude of the peak flow. For a lead-time of one-week, the forecasted peak flows ranged from an overprediction of 0.9% to an underprediction of 18.8% with an average underprediction of 4.4%. For a lead-time of 2 weeks, the forecasted peak flows ranged from an overprediction of 9.6% to an underprediction of 17% with an average underprediction of 1.3%. Three-week lead times produced forecasted peak flows ranging from a 3% overprediction to a 28.7% underprediction with an average underprediction of 10.6%. Four-week lead times produced forecasted peak flows ranging from

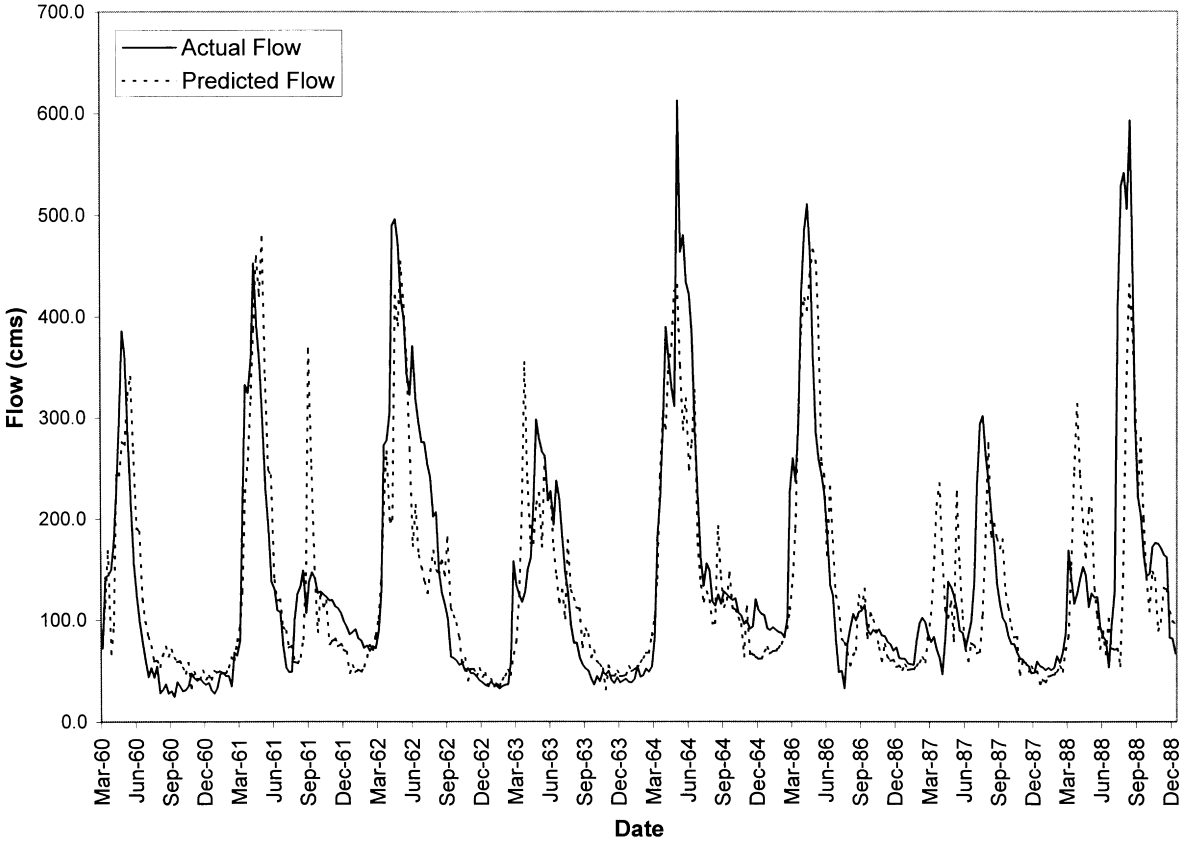

Fig. 9. 4-week ahead forecast of flow into Namakan Lake for 1960–1964 and 1986–1988 using the experiment no. 2 ANN model.

a 5.7% overprediction to a 29.6% underprediction with an average underprediction of 12.6%.

Peak flow timing for a 1-week ahead forecast ranged from being 2 weeks ahead to 3 weeks behind the actual location with an average of 0.90 weeks behind. Two and three-week ahead forecasts of peak location produced values in the range of one-week ahead to three weeks behind the actual peak. The two-week ahead forecasts averaged 1.13 weeks behind the actual peak with the three-week ahead forecast averaging 1.5 weeks behind. The 4-week ahead forecast produced results that ranged from a perfect forecast of peak timing to a forecast of three weeks behind with an average of 1.88 weeks behind. It is interesting to note that the average predicted timing of the forecasted peak deteriorates very slowly when forecasting from 1-week ahead to 4 weeks ahead.

5. Conclusions and recommendations

The ANN model applied to the streamflow forecasting problem seems to have reached encouraging results for the subwatershed under examination. A very close fit was obtained during the training (calibration) phase and the networks developed consistently outperformed the WIFFS model during the testing (verification) phase.

The results obtained with ANNs for 1, 2, 3 and 4-week ahead forecasts are better than those reached in the WIFFS model and confirm the ability of this approach to provide a useful tool in solving a specific problem in hydrology, streamflow forecasting. The initial success of the ANN models developed for the Namakan Lake subwatershed indicates a bright future for further applications in the Winnipeg River Basin as well as other watersheds in the area. The results

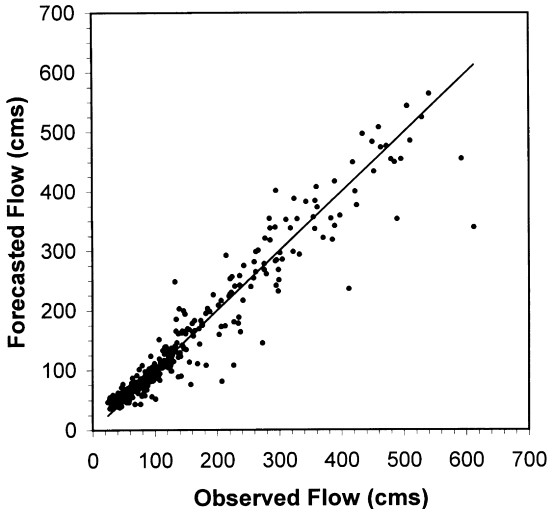

Fig. 10. Observed versus 1-week ahead forecast of flow into Namakan Lake for 1960–1964 and 1986–1988 using experiment no. 2 ANN.

suggest that the ANN approach may provide a superior alternative to the time-series approach for developing input–output simulations and forecasting models in situations that do not require modelling of the internal structure of the watershed.

The potential of ANN models for simulating the hydrologic behavior of watersheds has been presented in this article. The greatest difficulty was determining the appropriate model inputs for such a complex problem. Although ANNs belong to the class of data-driven approaches, it is important to determine the dominant model inputs, as this reduces the size of the network and consequently reduces the training times and increases the generalization ability of the network for a given data set. In the case study considered, sensitivity analyses were used in conjunction with judgement to reduce the number of model inputs from 18 to 10 (for the one-week forecast model). This reduced training times from approximately ten minutes to less than two minutes and reduced the RMSE (test) from 34.0 to 32.5 cms. Similar improvements were observed for the two, three, and four-week forecast models.

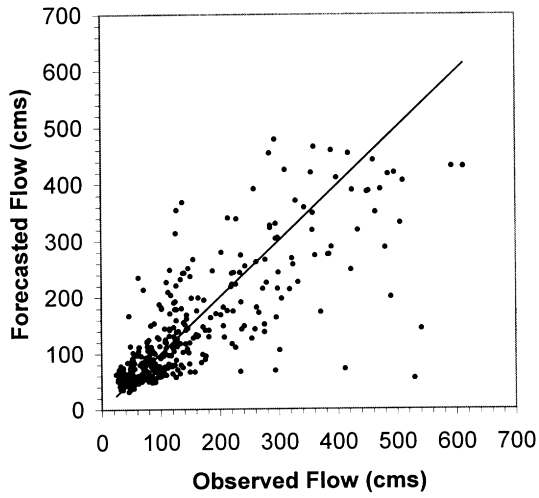

Fig. 11. Observed versus 4-week ahead forecast of flow into Namakan Lake for 1960–1964 and 1986–1988 using experiment no. 2 ANN.

Acknowledgements

The research reported in this article was partially supported by a grant from Manitoba Hydro's Research and Development Program as well as an Industrial Post-Graduate Scholarship from the Natural Sciences and Engineering Research Council of Canada (NSERC). This support is gratefully acknowledged. The authors gratefully acknowledge the helpful comments from two anonymous reviewers.

References

Acres International Ltd.1993. Winnipeg River Basin Flow Forecast Model Development Report. Niagara Falls, Ontario, Canada.

Cheng, X., Noguchi, M., 1996. Rainfall-runoff modelling by neural network approach. In: Proc. Int. Conf. Water Resour. Environ. Res., vol. II, October 29–31, Kyoto, Japan.

Dandy, G., Maier, H., 1996. Use of artificial neural networks for real time forecasting of water quality. Proc. Int. Conf. Water Resour. Environ. Res., vol II, October 29–31, Kyoto, Japan.

Duan, Q., Sorooshian, S., Gupta, V.K., 1992. Effective and efficient global optimization for conceptual rainfall–runoff models. Water Resour. Res. 28 (4), 1015–1031.

Duan, Q., Sorooshian, S., Gupta, V.K., 1994. Optimal use of SCE-UA global optimization method for calibrating watershed models. J. Hydrol. 158, 265–284.

Hipel, K.W., 1986. Time series analysis in perspective. Water Res. Bull. 21 (4), 609–623.

Hsu, K., Gupta, H.V., Sorooshian, S., 1995. Artificial neural network modeling of the rainfall–runoff process. Water. Resour. Res. 31 (10), 2517–2530.

Karunanithi, N., Grenney, W.J., Whitley, D., Bovee, K., 1994.

Neural networks for river flow prediction. J. Comp. Civ. Engrg. ASCE 8 (2), 201–220.

Kitanidis, P.K., Bras, R.L., 1980. Adaptive filtering through detection of isolated transient errors in rainfall–runoff models. Water Resour. Res. 16 (4), 740–748.

Kitanidis, P.K., Bras, R.L., 1980. Real-time forecasting with a conceptual hydrological model. Water Resour. Res. 16 (4), 740–748.

Lake of the Woods Control Board (LWCB), 1994a. Winnipeg river basin flow forecast model/user manual. Acres International LTD. Niagara Falls, Ontario.

Lake of the Woods Control Board (LWCB), 1994b. Managing the water resources of the Winnipeg river drainage basin. Ottawa, Ontario, Canada.

Lippmann, R.P., 1987. An introduction to computing with neural nets. IEEE ASSP Mag. 4–22.

Lorrai, M., Sechi, G.M., 1995. Neural nets for modeling rainfall–runoff transformations. Wat. Resour. Man. 9, 299–313.

Maier, H.R., Dandy, G.C., 1996. The use of artificial neural networks for the prediction of water quality parameters. Water Resour. Res. 32 (4), 1013–1022.

Promised Land Technologies Inc.1993 Braincel User Manual, Version 2.0. New Haven CT.

Raman, H., Sunilkumar, N., 1995. Multivariate modelling of water resources time series using artificial neural networks. J. Hydrol. Sci. 40 (2), 145–163.

Raman, H., Chandramouli, H., 1996. Deriving a general operating policy for reservoirs using neural networks. J. Wat. Resour. Plan. Man, 342–347.

Ranjithan, S., Eheart, J.W., Garrett Jr, J.H., 1993. Neural network-based screening for groundwater reclamation under uncertainty. Water Resour. Res. 29 (3), 563–574.

Rumelhart, D.E., Hinton, G.E., Williams, R.J., 1987. Learning internal representations by error propagation. In: Rumelhart, D.E., McClelland, J.L. (Eds.), Parallel Distributed Processing: Explorations in the Microstructure of Cognition, vol. 1. MIT Press, Cambridge, MA, pp. 318–362.

Smith, J., Eli, R.N., 1995. Neural network models of rainfall–runoff process. J. Wat. Resour. Plan. Mgmt, 499–508.

Sorooshian, S., Daun, Q., Gupta, V.K., 1993. Calibration of rainfall–runoff models: application of global optimization to the sacramento soil moisture accounting model. Water Resour. Res. 29 (4), 1185–1194.

Valluru, B. R., Hayagriva, R.V., 1993. C^{++}, Neural networks and fuzzy logic. Management information source, 1993.

Yapo, P., Gupta, V.K., Sorooshian, S., 1996. Calibration of conceptual rainfall–runoff models: Sensitivity to calibration data. J. Hydrol. 181, 23–48.