Journal of Hydroinformatics | 10.4 | 2008

# TDNN with logical values for hydrologic modeling in a cold and snowy climate

Yonas B. Dibike and Paulin Coulibaly


## ABSTRACT

Watershed runoff in areas with heavy seasonal snow cover is usually estimated using physically based conceptual hydrologic models. Such simulation models normally require a snowmelt algorithm consisting of a surface energy balance and some accounting of internal snowpack processes to be part of the modeling system. On the other hand, artificial neural networks are flexible mathematical structures that are capable of identifying such complex nonlinear relationships between input and output datasets from historical precipitation, temperature and streamflow records. This paper presents the findings of a study on using a form of time-delayed neural network, namely time-lagged feedforward neural network (TLFN), that implicitly accounts for snow accumulation and snowmelt processes through the use of logical values and tapped delay lines. The logical values (in the form of symbolic inputs) are used to implicitly include seasonal information in the TLFN model. The proposed method has been successfully applied for improved precipitation–runoff modeling of both the Chute-du-Diable reservoir inflows and the Serpent River flows in northeastern Canada where river flows and reservoir inflows are highly influenced by seasonal snowmelt effects. The study demonstrates that the TLFN with logical values is capable of modeling the precipitation–runoff process in a cold and snowy climate by relying on 'logical input values' and tapped delay lines to implicitly recognize the temporal input–output patterns in the historical data. The study results also show that, once the appropriate input patterns are identified, the time-lagged neural network based models performed quite well, especially for spring peak flows, and demonstrated comparable performance in simulating the precipitation–runoff processes to that of a physically based hydrological model, namely HBV.


**Key words** | cold and snowy climate, logical values, precipitation–runoff modeling, time delay neural networks

**Yonas B. Dibike**
National Water Research Institute,
Environment Canada,
Water & Climate Impact Research Centre,
University of Victoria,
Victoria, BC V8W 2Y2,
Canada

**Paulin Coulibaly** (corresponding author)
Department of Civil Engineering,
School of Geography and Earth Science,
McMaster University,
1280 Main Street West,
Hamilton, Ontario L8S 4L7,
Canada
Tel.: +1 905 525 9140 X23354
Fax: +1 905 529 9688
E-mail: *couliba@mcmaster.ca*

## INTRODUCTION

Seasonal snowpacks account for the major source of stream flow in most watersheds with a cold and snowy climate. In most regions in Canada, and especially in the province of Quebec, snowmelt runoff is responsible for the high flows in the spring season as well as about 40% of the annual flow volume (Coulibaly *et al.* 2000). Therefore, identifying the relationship between total precipitation falling on such watersheds and the corresponding runoff in the rivers and streams is a very important task for most hydrologic engineering design and management purposes. The precipitation–runoff relationship is known to be highly nonlinear and describing such relationships becomes even more complex when dealing with the hydrology of regions with cold and snowy climates. Watershed runoff in areas with seasonal snow cover is usually estimated using physically based snowmelt–runoff models. Such simulation models normally require a snowmelt algorithm to be part of the modeling system. Most snowmelt algorithms consist of a

doi: 10.2166/hydro.2008.049

surface energy balance and some accounting of internal snowpack processes. The more detailed surface energy balance models require inputs such as air temperature and radiation data, relative humidity, wind speed, as well as other data such as area–elevation curve and ratio of the snow-covered area to the total area of the basin (Melloh 1999). Many models have been developed around the world over the last four decades or so to describe snowmelt runoff. The World Meteorological Organization (1986) lists and summarizes 18 different snowmelt–runoff models. Some of the most widely used snowmelt models rely on empirical equations combining the degree–day method and the energy contribution of rain as described by Linsley *et al.* (1982). For example, Martinec *et al.* (1994) proposed a model where the water produced from rainfall and snowmelt is computed, superimposed on the calculated recession flow and transformed into daily discharge from the basin. However, in addition to the large amount of basin information that has to be collected to use such models, they also have a number of parameters whose values have to be determined through calibration based on actual observations. Moreover, natural watersheds are characterized by a large degree of heterogeneity in the important properties controlling snow accumulation, snowmelt and runoff processes. Therefore, more research is still needed to fully understand the hydrology of snow-covered areas in order to develop a distributed physically based hydrological model which accounts for the snow accumulation and snowmelt processes even better (Melloh 1999). At the same time, estimation of surface energy balance and internal snowpack processes may be of little concern for operational hydrologic modeling. Rather, most operational hydrologists are concerned in accurate, site-specific estimation of river flows and reservoir inflows. Recent studies have shown the potential of ANN-based models for rainfall–runoff modeling (see, for example, Coulibaly *et al.* (1999), ASCE Task Committee (2000), Maier & Dandy (2000), Dawson & Wilby (2001) for various reviews). Therefore, it would be appropriate to further explore the data-driven modeling paradigms, such as temporal neural networks, for the modeling of precipitation–runoff processes in watersheds with cold and snowy climates.

Artificial neural networks (ANNs) are flexible mathematical structures that are capable of identifying complex nonlinear relationships between input and output datasets of observed historical records without trying to explicitly represent the different components of the hydrological processes. A number of researchers have investigated the potential of neural networks in modeling watershed runoff based on rainfall inputs and a lot more ANN-based rainfall–runoff models have also been proposed for stream flow forecasting (Minns & Hall 1996; Campolo *et al.* 1999; Zealand *et al.* 1999; Thirumalaiah & Deo 2000; Dibike & Solomatine 2001). Coulibaly *et al.* (2001*a*) used time-lagged neural networks to predict reservoir inflow based on precipitation, snowmelt and temperature data, whilst See & Kneale (2004) applied time-delay neural networks to river level forecasting. Zhang & Govindaraju (2003) proposed geomorphology based neural networks for estimation of direct runoff over watersheds. How successful ANNs have been in dealing with hydrologic problems is discussed extensively in the review paper by the ASCE Task Committee (2000). However, not that many researches have been focused on finding an appropriate way to account for snowmelt processes within the ANN modeling framework. Applying ANNs to simulate watershed runoff in cold and snowy areas requires special attention in data preparation because of the delay between the snowfall (solid precipitation) and the corresponding snowmelt reaching the river system. Moreover, since the runoff from a watershed is a function of not only the concurrent precipitation but also the antecedent values, time-lagged neural networks with internal memory through time delay lines are more appropriate to capture the time history of the inputs. Fuhrman & Minns (2000) investigated the selection of input data for the hydrologic snowmelt model using ANN. They demonstrated that transforming the raw input data into elementary surrogate parameters using some physical insight and commonsense judgment which includes the use of antecedent values can significantly improve the performance of ANN. This paper presents the findings of a study where time-lagged neural networks have been successfully applied to capture snow accumulation and snowmelt periods through the use of 'logical input values' and tapped delay lines. The proposed approach is successfully applied for precipitation–runoff modeling of the Chute-du-Diable and the Serpent River watersheds in northern Quebec where the river flows and reservoir inflows are highly influenced by seasonal snow effects.

## STUDY AREA AND DATA

The study area considered for the application of the time-lagged neural-network-based precipitation–runoff modeling is the Chute-du-Diable watershed and the Serpent River sub-basin located in the Saguenay watershed (Figure 1) in northern Quebec. Saguenay is a well-known flood-prone region and there are a large number of reservoirs and dams in the watershed. The Chute-du-Diable sub-basin has an area of 9,700 $km^2$ and is located in the eastern part of the Saguenay watershed. The Serpent River watershed is a sub-basin of about 1,760 $km^2$ located in the northwestern part of the Chute-du-Diable watershed (Figure 1). The mean annual precipitation in the region is around 946 mm, of which some 30% is in the form of snowfall occurring mainly in the winter season between the months of November and March (Coulibaly *et al.* 2001*b*). The temperature in the area ranges between +30°C and

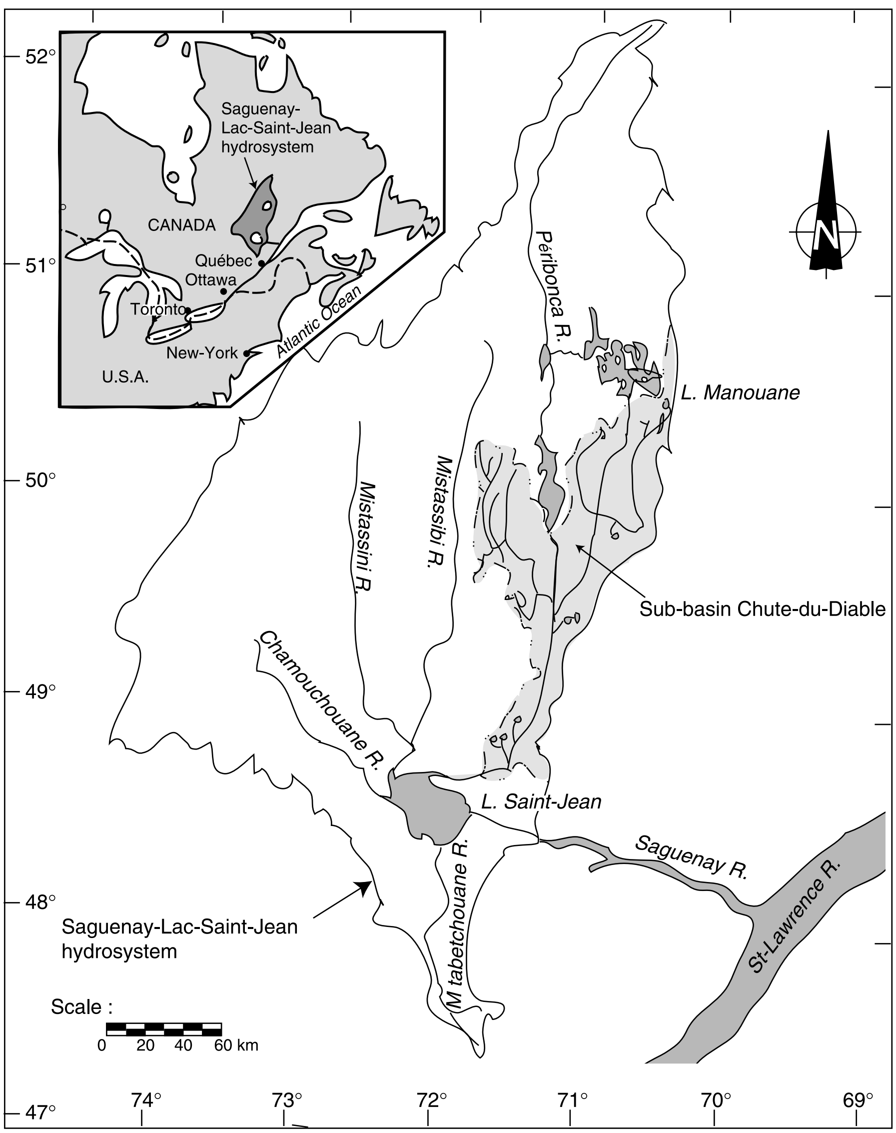


**Figure 1** | Location map of the Saguenay watershed, the Chute-du-Diable and Sepent sub-basins (Coulibaly *et al.* 2001*b*).

−40°C, with the mean annual temperature being of the order of +1°C. A detailed description of the study area is provided by Dibike & Coulibaly (2005). Two experiments of precipitation–runoff modeling are considered in this paper. The first one is the simulation of the total daily inflows into the Chute-du-Diable reservoir while the second one is a daily streamflow simulation in the Serpent River. Two sets of historical data with daily precipitation and temperature records between 1961 and 2002 are obtained from two meteorological stations located at Chute-du-Diable (48.75°N, 71.7°W) and Chute-des-Passes (49.9°N, 71.25°W). Precipitation and temperature data from both stations are used as input for the reservoir inflow modeling while only the data from the Chute-des-Passes station is used as input for the Serpent River flow modeling since the second station is far from the Serpent sub-basin. In the case of reservoir inflow modeling, the required output from the model is the daily inflow from the catchment into the Chute-du-Diable reservoir while, in the case of streamflow modeling, the required output is the mean daily discharge of the Serpent river (at station ID #062214). Forty years of reservoir inflow data (between 1961 and 2000) and twelve years of stream flow data in the Serpent River (between 1991 and 2002) are used for the simulation experiments. For the case of reservoir inflow simulation, the first thirty years' data (1961–1990) is used for training (calibrating) the models while the remaining ten years (1991–2000) is used for testing (validation) of the models. Similarly, for the case of the Serpent River flow simulation, the first seven years (1991–1997) is used for training (calibrating) the models and the remaining five years (1998–2002) is used for testing (validation) of the models. Each of the time-lagged neural-network-based hydrologic simulation models in this study are developed within the neural network simulation environment called NeuroSolutions (Principe *et al.* 2000).

## METHODS

### Artificial neural networks (ANNs)

A neural network can be, in general, characterized by its architecture, which is represented by the pattern of connections between the nodes, its method of determining the connection weights and the activation functions that it employs. As a result, ANNs constitute a diverse family of networks whereby the functionality of each type of network is determined by the network topology, the individual neural characteristics and the learning or training strategy employed. Within the last decade, the study of artificial neural networks has experienced a huge resurgence due to the development of more sophisticated algorithms and the emergence of powerful computational tools (ASCE Task Committee 2000). Multi-layer perceptrons (MLPs), which constitute probably the most widely used network architecture, are composed of a hierarchy of processing units where information flow in the network is restricted to a flow, layer by layer, from the input to the output, hence also called a feedforward network. While MLPs are popular in many application areas, they are not well suited to temporal sequence processing due to the lack of time lagged and/or feedback connections necessary to provide a dynamic model (Coulibaly *et al.* 2001*b*).

In temporal problems, the neural network must have access to the time dimension in order to exploit the signal time structure of the input data. Since natural systems are mostly causal, the search is restricted to the past of the signal. de Vries & Principe (1991) have put forward a theory of neural networks with time delay which has been implemented in different forms since then. There are different ways of introducing 'memory' in a neural network in order to develop a dynamic neural network. *Time-delayed or time-lagged feedforward networks* (TLFN) and *recurrent networks* (RNN) are the two major groups of candidate dynamic neural networks mostly used in time series analysis (Dibike *et al.* 1999; Coulibaly *et al.* 2001*a*,*b*). Chiang *et al.* (2004) compared static and dynamic neural networks for rainfall–runoff modeling and reported that the dynamic neural network generally produced better and more stable flow forecasting than the static network. However, RNN require complex training algorithms and hence are computationally costly. The analysis in this paper concerns the time-lagged neural networks that can be easily trained for practical application.

### Time-lagged feedforward networks (TLFN)

A neural network can be formulated by replacing the neurons of an MLP with a memory structure, which is sometimes called a *tap delay line*. If the memory structure is incorporated only with the input neurons, then it is called a focused time-lagged feedforward neural network (TLFN). The size of the memory layer (the tap delay) depends on the number of past samples that are needed to describe the input characteristics in time and it has to be determined on a case-by-case basis. TLFN uses *delay-line* processing elements (PE) which implements memory by delay, that is, by simply holding past samples of the input signal as shown in Figure 2 (Coulibaly *et al.* 2005). The output of such a network with one hidden layer is given by

$$y(n) = \varphi_1\left(\sum_{j=1}^{m} w_j y_j(n) + b_0\right)$$
$$= \varphi_1\left(\sum_{j=1}^{m} w_j \varphi_2\left(\sum_{l=0}^{k} w_{jl} x(n-l) + b_j\right) + b_0\right) \qquad (1)$$

where $m$ is the size of the hidden layer, $y_j$ is the output from the $j$th hidden node, $w_j$ and $w_{jl}$ are the connection weights, $\phi_1$ and $\phi_2$ are transfer functions at the output and hidden layers, respectively, and $x(n)$ represents the input pattern at time step $n$. Such a delay-line 'remembers' $k$ samples in the past. The coefficients $b_j$ and $b_0$ are additional network parameters (often called biases) to be determined during training. A detailed description of the TLFN as used here is provided by Coulibaly *et al.* (2005).

An interesting feature of the TLFN is that the tap delay line at the input does not have any free parameters; therefore the network can still be trained with the classical backpropagation algorithm. The TLFN topology has been successfully used in nonlinear system identification, time series prediction (e.g. Coulibaly *et al.* 2001*b*) and temporal

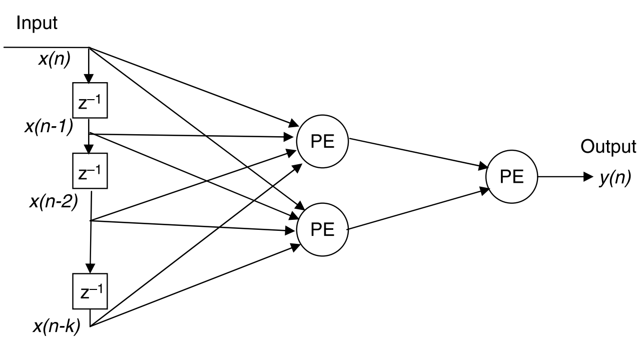


**Figure 2** | Focused time-lagged feedforward network (TLFN) with one input, one hidden layer and a tap delay line with $k + 1$ taps (Coulibaly *et al.* 2005).

pattern recognition (Principe *et al.* 2000). A particular feature of the TLFN is that the memory structure is focused on the input layer; this makes it different from the general time delay neural network which uses additional internal delays at each hidden neuron. A major advantage of the TLFN is that it is less complex than the conventional time delay and recurrent networks, and has the same temporal patterns processing capability (Dibike *et al.* 1999; Coulibaly *et al.* 2001*b*).

## Identifying optimal input patterns for TLFNs

While the precipitation is the main driving factor for the proposed TLFN-based hydrological model, the temperature data provides information on the state of the precipitation (rain or snow) and the available potential for evapotranspiration. Therefore, the daily total precipitation and the mean daily temperature at each of the stations in the watershed are provided as input to the neural networks. The tap-delay (time lag) in the input layer of the TLFN stores a few days of antecedent values for each input and uses it as additional information to the network. However, since accumulation of snow and its subsequent melting is a seasonal process, the TLFN model needs to have input information extending to several weeks or months in the past. Such information can be provided to the neural networks in terms of inputs of moving average temperatures or cumulative precipitation values. Such inputs provide the network with a long term memory to help 'remember' the past. Moreover, logical information regarding the season may also be provided to the network by including additional input information called 'logical input values'. Specifically, months have been considered as logical inputs using 0 and 1 for considering time and/or seasonal effects on the outputs of the system. In representing months as logical inputs, 12 input columns have been included for 12 months: for example, if the computation is in January then the logical input will be 1 for that month, and 0 for the other 11 months. Subsequently, if the computation is in February, then the logical input will be 1 for that month and 0 for the other 11 months, and so on. Although this approach introduces a discontinuity at monthly boundaries because of considering 31 January to be the same as 1 January, and similarly 1 February to be the same as 28 February, and so on, it permits us to effectively account for seasonal effects that are particularly important in cold and snowy regions. This logical input value (or symbolic input) identifying each month of the year provides the network with very important information regarding the time of the year corresponding to each input–output pattern and hence can be considered as an indicator of the season.

In order to identify the contribution of each type of input representation to the success of the TLFN model, four cases of each experiment are designed (see Table 1) with increasing degree of information in the input pattern as follows:

Case 1: No long-term memory and no symbolic input (or logical input values).
Case 2: No long-term memory and logical input values.
Case 3: Long-term memory and no logical input values.
Case 4: Long-term memory and logical input values.

These four cases were set up for each of the two modeling problems, namely, for the reservoir inflow modeling and for the modeling of Serpent River flow. In cases where long term memories are considered, moving average temperatures and cumulative precipitations with different window lengths of antecedent values are considered till the best performing networks are identified. The simulation outputs of the best neural network models are finally compared with that of the HBV physically based conceptual hydrologic model outputs.

## RESULTS AND DISCUSSION

For each case of TLFN modeling described previously, the tangent hyperbolic and the linear transfer functions are used in the hidden and output layers, respectively. The optimal lag time (tap delay) at the input layer and the number of neurons in the hidden layer are identified after a number of experimental trials for each case. Model performances (or the goodness of fit between observed and simulated outputs) for each simulation experiment are measured in terms of root mean square error (RMSE) and the Nash & Sutcliffe (1970) model efficiency ($R^2$). The model simulation efficiency indicates how well the observed and simulated values fit a 1:1 relationship. Values near 1 for $R^2$ indicate a good fit of the data, whereas a RMSE value closer to 0 indicates a better model performance.

### Performance of TLFN models

The input patterns of the best performing networks in each of the four experimental cases of Chute-du-Diable reservoir inflow and Serpent River flow simulation are presented in Table 1. The optimum input patterns identified in Table 1 suggest that the length of optimum long-term memory depends on the size of the drainage basin, with the larger basin (for Chute-du-Diable reservoir inflow modeling) requiring longer memory lengths than the smaller basin (for Serpent River flow modeling). While time lags of up to 15 d are used in the input layer of TLFN for reservoir simulation, time lags of up to 10 d are used for the river flow simulation. The optimum numbers of hidden nodes identified for the four cases of input representations in both simulation experiments are 15, 20, 24 and 30, respectively.

The performances of each model during the training and validation periods of the reservoir inflow and river flow simulation are presented in Table 2. The test results in the table show how the time-lagged neural network simulation performance increases whenever the logical values are used (Case 2 and Case 4). Both Case 1 and Case 3 are significantly improved when logical values are included. For the daily river flow simulation, the use of symbolic or logical input values without long-term memory (Case 2) provides about 70% of improvement in model efficiency (versus Case 1) whereas the use of long-term memory without logical inputs (Case 3) results in about 100% improvement in model efficiency (versus Case 1). This highlights the importance of both the logical values and the long-term memory as part of the input patterns. However, overall, the use of both logical values and long-term memory appears the best approach for river flow modeling in cold and snowy climates. Similar findings are shown for the reservoir inflow simulation results (Table 2). Consistently, it is shown that the use of logical values (Case 2 and Case 4) significantly improve the TLFN model performance for reservoir inflow simulation.

To further assess the model simulation performance in general, the scatter plots between observed and simulated reservoir inflows are presented for each of the four cases

**Table 1** | The different inputs of the TLFN models for the Chute-du-Diable reservoir inflow and the Serpent River flow simulations

| Case 1 | Case 2 | Case 3 | Case 4 |
|---|---|---|---|
| **Chute-du-Diable reservoir inflow simulation** | | | |
| Total daily precipitation at Chute-du-Diable and Chute-des-Passes | Total daily precipitation at Chute-du-Diable and Chute-des-Passes | Total daily precipitation at Chute-du-Diable and Chute-des-Passes | Total daily precipitation at Chute-du-Diable and Chute-des-Passes |
| Mean daily temperature at both stations | Mean daily temperature at both stations | Preceding three months' cumulative precipitation at both stations | Preceding three months' cumulative precipitation at both stations |
| | Logical input value (or symbolic input) representing the month | Mean daily temperature at both stations | Mean daily temperature at both stations |
| | | Preceding six weeks' average temperature at both stations | Preceding six weeks' average temperature at both stations |
| | | | Logical input value (or symbolic input) representing the month |
| **Serpent River flow simulation** | | | |
| Total daily precipitation at Chute-des-Passes | Total daily precipitation at Chute-des-Passes | Total daily precipitation at Chute-des-Passes | Total daily precipitation at Chute-des-Passes |
| Mean daily temperature at Chute-des-passes | Mean daily temperature at Chute-des-passes | Preceding two months' cumulative precipitation | Preceding two months' cumulative precipitation |
| | Logical input value (or symbolic input) representing the month | Mean daily temperature at Chute-des-Passes | Mean daily temperature at Chute-des-Passes |
| | | Preceding two weeks' average temperature | Preceding two weeks' average temperature |
| | | | Logical input value (or symbolic input) representing the month |

**Table 2** | TLFN model training and validation statistics for the four different cases of river flow and reservoir inflow simulations

| | Reservoir inflow simulation | | | | River flow simulation | | | |
|---|---|---|---|---|---|---|---|---|
| | Training | | Testing | | Training | | Testing | |
| | RMSE | $R^2$ | RMSE | $R^2$ | RMSE | $R^2$ | RMSE | $R^2$ |
| Case 1 | 139 | 0.62 | 140 | 0.61 | 32.4 | 0.50 | 35.9 | 0.38 |
| Case 2 | 77.8 | 0.82 | 107.2 | 0.78 | 26.7 | 0.70 | 28.1 | 0.61 |
| Case 3 | 88.8 | 0.80 | 102.1 | 0.81 | 20.3 | 0.82 | 25.7 | 0.78 |
| Case 4 | 83.5 | 0.82 | 93.3 | 0.83 | 9.9 | 0.86 | 22.6 | 0.83 |

(Figure 3). Figure 3 show how the model's performance is improved by providing appropriate inputs along with logical values (Case 4) successively to the model. The inclusion of inputs providing long term memory in terms of cumulative precipitation and moving average temperatures have a significant contribution to the improvement in the model performance. Moreover, the inclusion of logical inputs in terms of symbolic values identifying to which month a particular input pattern belong resulted in a further improvement in the performance of the TLFN model, especially for the peak flows (Figure 3) which are of particular interest in that region. The scatter plots of river flows are not shown but, they indicate even better improvement in the model performance as shown in Table 2. Figure 4 also clearly substantiates this improvement that has been achieved in the simulation outputs by plotting the model validation errors of Case 1 and Case 4 together. This figure highlights the reduction in the simulation error achieved by identifying the most appropriate input

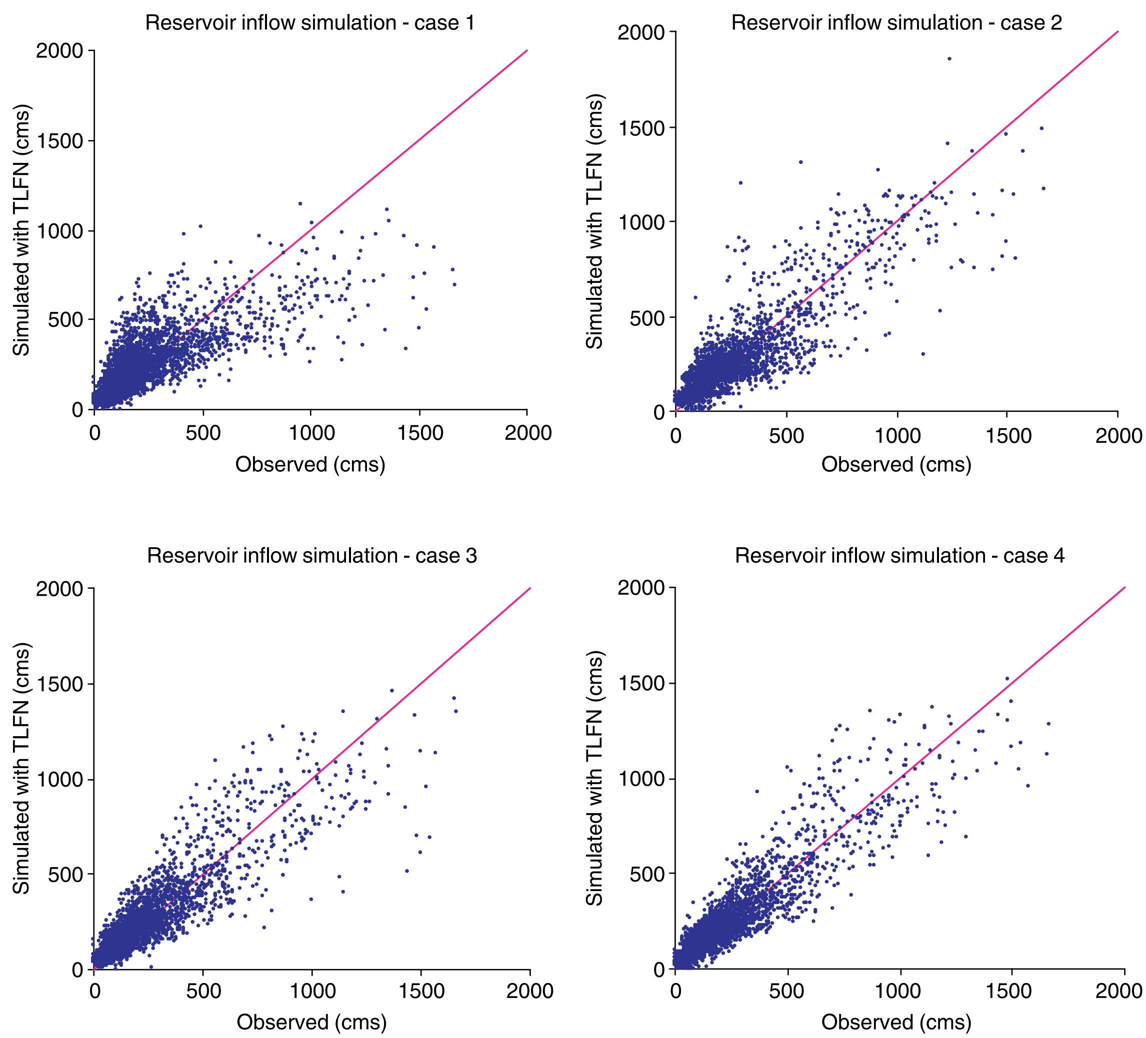


**Figure 3** | Scatter plots between observed and TLFN simulated reservoir inflow for the validation period corresponding to each of the four cases of input representation.

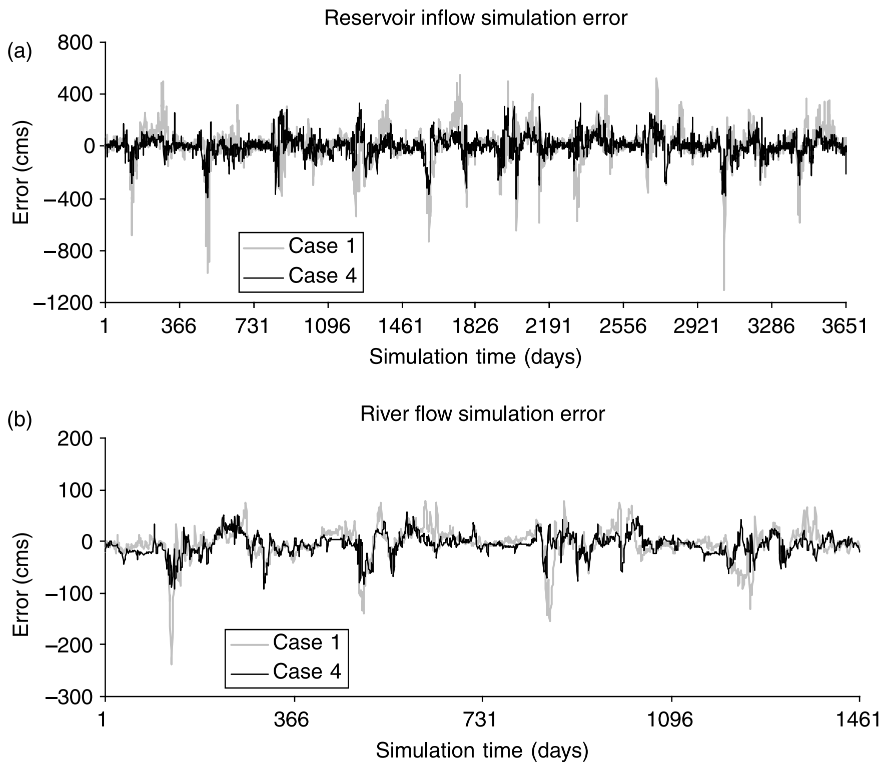


**Figure 4** | Improvement in the simulation error of TLFN-based (a) reservoir inflow and (b) river flow models as a result of improved input representation (logical values and long-term memory).

representation for the TLFN model. Another interesting point is that most of the large error reduction corresponds to periods of high flow. This indicates how better representation of the input pattern improved the simulation in the high flow season where a more accurate estimation of flood events is very important.

## Comparison of optimal TLFN and HBV model

HBV (abbreviation for Hydrologiska Byråns Vattenbalansavdelning) is an integrated hydrologic modeling system developed at the Swedish Meteorological and Hydrological Institute, and has been applied to a wide range of applications including the analysis of extreme floods (Harlin & Kung 1992), the assessment of the effects of land-use change (Arheimer & Brandt 1998) and the effects of climate change (Lidén & Harlin 2000; Dibike & Coulibaly 2005). It can best be described as a semi-distributed conceptual hydrologic model. Daily total precipitation and daily maximum and minimum temperature data are provided as input to the model and it provides mean daily discharge as output. The model has a routine for snow accumulation and snowmelt based on a degree–day relation with an altitude correction of temperature. The soil moisture accounting routine accounts for soil field capacity and change in soil moisture storage due to rainfall/snow melt and evapotranspiration while the runoff generation routine transforms water from the soil moisture zone to runoff.

HBV models are set up for both the Chute-du-Diable reservoir inflow and Serpent River flow simulations. While the daily total precipitation and temperature data at the two meteorological stations (Chute-du Diable and Chute-des-Passes) are used as input to the reservoir inflow model, the same data from only the Chute-des-Passes station is used as input to the river flow model. The same training and testing periods used for the TLFNs are used for calibration and validation of the two HBV models. The optimum model parameters for both models are determined through a calibration process, where the different model parameters are adjusted until simulated and observed runoff show satisfactory agreement. The comparative performance of the best TLFN models (Case 4) with that of the HBV models is presented in Table 3. While the two types of model gave a

**Table 3** | Comparative performance of the best TLFN models with that of the HBV models for reservoir inflow and river flow simulations

| | Calibration | | | | Validation | | | |
|---|---|---|---|---|---|---|---|---|
| | TLFN | | HBV | | TLFN | | HBV | |
| | RMSE | $R^2$ | RMSE | $R^2$ | RMSE | $R^2$ | RMSE | $R^2$ |
| Reservoir inflow | 80.9 | 0.82 | 81.8 | 0.81 | 93.3 | 0.83 | 100.9 | 0.80 |
| River flow | 9.9 | 0.86 | 10.6 | 0.85 | 22.6 | 0.83 | 5.17 | 0.84 |

comparable performance in the case of the Serpent River flow simulation, the optimal TLFN (with logical input values) seems to show a slightly better performance for the case of the Chute-du-Diable reservoir inflow simulation. To substantiate the model validation statistics shown in Table 3, the simulation outputs of the two models for the validation period are also plotted in Figure 5 along with the observed flow data. Once again, the plots show how the TLFN model with logical values and long-term memory resulted in as close a fit to the observed flow data as the HBV model – suggesting that the TLFN with logical inputs effectively captures the seasonal patterns needed to account for snow accumulation and snowmelt processes. Therefore, the proposed TLFN with logical values appears a good alternative approach for hydrologic modeling in a cold and snowy climate.

## SUMMARY AND CONCLUSIONS

Most physical systems in the real world can be represented by models to different degrees of accuracy with different forms of representation. The more conventional way to

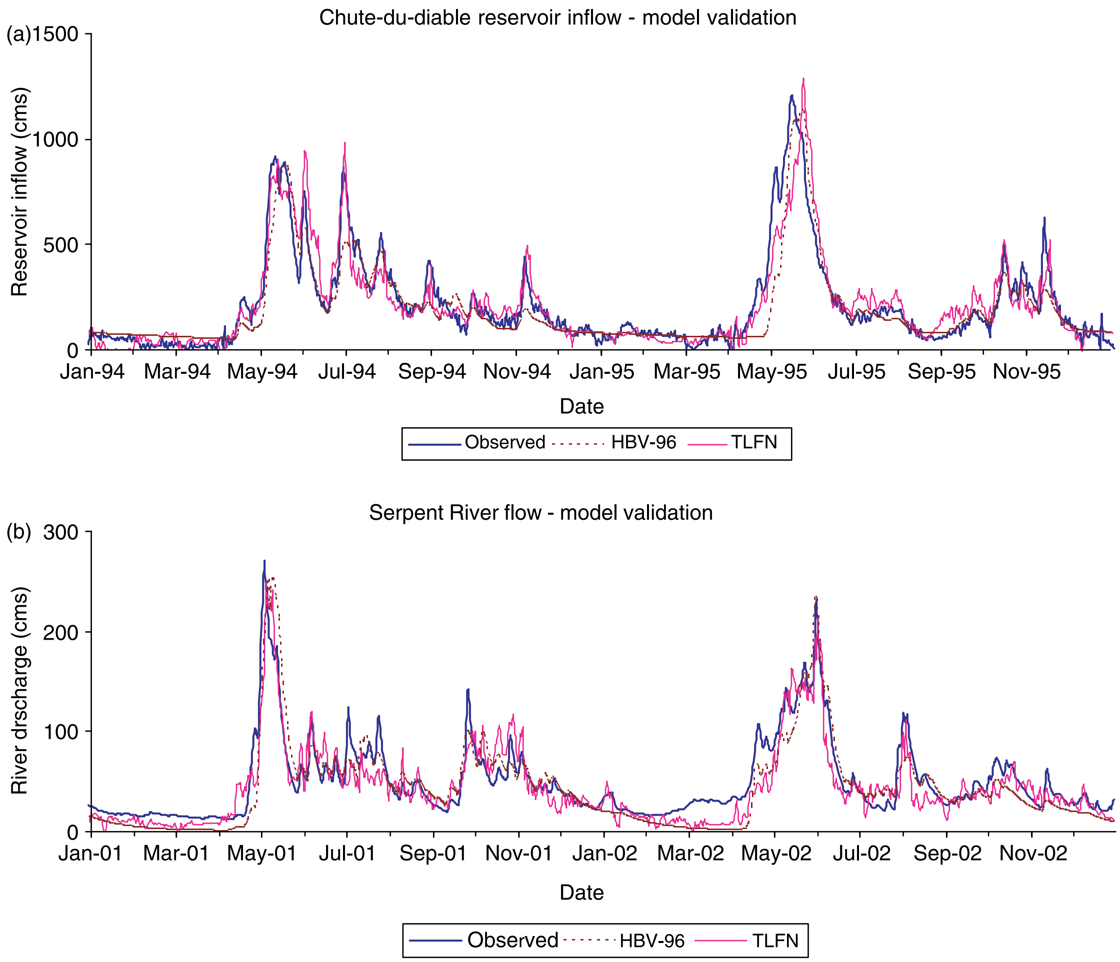


**Figure 5** | Comparisons of simulation performance by TLFN and HBV models for (a) reservoir inflow and (b) river flow.

model physical processes is often named *physically -based mathematical modeling* (or *knowledge-driven modeling*) because it tries to explain the underlying processes by some form of mathematical relationship. Therefore, in addition to the large quantity of basin information, the application of physically based precipitation–runoff models requires a thorough understanding of the underlying physical processes. Understanding of the linkage between the process controlling snow accumulation and melt is critical to developing a predictive ability to describe the response of a basin to changes in precipitation and energy inputs. In contrast, the so-called *data-driven models* are based on a limited knowledge of the process in question and rely on the available historical data describing input and output characteristics. Data-driven modeling techniques such as artificial neural networks (ANNs) are highly nonlinear and they need observed historical data to be trained. If implemented properly, data-driven models can perform the precipitation–runoff process in a cold and snowy climate based on only the historical precipitation, temperature and runoff data along with temporal indicators such as the delay lines and the logical values used herein. They are able not only to approximate the underlying relationship in the historical data, but also to generalize, providing a good approximation of the output for previously unseen inputs.

The main objective of this paper was to show how a time-lagged neural network can be applied for modeling such precipitation–runoff processes in areas with a cold and snowy climate. The research shows that identifying the appropriate input patterns and lag time, and adding logical information (in the form of symbolic inputs), are the crucial factors in the TLFN performance in this context. The performance of the TLFN is substantially improved with the addition of symbolic inputs which identify the month corresponding to each input pattern, and is believed to provide the TLFN with additional seasonal information needed for snowmelt generation. The study also shows that TLFN can be provided with long term memory in terms of time series of cumulative precipitation and long-term average temperature. The study results also suggest that the length of optimum long-term memory depends on the size of the drainage basin, with larger basins requiring longer memory lengths. Comparison of the optimal TLFN with the physically based hydrologic model shows that the data-driven TLFN models performed as good as the conceptual HBV hydrologic model both for the Chute-du-Diable reservoir inflow and Serpent River flow simulations. Moreover, the TLFN model simulates the high flows resulting from spring snowmelt very well, confirming its ability to implicitly account for the snowmelt processes in the precipitation–runoff modeling in a cold and snowy climate.

## ACKNOWLEDGEMENTS

This work was made possible through a grant from the Canadian Climate Change Action Fund, Environment Canada and a grant from the Natural Sciences and Engineering Research Council of Canada. The authors would like to thank the Aluminum Company of Canada (Alcan) for providing the experiment data. HBV-96 has kindly been made available by the Swedish Meteorological and Hydrological Institute (SMHI).

## REFERENCES

Arheimer, B. & Brandt, M. 1998 Modelling nitrogen transport and retention in the catchments of southern Sweden. *Ambio* **27**, 471–480.

ASCE Task Committee on Application of Artificial Neural Networks in Hydrology 2000 Artificial neural networks in hydrology. I: preliminary concepts. *ASCE J. Hydrol. Eng.* **5** (2), 115–123.

Campolo, M., Andreussi, P. & Soldati, A. 1999 Food forecasting with a neural network model. *Water Resour. Res.* **35** (4), 1191–1197.

Chiang, Y. M., Chang, L. C. & Chang, F. J. 2004 Comparison of static-feedforward and dynamic-feedback neural networks for rainfall-runoff modeling. *J. Hydrol.* **290**, 297–311.

Coulibaly, P., Anctil, F. & Bobée, B. 1999 Hydrological forecasting using artificial neural networks: the state of the art. *Can. J. Civ. Eng.* **26**(3), 293–304.

Coulibaly, P., Anctil, F., Rasmussen, P. & Bobée, B. 2000 A recurrent neural networks approach using indices of low-frequency climatic variability to forecast regional annual runoff. *Hydrol. Process* **14**, 2755–2777.

Coulibaly, P., Anctil, F., Aravena, R. & Bobée, B. 2001*a* ANN modeling of water table depth fluctuations. *Water Res. Res.* **37** (4), 885–896.

Coulibaly, P., Anctil, F. & Bobée, B. 2001*b* Multivariate reservoir inflow forecasting using temporal neural networks. *J. Hydrol. Eng. ASCE* **6** (5), 367–376.

Coulibaly, P., Dibike, Y. B. & Anctil, F. 2005 Downscaling precipitation and temperature with temporal neural networks. *J. Hydrometeorol.* **6** (4), 483–496.

Dawson, C. W. & Wilby, R. L. 2001 Hydrological modelling using artificial neural networks. *Prog. in Phys. Geog.* **25**(1), 80–108.

de Vries, B. & Principe, J. C. 1991 A theory of neural networks with time delay. In *Advances in Neural Information Processing Systems* (ed. J. E. Moody, R. P. Lippmann & D. S. Touretzky), Vol. 3, pp. 162–168.

Dibike, Y. B. & Coulibaly, P. 2005 Hydrologic impact of climate change in the Saguenay watershed: comparison of downscaling methods and hydrologic models. *J. Hydrol.* **307**, 145–163.

Dibike, Y. B. & Solomatine, D. 2001 River flow forecasting using artificial neural networks. *J. Phys. Chem. Earth B Hydrol. Oceans Atmos.* **26** (1), 1–8.

Dibike, Y. B., Solomatine, D. & Abbott, M. B. 1999 On the encapsulation of numerical-hydraulic models in artificial neural network. *J. Hydraul. Res.* **37** (2), 147–161.

Fuhrman, D. R. & Minns, A. W. 2000 Selection of input parameters for a hydrologic runoff model using a neural network. In: *Proc. Fourth Int. Conf. of Hydroinformatics*, Iowa City, IA.

Harlin, J. & Kung, C-S. 1992 Parameter uncertainty and simulation of design floods in Sweden. *J. Hydrol.* **137**, 209–230.

Liden, R. & Harlin, J. 2000 Analysis of conceptual rainfall-runoff modelling performance in different climates. *J. Hydrol.* **238**, 231–247.

Linsley, R. K., Kohler, M. A. & Paulhus, J. 1982 *Hydrology for Engineers*, 3rd edition. McGraw-Hill, New York.

Maier, H. R. & Dandy, G. C. 2000 Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications. *Env. Model. Soft.* **15**(1), 101–123.

Martinec, J., Rango, A. & Roberts, R. 1994 *The Snowmelt Runoff Model (SRM) User's Manual* (ed. M. F. Baumgartner),. Geographica Bernensia, Department of Geography, University of Berne, Berne, Switzerland.

Melloh, Rae A. 1999 *A Synopsis and Comparison of Selected Snowmelt Algorithms*. CRREL report, US Army Corps of Engineers, Hanover.

Minns, A. W. & Hall, M. J. 1996 Artificial neutral networks as rainfall-runoff models. *Hydrol. Sci. J.* **41** (3), 399–417.

Nash, J. E. & Sutcliffe, J. V. 1970 River flow forecasting through conceptual models, I. A discussion of principles. *J. Hydrol.* **10**, 282–290.

Principe, J. C., Euliano, N. R. & Lefebvre, W. C. 2000 *Neural and Adaptive Systems: Fundamentals through Simulations*. Wiley, New York.

See, L. M. & Kneale, P. E. 2004 Time delay neural networks for river level forecasting. In *Neural Networks for Hydrological Modelling* (ed. R. J. Abrahart, P. E. Kneale & L. M. See). Taylor & Francis, London, ch. 5.

Thirumalaiah, K. & Deo, M. C. 2000 Hydrological forecasting using neural networks. *J. Hydrol. Eng. ASCE* **5** (2), 180–189.

WMO 1986 *Intercomparison of Models of Snowmelt Runoff*. Operational Hydrology Report No. 23, WMO-No. 646 WMO, Geneva.

Zealand, C. M., Burn, D. H. & Simonovic, S. P. 1999 Short term streamflow forecasting using artificial neural networks. *J. Hydrol.* **214**, 32–48.

Zhang, B. & Govindaraju, R. S. 2003 Geomorphology-based artificial neural networks (GANNs) for estimation of direct runoff over watershed. *J. Hydrol.* **273**, 18–34.

First received 15 August 2006; accepted in revised form 8 June 2008