Baseflow separation techniques for modular artificial neural network modelling in flow forecasting

GERALD CORZO & DIMITRI SOLOMATINE
UNESCO-IHE Institute for Water Education, Westvest 7, Delft, 2611AX, The Netherlands
g.corzo@unesco-ihe.org

Abstract In hydrological sciences there is an increasing tendency to explore and improve artificial neural network (ANN) and other data-driven forecasting models. Attempts to improve such models relate, to a large extent, to the recognized problems of their physical interpretation. The present paper deals with the problem of incorporating hydrological knowledge into the modelling process through the use of a modular architecture that takes into account the existence of various flow regimes. Three different partitioning schemes were employed: automatic classification based on clustering, temporal segmentation of the hydrograph based on an adapted baseflow separation technique, and an optimized baseflow separation filter. Three different model performance measures were analysed. Three case studies were considered. The modular models incorporating hydrological knowledge were shown to be more accurate than the traditional ANN-based models.

Key words rainfall–runoff modelling; modular models; baseflow separation; hydrograph separation; artificial neural network forecasting; data-driven models

Techniques de séparation de l'écoulement de base pour la modélisation modulaire par réseau de neurones artificiels à vocation de prévision de débit

Résumé En sciences hydrologiques il y a une tendance croissante à explorer et améliorer les réseaux de neurones artificiels (RNA) et d'autres modèles de prévision conditionnés par les données. Les tentatives d'amélioration de tels modèles sont largement reliées aux problèmes reconnus de leur interprétation physique. Cet article traite du problème de l'incorporation de connaissance hydrologique dans le processus de modélisation grâce à l'utilisation d'une architecture modulaire qui tient compte de l'existence de divers régimes d'écoulement. Trois différents schémas de partition ont été employés: une classification automatique basée sur le regroupement, une segmentation temporelle de l'hydrogramme basée sur une technique adaptée de séparation de l'écoulement de base, et un filtre optimisé de séparation l'écoulement de base. Trois différentes mesures de performance de modélisation ont été analysées. Trois études de cas ont été considérées. Les modèles modulaires incorporant de la connaissance hydrologique se sont révélé être plus précis que les modèles traditionnels à base de RNA.

Mots clefs modélisation pluie–débit; modèles modulaires; séparation de l'écoulement de base; séparation d'hydrogramme; prévision par réseau de neurones artificiels; modèles conditionnés par les données

INTRODUCTION

Traditionally, modellers were, and often still are, trying to build a general, all-encompassing model of a studied natural phenomenon. Hydrological forecasting models that involve the use of data-driven techniques are not exceptions in this sense: they tend to be developed on the basis of using a comprehensive (global) model that covers all the processes in a basin (ASCE Task Committee on Application of Artificial Neural Networks in Hydrology, 2000; Dibike *et al.*, 1999; Abrahart & See, 2002; Dawson *et al.*, 2005). However, such models (very often these are artificial neural networks, ANN) do not encapsulate the knowledge that experts may have about the studied system, and in some cases suffer from low accuracy in extrapolation. In many applications of data-driven models, the hydrological knowledge is "supplied" to the model via the proper analysis of the input/output structure and choice of adequate input variables (Bowden *et al.*, 2005). However, much more can be done to incorporate domain knowledge into these models.

One of the ways of doing this is to try to discover different physically interpretable regimes of a modelled process (or sub-processes), and to build separate specialized ("local") models for each of them—either process (physically-based) models, or data-

Open for discussion until 1 December 2007

driven. Such an approach is seen as one of the ways of including hydrological knowledge and improving the model's performance. In order to combine the local models one may refer to the methods developed in machine learning (Haykin, 1999; Kuncheva, 2004) or, additionally, to enhance them by using a so-called fuzzy committee approach (Solomatine & Corzo, 2006). The latter paper also presents one possible classification of modular models.

Lately, a number of studies were reported where such an approach was undertaken (often being named differently, however). Solomatine & Xue (2004) applied an approach where separate ANN and M5 model-tree basin models were built for various hydrological regimes (identified on the basis of hydrological domain knowledge). Anctil & Tapé (2004) applied wavelet and Fourier transforms to the identification of high and low flows based on their frequency patterns. Some attempts have been made to find correlations between ANN components and processes in a conceptual model (Wilby *et al.*, 2003). Solomatine & Siek (2004) presented the M5flex algorithm where a domain expert is given more freedom in influencing the process of building a machine learning model. Wang *et al.* (2005) used a combination of ANNs for forecasting daily streamflow: different networks were trained on the data subsets determined by applying either a threshold discharge value, or clustering in the space of inputs (several lagged discharges, but no rainfall data, however). Jain & Srinivasulu (2006) also applied decomposition of the flow hydrograph by a certain threshold value and then built separate ANNs for low and high flow regimes.

All the studies demonstrated the higher accuracy of the modular models when compared to the models built to represent all possible regimes of the modelled system (such models are referred to herein as *global models*). It is also worth mentioning an approach that closely relates to the modular approach (and can even be seen as part of it), namely ensemble modelling and its variations. In it, several different models responsible for the whole process under question are built, an ensemble of the models is constructed and their outputs are combined. (Note that these models are global and not models of the sub-processes, as is the case in modular modelling.) Several authors presented and tested such an approach in hydrological forecasting. See & Openshaw (2000) used a hybrid multi-model approach to river flow forecasting; they combined artificial neural networks (ANNs), fuzzy rule-based systems and ARMA models in an ensemble using several averaging and Bayesian methods. Xiong *et al.* (2001) used a nonlinear combination of the forecasts of rainfall–runoff models using fuzzy logic. Abrahart & See (2002) performed a comprehensive study comparing six alternative methods to combine data-driven and physically-based hydrological models. Georgakakos *et al.* (2004) analysed the advantages of multi-model ensembles where each model is a hydrological distributed model with the same structure but different parameters. The ensemble approach has a number of advantages, but is not a topic of this paper.

If we want to follow the idea of a modular approach, there is the possibility to take a somewhat deeper view of the underlying sub-processes to be modelled for accurate flow forecasting. In basin modelling, a typical approach would be to identify baseflow and the direct runoff (also called excess runoff or excess flow). Such an approach was also taken in our earlier publication (Corzo & Solomatine, 2006), and this paper continues to develop it further.

The main idea that we apply is to use specialized algorithms for the hydrograph analysis to separate baseflow from excess flow, form training data sets and build local

ANN-based models for each component. The focus is on optimization of the model structures, and of the parameters of the data separation and combination algorithms. The paper is structured in the following way. First a general description of modular models and committee machines is given. Second, the separation schemes applied are presented and the performance criteria are defined. Third, the case studies are presented. Lastly, results are discussed and conclusions are drawn.

METHODOLOGY

Modular modelling using baseflow separation

The problem of hydrological modelling of a basin considered in this paper is characterized by precipitation and discharge measured at different moments in time in the past (which can be seen as multivariate time series), and the reaction of the basin represented by the forecast of the discharge (flow) hours or days ahead. Hydrology of a basin is typically modelled by physically-based models implementing the partial differential equations describing the water motion. An alternative way of modelling is to use data-driven models, i.e. ANNs. One of the common problems with ANNs is the seasonality which is accompanied by the overfitting and the misrepresentation of noisy data. Data-driven models are trained to be accurate on average across the whole span of the time series characterizing the output. In some regimes they may not generalize well (so-called local overfitting), while on the other hand, before a particular pattern is learned the model potentially could switch into another regime (so-called local unbefitting) (Weigend *et al.*, 1995).

The aforementioned modelling problems can be at least partly resolved by using modular models (Fig. 1), where for each regime a separate model is built. In computational intelligence, the overall model is often called a committee machine (Haykin, 1999). However, to stress the fact that each model is trained on separate data sets, in this paper we use the more appropriate term *modular model* (Osherson *et al.*, 1990).

As mentioned before, the main idea of modularization is based on baseflow separation (Fig. 2): instead of building one model responsible for representing the water flow for all regimes, two models are built. One model simulates the baseflow, and the other simulates the direct runoff or the total flow. These two models work sequentially: before time moment t_s, the first model simulates only baseflow; between

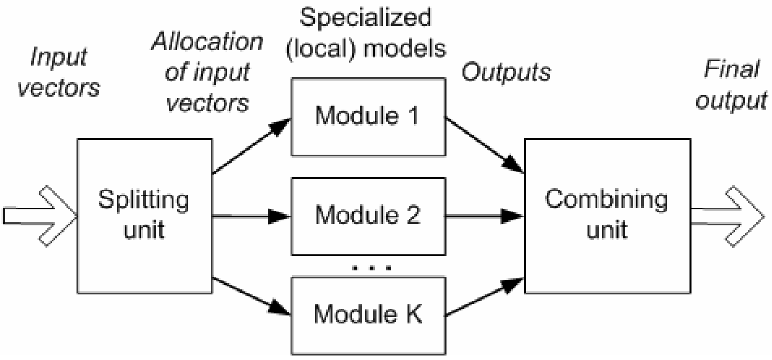

Fig. 1 Modular models based on local specialized models.

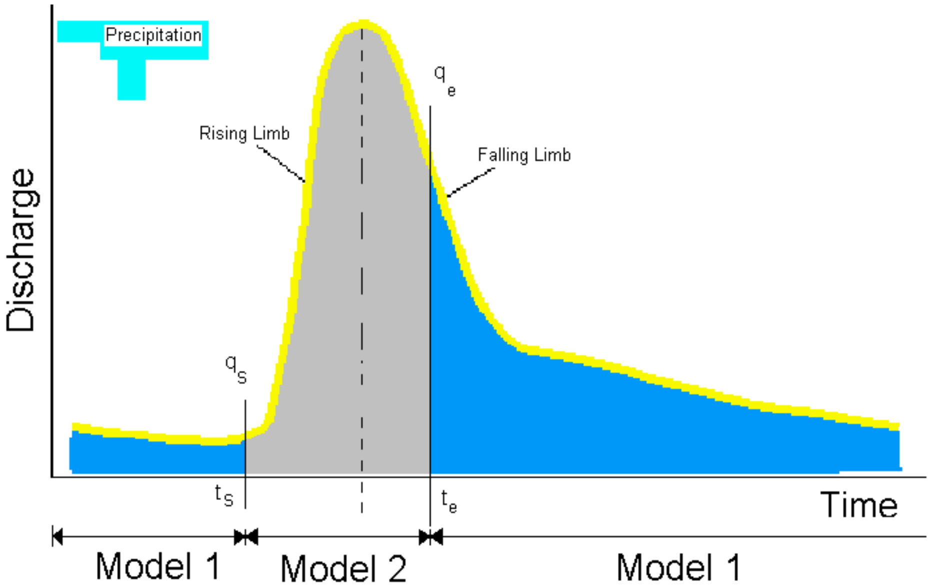

Fig. 2 Flow separation for local specialized models (t_s = beginning of storm, t_r = beginning of recession).

t_s and t_r the second one operates if total flow is modelled, or both of them operate if direct runoff is modelled; then the first model operates again. The problem here is to find the moments when the flow regime changes from baseflow only to the regime where direct flow is present as well. The detection of changes in state of a highly variable system has been studied in many areas (e.g. Valdés & Bonham-Carter, 2005).

In this research, three modular schemes are tested: cluster-based, time-based and process-based, which are described in the following sections. In this paper we follow the definitions of baseflow given by Hall (1968) for time-based separation, and by Chapman (2003) for process-based separation. Note that there are also other interpretations of how to define the baseflow (Beven, 2003; Uhlenbrook *et al.*, 2002).

Scheme MM1: clustering high and low flows

Under certain assumptions or conditions, the use of a clustering technique can be interpreted as an automatic identification of the ongoing regimes (Geva, 1999). There is no guarantee, of course, that a direct relationship between clusters and identifiable hydrological regimes would be discovered. In this paper, such conditions were not identified or checked—this is something to undertake in subsequent research. Clustering-related experiments should be seen as a demonstration of a possibility of such an approach.

A standard *k*-means algorithm was used to find groups of input vectors of discharge and precipitation (Späth, 1985). This algorithm is based on a two-phase iterative process, which minimizes the sum of point-to-centroid distances, summed over a number of clusters (*k*). At each iteration, points are assigned to their nearest cluster centre (chosen randomly at the very first iteration), followed by recalculation of cluster centres. The number of clusters has to be chosen *a priori*.

There is a wide variety of distance metrics to be used and for the purpose of this study the city-block (Manhattan) was selected. (We compared errors of the models built using several distance metrics and Manhattan appeared to result in the lowest error.) The distance metric used to build the cluster can be described as follows. Given an ($m \times n$) data matrix **X**, the various distances between the vector $\mathbf{x}_r$ (r = row) and $\boldsymbol{x}_s$ (s = row) are:

$$d_{rs} = \sum_{j=1}^{n} \left| \mathbf{x}_{rj} - \mathbf{x}_{sj} \right| \qquad (1)$$

Note that a separate classifier for data splitting has to be built to serve as a splitting unit during operation (Fig. 1); its purpose is to attribute new examples to an appropriate model. The training data for such a classifier is constructed in the following way: the input data are the same as the input data for models m1 and m2, and the output data are the class labels corresponding to the identified clusters.

The result of applying the described procedure is model MM1 consisting of:

(a) models m1 and m2 to model low flows and total flow, trained separately on the data subsets corresponding to the identified clusters;

(b) a classification model for data splitting.

Scheme MM2: time-based baseflow separation

Instead of grouping the input vectors, one may explicitly use hydrological domain knowledge to partition the data into groups that would be modelled separately. The flow of water through the basin is heterogeneous, follows various routes and, in hydrological analysis, it is often beneficial to identify the baseflow and the direct runoff (Hall, 1968; McCuen, 1998). Classical hydrograph baseflow separation analysis is in fact a graphical semi-empirical technique that splits the discharge values based on the measurement of discharge and precipitation (Fig. 2). In it, the values q_s and q_b of flow during the multiple storms are found, and the starting (t_s) and ending (t_b) of a storm phenomenon are identified.

The traditional "constant slope" method (McCuen, 1998) is manual: the beginning of the storm is identified as the point where the discharge is minimum, and the end of the storm corresponds to an inflection point. These two points are connected thus determining the sought baseflow area, and the slope of this line is recorded. Recently a number of other, simpler, methods have been introduced (Engel & Kyoung, 2005; Sloto & Cruise, 1996). However, in this paper, we use the "constant slope" method as the main foundation for building the separation algorithm. However, we made a slight modification that allowed for an easier algorithmic implementation: instead of looking for a hydrograph minimum, it is based only on the found inflection points. To connect to the point where the direct flow finishes, an imaginary line can be drawn from one inflection point to an inflection point at the end of the storm period (instead of connecting the minimum to an inflection point). The resulting line was found to be almost the same as the one identified by the traditional manual method.

The accurate determination of the inflection point by analysing the time series derivative estimates requires analysis of the complete event. Due to the high variability of data in a typical hydrological time series, the implementation of this method is not straightforward since there are too many inflection points. One way to resolve this is by smoothing of the time series and then identifying the inflection points by analysing the second derivatives.

However, our experiments have shown that such analysis will still result in finding many inflection points, and we are interested only in those that correspond to the beginning of a storm event. To remove such "false" points, we have chosen to use a method used by Sloto & Cruise (1996) (who, however, applied it to the minimum points rather than to inflection ones). Their idea was to remove points that lie on the hydrograph within the period of a storm event, but to identify this period is of course not a trivial problem. However, there are empirical methods known to do this, and Sloto & Cruise used the method of Linsley & Kohler (1982). The latter suggested that the average storm duration should be close to two times the number N of days between the peak flow and the end of the direct runoff. To assess N the following equation is used: $N = A^p$, where A is the basin area, and p varies depending on the basin characteristics. We used p = 0.2 based on the recommendations by McCuen (1998). This approach is not very accurate, but appears to work well for the purpose of the baseflow separation algorithm.

The time-based baseflow separation algorithm was implemented as follows:

1. **Smooth the data** This step is made to ease the identification of inflection points. A moving average filter is used; the span n of the filter can be changed according to the hydrological conditions of the case study (i.e. concentration time):

$$Q_{st} = \frac{1}{(2n+1)}\left(Q_{t+n} + Q_{t+n-1} + ... + Q_{t-n}\right) \qquad (2)$$

   where Q_t is original discharge; Q_{st} is the smoothed discharge; and n is the filter span.

2. **Find the inflection points** The inflection point is defined as the point where the second derivative of the discharge is zero, as follows:

$$\frac{\partial^2 Q_s}{\partial t^2} \approx \frac{\Delta^2 Q_s}{\Delta t^2} = \frac{\Delta}{\Delta t}\left(\frac{\Delta Q_s}{\Delta t}\right) = 0 \qquad (3)$$

3. **Remove the "false" inflection points** using the notion of the average storm duration, as described above.

4. **Separation** Finally, a (virtual) line is drawn between the inflection points, which separates the baseflow from direct runoff and thus separates the two regimes: one when only the baseflow is present and the other one when the baseflow is accompanied by the direct runoff (thus constituting the total flow). It also graphically represents the switching rule for the two predictive models. Algorithmically, a linear separation model is used.

   The baseflow separation method described above requires the data set corresponding to the whole storm. Such data are available during training (calibration), so the algorithm can be used to separate the data and train two separate models, which are referred to as m1 and m2.

   Note, however, that the algorithm cannot be applied during operation (and testing) since it needs future data characterizing the whole storm event. A solution is to train yet another data-driven model (i.e. a surrogate or meta-model) that replicates the baseflow algorithm. This model is referred to as MM2-IP (MM2-related model for Inflection Points), and it predicts the position of the limiting inflection points. The training data were generated by running the described baseflow algorithm on historical data. To then apply the time-based separation in operation, the algorithm should also include the following additional steps:

5. **Generate data for the MM2-IP model** Apply steps 1 to 3 to the historical data and generate enough data for training.

6. **Train the MM2-IP model** Use the generated data to train the MM2-IP model (a machine learning model, e.g. ANN) that would predict the location of inflection points in the operation phase of MM2.

The result of applying the described procedure is the MM2 model consisting of:

(a) models m1 and m2 trained separately to model baseflow and direct flow; and
(b) model MM2-IP that in the operation (or testing) phase identifies the inflection points used for data splitting.

Scheme MM3: process-based baseflow separation using optimized filters

The traditional baseflow separation methods mentioned above cannot be effectively used when separations are to be undertaken on a long continuous record of streamflows, rather than just a few storm period hydrographs. This has led to the development of a class of algorithms sometimes referred to as "numerical". Relatively recent research has applied flow separation filters that consider one or two variables in recursive algorithms (Arnold *et al.*, 2005; Chapman, 2003). However, these authors define baseflow slightly differently, assuming that even in the periods of low flow there are two components of flow which can be interpreted as direct runoff and baseflow. The method used in this study is based on the baseflow recursive filter (Ekhardt, 2005).

Ekhardt (2005) compared many of the existing baseflow filtering algorithms and proposed the following equation:

$$q_{b_t} = \frac{(1-\mathrm{BFI}_{\max})aq_{b(t-1)} + (1-a)\mathrm{BFI}_{\max}Q_t}{1-a\mathrm{BFI}_{\max}} \tag{4}$$

where q_{bt} is the baseflow at time step t; $q_{bt\text{-}1}$ is the baseflow at the previous time step; Q_t is the measured total flow; $\mathrm{BFI}_{\max}$ is a constant that can be interpreted as the maximum baseflow index; and a is a filtering coefficient, or recession constant. The three coefficients q_{b0}, $\mathrm{BFI}_{\max}$ and a are unknown, and there is no commonly accepted method to identify them. In principle, identification of coefficients is based on trial and error and sometimes it is possible to use the recession curve coefficients. In this paper, the three mentioned coefficients are found through an optimization process. The goal is to find such coefficients which ensure that the modular model has the best performance.

The MM3 model (Fig. 3) consists of two models: m1 for modelling the excess flow, and m2 for modelling the baseflow. The ultimate goal is to predict the total flow

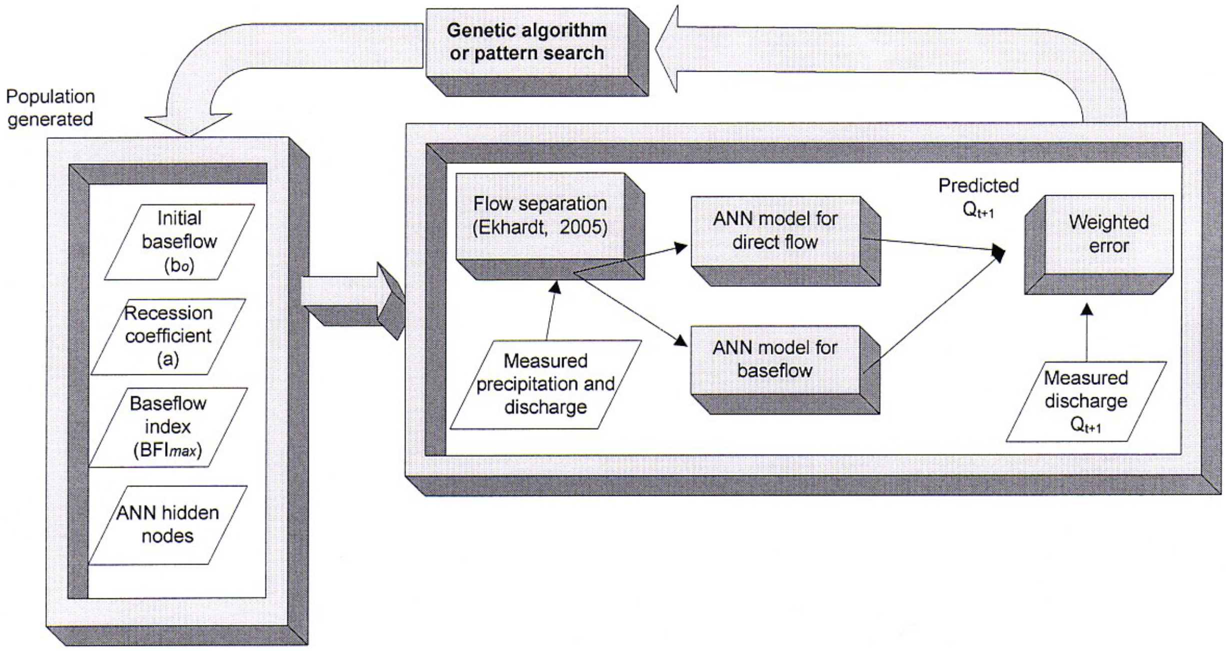
Genetic algorithm or pattern search
Population generated
Initial baseflow (b_0)
Recession coefficient (a)
Baseflow index (BFI_{max})
ANN hidden nodes
Flow separation (Ekhardt, 2005)
ANN model for direct flow
ANN model for baseflow
Measured precipitation and discharge
Predicted Q_{t+1}
Weighted error
Measured discharge Q_{t+1}

Fig. 3 Optimization of the process-based separation model (MM3).

Q_{t+1} at the next time step. The baseflow filter (equation (4)) separates the flow into two components and they are fed as inputs to the models. Both models must be trained on measured data and the model structure must also be optimized (e.g. finding the optimal number of hidden nodes in ANNs).

Since the baseflow cannot be measured, its values are approximated by equation (4). The three unknown parameters of this equation are found as a result of the optimization process in which the total model error (RMSE) is minimized. Optimization can be performed by any direct search method, for example, by randomized search, genetic algorithm (GA) or adaptive cluster covering (Solomatine *et al.*, 1999). In the present study we used a GA and pattern search (Abramson *et al.*, 2004). Note that during the search, in order to calculate the error (RMSE) for every new parameter vector, two ANNs are to be trained, so the process can be computationally expensive.

The objective function involves a weighted function with a high weight for the overall process and a low weight for the baseflow component:

$$E_T = w_0 \mathrm{RMSE}_{(Overall)} + w_1 \mathrm{RMSE}_{m1} + w_2 \mathrm{RMSE}_{m2} \tag{5}$$

where $m1$ and $m2$ refer to models of direct flow and baseflow, respectively; and w_i are the corresponding weights. The weights in equation (5) were selected on the basis of a (limited) number of experiments and take the values 0.6, 0.3 and 0.1, correspondingly. Such weights are justified by the assumption that our main objective is to forecast the flood situation; their values, however, could be subject to optimization as well.

Each calculation of the objective function (related to the model error to be minimized) involves the following steps:

– generate a random vector $\{b_0, \mathrm{BFI}_{\max}, a$, number of hidden nodes in each ANN$\}$;
– run equation (4) to perform the baseflow separation, generating two different training sets;

- train two different ANN models m1 and m2 of direct and baseflow respectively;
- calculate the overall error E_T using equation (5) (total flow is found as the sum of the models outputs).

The resulting MM3 model consists of:

(a) models m1 and m2 trained separately to model baseflow and direct flow; and
(b) an optimized numerical filter (equation (4)) calculating the baseflow that is used to split the data.

APPLICATION AND RESULTS

Case studies

Three basins, Bagmati in Nepal (B1), Sieve in Italy (B2) (Brath *et al.*, 2002; Solomatine & Dulal, 2003), and Brue in the UK (B3) (Moore, 2002) were considered as case studies (Table 1).

The size, location and other characteristics of the basins are significantly different, and this allowed for validating the presented modelling approach under different spatial and temporal forecasting conditions. The detailed hydrogeological description of these three basins and the detailed description of the global ANN models (i.e. the model trained on the whole data set) for cases B1 and B2 can be found in the references mentioned above. The choice of the input variables for global and modular ANN models was based on a correlation and mutual information analysis between the input and output variables. The variables chosen for the ANN models are shown in Table 2.

Table 1 General hydrological characteristics of the three basins.

Basin name	B1 (Bagmati)	B2 (Sieve)	B3 (Brue)
Topography	Mixed	Mountain	Mountain
River length (km)	170	56	20
Area (km^3)	3500	836	135.2
Data set: from	January 1988	December 1959	1 September 1993
to	December 1995	February 1960	30 August 1995
Training	1 January 1988– 22 June 1993	13 December 1959 19:00– 28 February 1960	September 1993– August 1994
Verification	23 June 1993– 31 December 1995	1 December 1959 07:00– 13 December 1959 18:00	September 1994– August 1995
Location	Nepal	Italy	England

Table 2 ANN model structures and training parameters.

Basin name	B1	B2	B3
Forecast period	1 day	1 hour	2 hours
Input variables	P_t, $P_{t\text{-}1}$, $P_{t\text{-}2}$, $Q_{t\text{-}1}$, Q_t	P_t, $P_{t\text{-}1}$, $P_{t\text{-}2}$, $P_{t\text{-}3}$, $P_{t\text{-}4}$, $P_{t\text{-}5}$, $Q_{t\text{-}1}$, $Q_{t\text{-}2}$, Q_t	P_t, $P_{t\text{-}1}$, $P_{t\text{-}2}$, $P_{t\text{-}3}$, $P_{t\text{-}4}$, $Q_{t\text{-}1}$, $Q_{t\text{-}2,}$ Q_t
Network structure GM	5-4-1	9-5-1	8-16-1

P = precipitation, Q = discharge.

Error metrics

For proper model assessment, various error metrics were applied. The two statistical measures most widely used, are the normalized root mean squared error (NRMSE, equation (6)) and the coefficient of efficiency (CoE, equation (7)). We also used two other measures: persistence index and the relative error.

$$\mathrm{NRMSE} = \frac{\sqrt{\frac{\mathrm{SSE}}{n}}}{\sigma_{\mathrm{obs}}} \tag{6}$$

where $\mathrm{SSE} = \sum_{t=1}^{n} \left(Q_{\mathrm{est},t} - Q_{\mathrm{obs},t}\right)^2$ and σ_{obs} is standard deviation of the observed value.

$$\mathrm{CoE} = 1 - \frac{\sqrt{\mathrm{SSE}}}{\sqrt{\sum_{t=1}^{n} (Q_{\mathrm{obs},t} - \overline{Q}_{\mathrm{obs},t})^2}}, \text{ where } \overline{Q}_{\mathrm{obs}} = \frac{\sum_{t=1}^{n} Q_{\mathrm{obs},t}}{n} \tag{7}$$

The persistence index (PERS) focuses on the relationship of the model performance and the performance of the naïve (“no-change”) model which assumes that the forecast at each time step is equal to the current value (Kitanidis & Bras, 1980):

$$\mathrm{PERS} = 1 - \frac{\mathrm{SSE}}{\mathrm{SSE}_{\mathrm{naive}}} \tag{8}$$

where $\mathrm{SSE}_{\mathrm{naive}} = \sum_{t=1}^{n} \left(Q_{\mathrm{obs},t+L} - Q_{\mathrm{obs},t}\right)^2$, which is a scaling factor based on the performance of the naïve model; $Q_{\mathrm{est},t}$ is the neural network forecast of the observed discharge $Q_{\mathrm{obs},t}$ at time t where $t = 1, 2, \ldots, n$; L is the lead time ($L = 1$ for one-day-ahead forecast); and n is the number of steps for which the model error is to be calculated.

A value of PERS < 0 means that the considered model is less worthy than the naïve model (i.e. it is degrading the provided information) while 0 < PERS < 1 indicates that the considered model is better than the naïve model (where the closer to 1 the better). Lauzon *et al.* (2006) suggest using PERS in cases when the discharge forecast is made on the basis of previous values.

Another useful error measure is the relative error (RE):

$$\mathrm{RE}_t = \frac{|\, Q_{\mathrm{obs},t} - Q_{p,t} \,|}{Q_{\mathrm{obs},t}} \times 100\% \tag{9}$$

The use of RE as an additional error measure is justified by the following. If RMSE or CoE are used, the same error value may be considered to be high in the low-flow season, and relatively low for the high-flow season. One solution could be to weigh the error values differently for different seasons, but such an approach will still depend on the objective identification of the low- and high-flow regions. Another solution is to use RE, which automatically takes into account the value of the measured variable, so that a value of RE corresponding to large absolute errors in the case of low flows is large while it will be relatively lower in the case of high flows.

In this study RE_t is used to identify the percentage of samples belonging to one of three groups: "low relative error" with RE less than 15%, "medium error" with RE between 15 and 35%, and "high error" with RE higher than 35%. The ranges were determined after experiments with the two trial models. The low error value is expected to cover possible measurement errors that could be around 20% (Beven, 2003).

Model setup

All ANN flow forecasting models were three-layer MLP with tangent transfer functions in the hidden and output layers. The input parameters were determined by the use of correlation analysis and average mutual information. The composition of the training and verification data sets was performed by analysing the statistical homogeneity. The training was done using the Levenberg-Marquardt algorithm; termination was based on cross-validation and on reaching the maximum number of epochs (150).

For initial clustering in MM1, we used the *k*-means algorithm. Classifiers in splitting models for MM1 and MM2 used an RBF ANN and the Fisher discriminant method, respectively. These were selected based on their performance. Regression trees were also tested to serve as classifiers but they were less accurate.

For the ANN MLP and RBF models, the MATLAB Neural Network toolbox was used. The Fisher discriminant algorithm method was based on the MATLAB Statistical Toolbox. Optimization of the MM3 model was based on the algorithms from the MATLAB Genetic and Direct Search Toolbox. A Pentium 4 3.2 GHz PC was used.

RESULTS AND DISCUSSION

The models were successfully trained and optimized. In training the MLP ANNs, 150 iterations appeared to be sufficient to reach convergence in all cases. For optimization, a GA and Pattern Search (Abramson *et al.*, 2004) were tested in all experiments. It appeared, however, that the GA was too slow, especially for basins B2 and B3 where it did not show any sign of convergence even after 24 hours of computation. The experience with the GA cannot be characterized as positive; however, this could be attributed to the details of its implementation in MATLAB and, probably, not enough effort invested in tuning its parameters. In the end all the results in all cases reported were achieved by the pattern search.

The calculated NRMSE values in verification for the different modular model structures are presented in Table 3. The modular models in the three case studies show variable performance. In the case of the more complex basin B1, with the largest area and largest forecast horizon, modular models improve on global models in relative terms more than for other basins. The mountainous region and the size make it a highly nonlinear system, and this implies that there is probably a large influence of baseflow components in the forecast streamflow and, consequently, higher importance of modelling it by a specialized model. The highest performance is shown by the MM3 model (NRMSE lower than that of the global ANN by almost 24%).

Table 3 Performance in verification of the different modular models and global models for each basin (NRMSE).

	GM	MM1	MM2	MM3	Naïve
B1	58.63	48.46	50.92	44.30	67.08
B2	7.65	12.02	6.54	6.31	13.86
B3	10.03	10.10	10.38	9.58	12.68

Table 4 Performance in verification of the different modular models and global models for each basin (CoE).

	GM	MM1	MM2	MM3
B1	0.66	0.77	0.74	0.80
B2	0.9942	0.9855	0.9957	0.9960
B3	0.9899	0.9898	0.9892	0.9908

Table 5 Persistence index in verification of the different modular models and global models for each basin (PERS).

	GM	MM1	MM2	MM3
B1	0.23	0.47	0.41	0.51
B2	0.69	0.24	0.78	0.79
B3	0.38	0.37	0.33	0.43

Basins B2 and B3 are small in size with a relatively fast response. In general, it can be said that these catchments were modelled with high accuracy by both global and modular models. High accuracy makes it difficult to compare the models using CoE which is very close to 1 (Table 4). Nevertheless it is clear that the use of hydrological knowledge in flow separation gives good results for these basins as well. Once again the MM3 models show the largest reduction in NRMSE compared with the global models for these basins.

The results show that the error of the modular and the global models are less than that of the naïve model. The naïve model is the simplest solution and could be interpreted as a measure of the simplest form of linearity in the time series. All other models include the precipitation as an input variable (which the naïve model does not), so it is not surprising that they have better performance. It is also worth noting that the relatedness (measured by correlation) between the precipitation and the future values of discharge is variable and our experiments (not presented here) show that it depends on the different seasons (since under different flow conditions soil moisture and the time lags are also different). This prompts the idea of using the different model structures for different seasons—something to undertake in future research.

In terms of the coefficient of persistence (PERS, Table 5), the results are consistent with the NRMSE and CoE. MM3 outperforms all other models in all three case studies. The persistence index for the MM3 is near or above 0.5, showing a significant increase in performance over the naïve predictor. In terms of PERS, MM2 is better than the global model for the B1 and B2 basins; however, results for the B3 basin are the worst amongst all the models.

The percentages of samples in the three different relative error (RE) groups are shown in Fig. 4. Using more than one error metric in the analysis makes it possible to

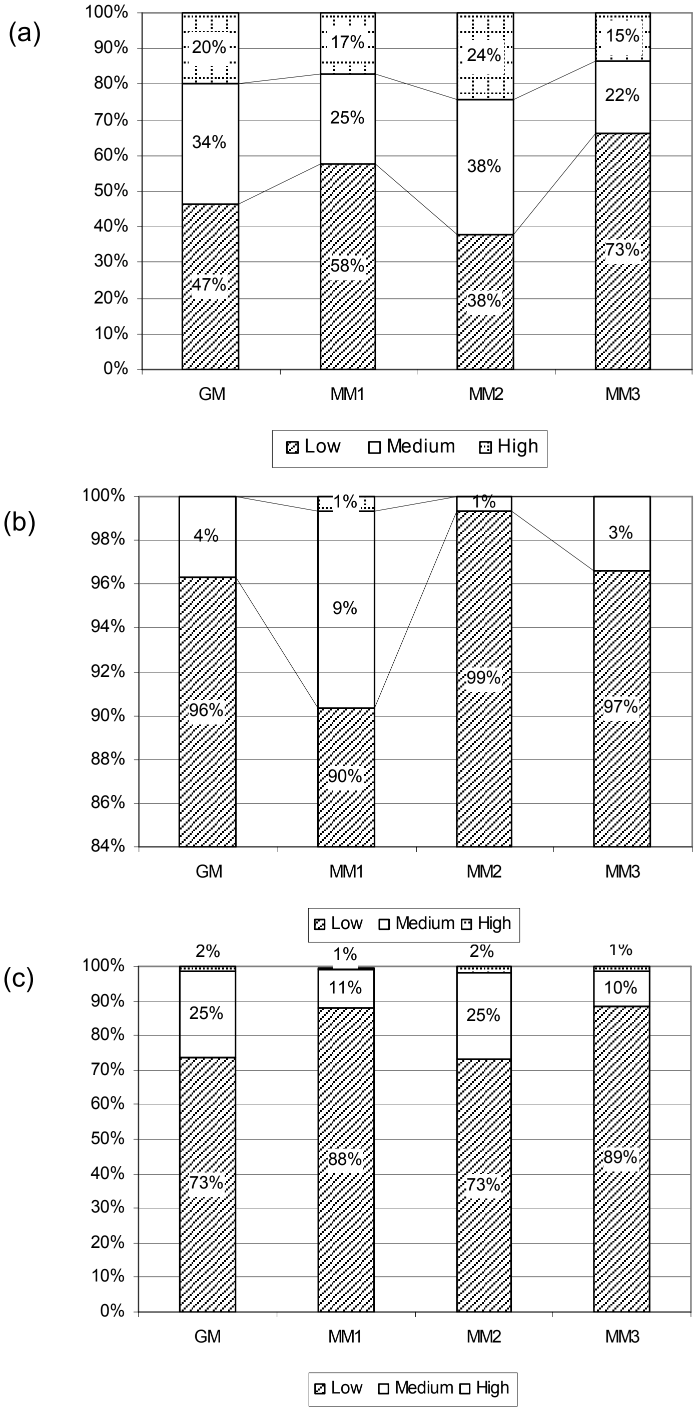

Fig. 4 Relative error level in percentage for: (a) the Bagmati River basin (B1); (b) the Sieve River basin (B2); and (c) the Brue River basin (B3).

evaluate better the performance of models for various hydrological regimes. The use of the additional error measure, RE, can be justified, for example, by the following observations: NRMSE of MM2 for B1 is less than that of the GM, but at the same time there are fewer samples with low RE than for the GM. This seems contradictory, but the NRMSE squares the absolute error so the high flow samples with low RE may have high absolute error, which, being squared, will contribute considerably to the total NRMSE. At the same time, since their RE is low, they would be attributed to the "low RE" group which would contain a large number of such examples. This is what happened with the samples predicted by GM for which the "low RE" group appeared to be larger than for MM2.

For basin B2 there is an increase in the low relative error percentages for models MM2 and MM3 (Fig. 4(b)). This is consistent with the NRMSE measure. In this case the MM2 model shows a more precise and accurate result having 99% of the sample with a low RE.

Figure 5 shows a fragment of a hydrograph generated by MM3 for B2 using test data. This model has been optimized, so that BFI_{max} is 0.25 and the recession coefficient *a* is 0.96. Indeed, for the fast response basins, values of this order are expected: the BFI should be small, and the recession coefficient should be high due to the relatively high slope in the recessions.

In contrast to B2, the B1 catchment in Himalaya, which is large, has a high groundwater storage. This basin has a BFI_{max} of 0.95 and coefficient *a* of 0.23. Note that BFI_{max} as defined by Ekhardt (2005; equation (6)) is the maximum value of the baseflow index (BFI) which is defined as the total volume of baseflow divided by the total volume of runoff for a period of time (Wahl & Wahl, 1995). A value of 0.23 therefore does not mean that the volume of baseflow is 23% of the total volume.

In general, the modular model MM3 outperforms the global model; this can also be illustrated graphically as in Fig. 5 with a typical fragment of the hydrograph from Fig. 6. Figure 5 shows that the baseflow for this catchment does not have a high contribution. This may actually explain why the accuracy of the modular model (where baseflow is modelled separately in this case) is not much higher than that of the global one. Another reason, of course, is that the GM is already very accurate.

An interesting question to ask is why the MM3 algorithm results in a better performance than that of MM1 and MM2. One may conclude that this can be attributed to the fact that Ekhardt's filter is a better device to identify the baseflow, so the MM3 model is therefore better than the other models. However, the better performance of MM3 may also be a result of other factors, so that further analysis is required. A more general question is whether the flow components identified by the separation algorithms really do correspond to different sub-processes (which we tried to model separately), or do these algorithms produce "baseflow" while not necessarily representing a clearly identifiable sub-process? One may argue, however, that accurate separation of sub-processes corresponding to base and excess flow (which are currently defined in a quite approximate fashion, and differently by different authors) is simply not possible in principle. However, answering these interesting questions is beyond the scope of this paper.

CONCLUSIONS

In this study, an attempt to introduce more knowledge about the hydrological processes into data-driven modelling was undertaken. The existence of two flow regimes was considered (base- and excess flow), and instead of training a global model on the whole data set, the training set was partitioned into two subsets, and two local

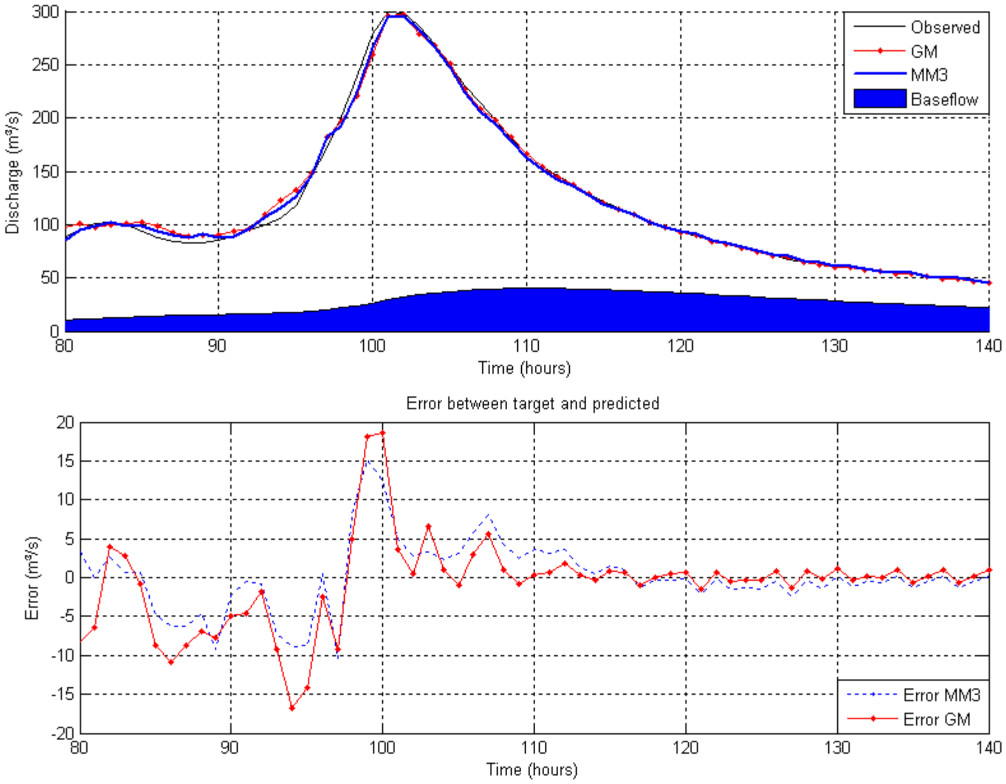

Fig. 5 MM3 performance in basin B2 (with the baseflow component shown). Error indicated is the difference between the predicted and measured discharges.

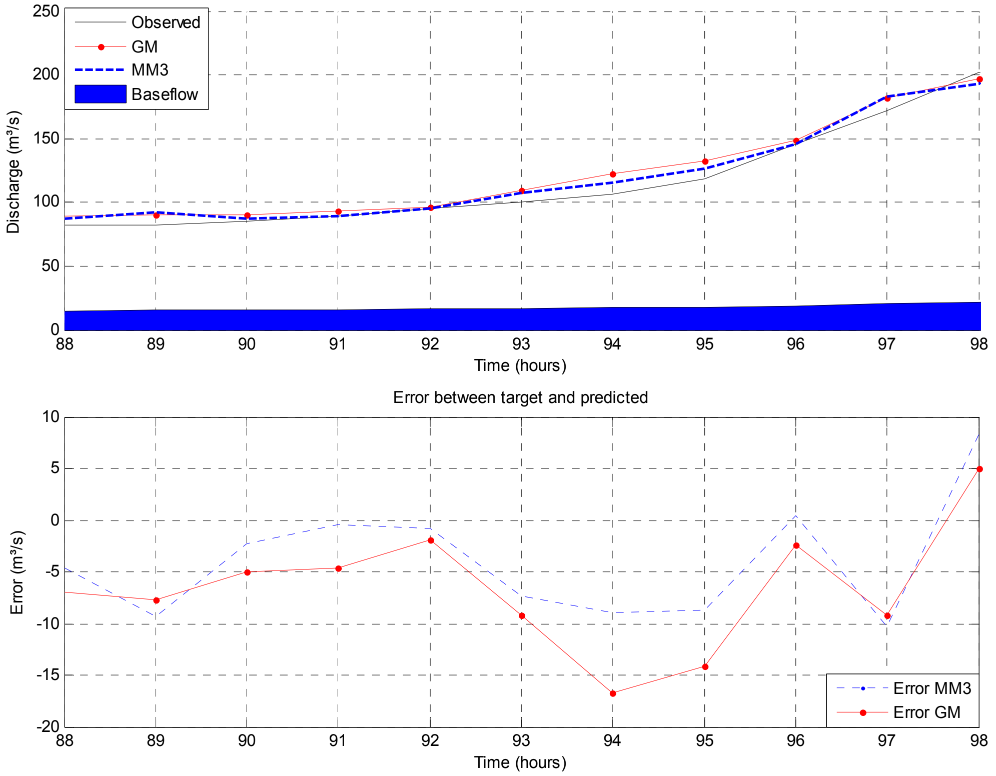

Fig. 6 Performance comparison of MM3 and GM models in basin B2 (fragment). Error indicated is the difference between the predicted and measured discharges.

models, each responsible for a particular hydrological regime, were built. Three different partitioning schemes were employed. In one of them, the baseflow separation filter of Ekhardt (2005) was used for flow separation, and two different ANN models were trained, one for each flow component. This last modular model scheme was complemented by a randomized search optimization of the coefficients in the baseflow filtering equation.

The use of domain knowledge in the modelling framework presented proved to be beneficial. Even the traditional semi-empirical flow separation algorithms, such as constant slope, can add to the accuracy of data-driven hydrological models. Partitioning the data by clustering gave good results only in some of the basins. Such partitioning is simple, but does not directly relate to hydrological regimes and is highly sensitive to the distance measure in the clustering. The best model overall is the one using the optimized baseflow filtering equation. In general, it can be concluded that the use of domain knowledge with a modular approach is effective in predictive modelling in a hydrological context.

There are several research issues that are to be addressed (and which are already being addressed in the ongoing research): the proper "soft" combination of the modelling modules especially at the transition point from one regime to another; complementing hydrological knowledge by the routines for automatic identification of regimes (for example, using the apparatus of hidden Markov models); modularization of physically-based models (following, for example, an approach outlined in Solomatine, 2006); and combining data-driven and physically-based models.

Acknowledgements Part of this work was performed in the framework of the Delft Cluster research programme (project "Safety from flooding") supported by the Dutch government. We also wish to thank Professors M. Franchini, E. Todini and S. Marsili-Libelli for providing data and materials on the Sieve basin, Durga Lal Shrestha and Khada Dulal for providing data on the Bagmati basin, and the British Atmospheric Data Centre and the HYREX project for providing data on the Brue basin.

REFERENCES

Abrahart, R. J. & See, L. (2002) Multi-model data fusion for river flow forecasting: an evaluation of six alternative methods based on two contrasting catchments. *Hydrol. Earth System Sci.* **6**(4), 655–670.

Abramson, M. A., Audet, C. & Dennis, J. E. (2004) Generalized pattern searches with derivative information. *Mathematical Programming* **100**(1), 3–25.

Anctil, F. & Tape, D. G. (2004) An exploration of artificial neural network rainfall–runoff forecasting combined with wavelet decomposition. *J. Environ. Engng Sci.* **3**(1), 121–128.

Arnold, J. G., Allen, P. M., Muttiah, R. & Bernhardt, G. (2005) Automated baseflow separation and recession analysis techniques. *Ground Water* **33**, 1010–1018.

ASCE Task Committee on Application of Artificial Neural Networks in Hydrology (2000) Artificial neural networks in hydrology. II: Hydrologic application. *J. Hydrol. Engng* **5**(2), 124–136.

Beven, K. J. (2003) *Rainfall–Runoff Modelling. The Primer*. Wiley, Chichester, UK.

Bowden, G. J., Dandy, G. C. & Maier, H. R. (2005) Input determination for neural network models in water resources applications. Part 1—background and methodology. *J. Hydrol.* **301**, 93–107.

Brath, A., Montanari, A. & Toth, E. (2002) Neural networks and non-parametric methods for improving real-time flood forecasting through conceptual hydrological models *Hydrol. Earth System Sci.* **6**, 627–639.

Chapman, T. G. (2003) Modelling stream recession flows. *Environ. Modell. Software* **18**, 683–692.

Corzo, G. A. & Solomatine, D. P. (2006) Optimization of base flow separation algorithm for modular data-driven hydrologic models. Proc. 7th International Conference on Hydroinformatics, Nice, France.

Dawson, C. W., See, L. M., Abrahart, R. J., Wilby, R. L., Shamseldin, A. Y., Anctil, F., Belbachir, A. N., Bowden, G., Dandy, G. & Lauzon, N. (2005) A comparative study of artificial neural network techniques for river stage forecasting. In: *Proc. 2005 IEEE Int. Joint Conf. on Neural Networks* (Vancouver, Canada).

Dibike, Y. B. & Abbott, M. B. (1999) Application of artificial neural networks to the simulation of a two dimensional flow. *J. Hydraul. Res*. **37**(4), 435–446.

Ekhardt, K. (2005) How to construct recursive digital filters for baseflow separation. *Hydrol. Processes* **19**, 507–515.

Engel, B. A. & Kyoung, L. J. (2005) WHAT (Web-Based Hydrograph Analysis Tool). Purdue University and USGS. http://pasture.ecn.purdue.edu/~what/open_web_gis_interface.html.

Georgakakos, K. P., D. Seo, H. Gupta, J. Schaake & Butts, M. B. (2004) Towards the characterization of streamflow simulation uncertainty through multimodel ensembles. *J. Hydrol*. **298**, 222–241.

Geva, A. B. (1999) Non-stationary time-series prediction using fuzzy clustering. In: *Proc. 18th International Conference of the North American Fuzzy Information Processing Society* (NAFIPS), 413–417.

Hall, F. R. (1968) Base-flow recessions review. *Water Resour. Res.* **4**, 973–983.

Haykin, S. (1999) *Neural Networks: A Comprehensive Foundation*. Prentice Hall, Englewood Cliffs, New Jersey, USA.

Jain, A. & Srinivasulu, S. (2006) Integrated approach to model decomposed flow hydrograph using artificial neural network and conceptual techniques. *J. Hydrol.* **317**, 291–306.

Kitanidis, P. K. & Bras, R. L. (1980) Real-time forecasting with a conceptual hydrologic model: analysis of uncertainty. *Water Resour. Res.* **16**(6), 1025–1033.

Kuncheva, L. I. (2004) *Combining Pattern Classifiers: Methods and Algorithms*. Wiley, Chichester, UK.

Lauzon, N., Anctil, F. & Baxter, C. W. (2006) Clustering of heterogeneous precipitation fields for the assessment and possible improvement of lumped neural network models for streamflow forecasts. *Hydrol. Earth System Sci.* **10**, 485–494.

Linsley, R. K., Kohler, M. A. & Paulhus, J. L. (1982) *Hydrology for Engineers*. McGraw-Hill, New York, USA.

McCuen, R. H. (1998) *Hydrologic Analysis and Design*. Prentice-Hall, Inc., Englewood Cliffs, New Jersey, USA.

Moore, B. (2002) Special Issue: HYREX: the Hydrological Radar Experiment. *Hydrol. Earth System Sci.* **4**(4), 521–522.

Osherson, D. N., Weinstein, S. & Stoli, M. (1990) *Modular Learning. Computational* Neuroscience, 369–377. MIT Press, Cambridge, Massachusetts, USA.

See, L. & Openshaw, S. (2000) A hybrid multi-model approach to river level forecasting. *Hydrol. Sci. J.* **45**(4), 523–535.

Sloto, R. A. & Cruise, M. (1996) HYSEP: a computer program for streamflow hydrograph separation and analysis. US Geol. Survey.

Solomatine, D. P. (2006) Optimal modularization of learning models in forecasting environmental variables. In: *Proc. iEMSs Third Biennial Meeting: "Summit on Environmental Modelling and Software"* (ed. by A. Voinov, A. Jakeman & A. Rizzoli) (Burlington, USA). http://www.iemss.org/iemss2006/papers/w7/342_solomatine_1.pdf.

Solomatine, D. P. & Dulal, K. N. (2003) Model tree as an alternative to neural network in rainfall–runoff modelling. *Hydrol. Sci. J.* **48**(3), 399–411.

Solomatine, D. P. & Siek, M. B. (2004) Semi-optimal hierarchical regression models and ANNs. In: *Proc. Int. Joint Conf.on Neural Networks* (Budapest, Hungary).

Solomatine, D. P. & Xue, Y. (2004) M5 model trees compared to neural networks: application to flood forecasting in the upper reach of the Huai River in China. *J. Hydrol. Engng* **9**(6), 491–501.

Solomatine, D. P. & Corzo, G. A. (2006) Learning hydrologic flow separation algorithm and local ANN committee modelling. In: *Proc. 2005 Int. Joint Conf. on Neural Networks* (Vancouver, Canada).

Solomatine, D. P., Dibike, Y. & Kukuric, N. (1999) Automatic calibration of groundwater models using global optimization techniques. *Hydrol. Sci. J.* **44**(6), 879–894.

Späth, H. (1985) *The Cluster Dissection and Analysis Theory FORTRAN Programs Examples*. Prentice-Hall, Inc., Upper Saddle River, New Jersey, USA.

Uhlenbrook, S., Frey, M., Leibundgut, C. & Maloszewski, P. (2002) Hydrograph separations in a mesoscale mountainous basin at event and seasonal timescales. *Water Resour. Res.* **38**(6), 31.1–14.

Valdés, J. & Bonham-Carter, G. (2005) Time dependent neural network models for detecting changes of state in Earth and planetary processes. In: *Proc. 2005 IEEE Int. Joint Conf. on Neural Networks* (Vancouver, Canada).

Wahl, K. L. & Wahl, T. L. (1995) Determining the flow of Comal Springs at New Braunfels. In: *Texas Water '95* (A Component Conference of the First International Conference on Water Resources Engineering), 77–86. American Society of Civil Engineers, Texas, USA.

Wang, W., Gelder, P., Vrijling, J. K. & Ma, J. (2006) Forecasting daily streamflow using hybrid ANN models. *J. Hydrol.* **324**(1), 383–399.

Weigend, A. S., Mangeas, M. & Srivastava, A. N. (1995) Nonlinear gated experts for time series: discovering regimes and avoiding overfitting. *Int. J. Neural Systems* **6**(4), 373–399.

Wilby, R. L., Abrahart, R. J. & Dawson, C. W. (2003) Detection of conceptual model rainfall–runoff processes inside an artificial neural network. *Hydrol. Sci. J.* **48**(2), 163–181.

Xiong, L., Shamseldin, A. Y. & O'Connor, K. M. (2001) A non-linear combination of the forecasts of rainfall-runoff models by the first-order Takagi-Sugeno fuzzy system. *J. Hydrol.* **245**(1), 196–217.

Received 11 December 2006; accepted 8 March 2007