Quantification and Reduction of Predictive Uncertainty for Sustainable Water Resources Management (Proceedings of Symposium HS2004 at IUGG2007, Perugia, July 2007). IAHS Publ. 313, 2007.

RNN-based inflow forecasting applied to reservoir operation via implicit stochastic optimization

CAMILO ALLYSON SIMÕES DE FARIAS[1], AKIHIRO KADOTA[1], ALCIGEIMES B. CELESTE[2] & KOICHI SUZUKI[1]

1 *Dept of Civil and Environmental Engineering, Ehime University, 3 Bunkyo-Cho, Matsuyama, Ehime 790-8577, Japan*
camiloallyson@yahoo.com.br

2 *Dept of Civil Engineering, Federal Univ. of Campina Grande, Aprígio Veloso 882, Campina Grande 58109-970, PB, Brazil*

Abstract A Recurrent Neural Network (RNN) is proposed for monthly reservoir inflow forecasting. In order to verify its performance and applicability, the network is used to assist reservoir operations carried out by Implicit Stochastic Optimization (ISO). The ISO approach defines the release at each month conditioned on the month's initial storage and the forecasted inflow for the month. This inflow is determined by the RNN. For comparison, optimal ISO-based releases assuming the inflows as perfect forecasts are also conducted. The RNN estimates the current-period inflow as a function of the previous inflow and current forecasted rainfall. The excellent accuracy obtained by the RNN suggests that it is very effective for one-month-ahead forecasting of reservoir inflows. Furthermore, the optimal reservoir releases obtained by the ISO using the RNN-based forecasts were shown to be highly correlated with those using perfect forecasts and superior to those obtained by standard rules of operation.

Key words inflow forecasting; recurrent neural networks; reservoir operation

INTRODUCTION

Matsuyama is a city situated in the southwestern part of Japan which has suffered from several problems relating to water shortages. Recently, many computational techniques have been used in order to improve and find sustainable operating policies for the reservoir that supplies the city. Farias *et al*. (2006) and Celeste *et al*. (2005) successfully applied Implicit Stochastic Optimization (ISO) together with numerical interpolation and artificial neural networks (ANNs) to derive operating rules for the reservoir.

The ISO procedure applied to Matsuyama City used a deterministic optimization model to find optimal reservoir releases over an operating horizon assuming a particular sequence of reservoir inflows. The ensemble of optimal releases was then related to current reservoir storage and projected inflow by means of numerical interpolation or ANNs. The relationships obtained could be thus used to obtain the reservoir releases at any time given the present storage and forecasted inflow (Farias *et al*., 2006; Celeste *et al*., 2005).

This study aims at proposing a recurrent neural network (RNN) model for monthly reservoir inflow forecasting to be used together with the interpolation-based ISO procedure. RNNs are dynamic artificial neural networks capable of accounting for nonlinearities and representing the temporal information of input sequences (Coulibaly *et al*., 2001; Elman, 1990). The RNN architectures have recurrent connections that implicitly allow the network to detect and produce time-varying patterns, which makes them very suitable for the prediction of water resource time series. The process consists of generating the current-month inflow by the RNN and using it as the input required by the ISO procedure to define the reservoir release for the month.

DETERMINISTIC OPTIMIZATION MODEL

It is assumed that the main objective of the operation is to find the allocations of water that best satisfy the respective demands without compromising the system. Another aim is to keep the storage high whenever possible, i.e. every time there are alternative optimal solutions for the releases. The objective function of the optimization problem is thus written as follows:

$$\min \sum_{t=1}^{N} \left\{ \alpha_R \left[\frac{R(t) - D(t)}{D(t)} \right]^2 + \alpha_S \left[\frac{S(t) - S_{\max}}{S_{\max}} \right]^2 \right\} \tag{1}$$

where t is the time index; N is the operating horizon; α_R is the priority coefficient for the first term of the objective function; α_S is the priority coefficient for the second term of the objective function; $R(t)$ is the release during period t; $D(t)$ is the demand during period t; $S(t)$ is the reservoir storage at the end of time interval t; and $S_{\max}$ is the storage capacity of the reservoir.

Release and storage at each period are related to inflow and spill through the continuity equation:

$$\begin{aligned} S(1) &= S_0 + I(1) - R(1) - Sp(1) \\ S(t) &= S(t-1) + I(t) - R(t) - Sp(t); \quad t = 2, \ldots, N \end{aligned} \tag{2}$$

in which S_0 is the initial reservoir storage; $I(t)$ is the inflow during time t; and $Sp(t)$ is the spill that eventually might occur during time t.

The physical limitations of the system define intervals which release, storage and spill must belong to:

$$0 \le R(t) \le \min[D(t), R_{\max}]; \quad \forall t \tag{3}$$

$$S_{\text{dead}} \le S(t) \le S_{\max}; \quad \forall t \tag{4}$$

$$Sp(t) \ge 0; \quad \forall t \tag{5}$$

where $R_{\max}$ is the maximum possible release and S_{dead} is the dead storage.

Since the objectives are extremely conflicting, the priority coefficient α_R was kept superior to α_S (α_R = 100 and α_S = 0.001). As a consequence, the release was given priority over the option of maintaining reservoir storage close to the maximum level.

The optimization model developed is formulated as a quadratic programming (QP) problem with linear constraints. The model is implemented in MATLAB and solved by a quadratic programming procedure that is part of the MATLAB Optimization Toolbox (Optimization Toolbox User's Guide, 2000).

IMPLICIT STOCHASTIC OPTIMIZATION PROCEDURE

The ISO procedure has the three basic steps described below:
(a) Generate M synthetic N-month sequences of inflow.
(b) For each inflow realization, find the optimal releases for all N months using the deterministic optimization model with equations (1)–(5).
(c) Use the ensemble of optimal releases ($M \times N$ data) to develop operating rules for each month of the year.

The releases obtained by the optimization model, $R(t)$, are related to reservoir storage at the end of the previous time period, $S(t - 1)$, and the inflow during the current time period, $I(t)$. One relationship (rule) is determined for each month of the year. Therefore, with information regarding initial reservoir storage and the forecasted inflow for the current month, the amount of water that should be released can be defined by the particular rule.

The relationships are established by surface graphs which are fitted to the data via numerical interpolation. Thus, the release for any condition of storage and inflow can be found by accessing the corresponding surface. It should be noted that no equation is necessary and the allocations are determined only through interpolation.

Like the optimization model defined by equations (1)–(5) and the ISO algorithm, the surface fitting procedure was constructed in MATLAB and is founded on triangle-based cubic interpolation and Delaunay triangulation.

RECURRENT NEURAL NETWORK MODEL

A recurrent artificial neural network trained by the back-propagation algorithm is proposed for monthly reservoir inflow forecasting. The RNN of this study consists of the well-known Elman network (Elman, 1990). In this model, the network has feedback connections from the output of each hidden neuron to its input, which provide to the network a dynamic memory. These recurrent connections allow the RNN to implicitly detect and produce time-varying patterns, making them very suitable for time series modelling.

Architecture and topology

The architecture of the network is formed by the input layer, one hidden layer, context units (elements that receive the values from the recurrent connections) and the output layer. The input layer is composed of three neurons, which are: previous inflow, current-period forecasted rainfall, and a dummy variable for identifying the current month. The number of neurons in the hidden layer and context units is determined based on a trial-and-error procedure. The best training results were achieved with 24 neurons in the hidden layer and therefore 24 context units. The current inflow is the single neuron of the output layer.

The difference between Elman's and conventional networks is in the fact that the first has recurrent connections from the hidden layer output to its input. The delay in the recurrent connections stores information of previous time steps and consequently is capable of learning temporal patterns in the sequential inputs (Elman, 1990).

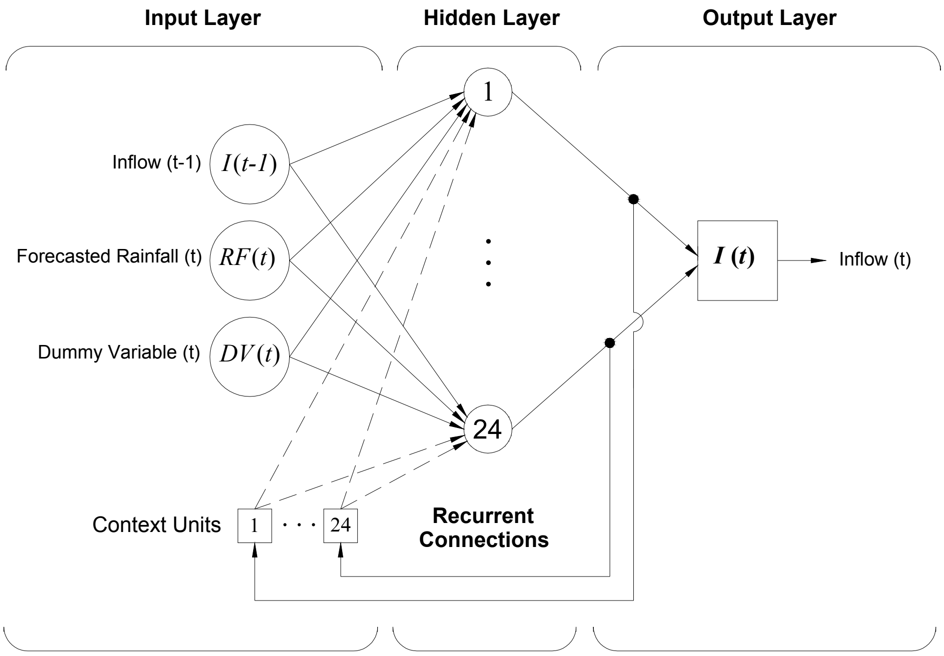

Fig. 1 Topology of the RNN.

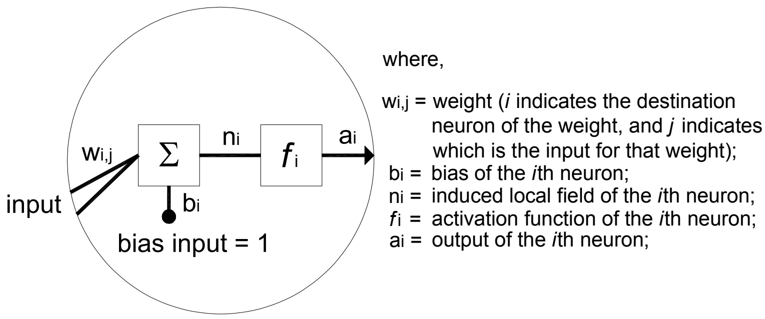

Fig. 2 Details of a neuron in the hidden layer.

Figure 1 illustrates the network topology of this study and Fig. 2 the details of a neuron in the hidden layer.

Activation functions

The tan-sigmoid function is chosen as the activation function for the hidden neurons. For the output layer neuron, a linear activation function is used.

Training process

The original data (input and desired outputs) are conveniently scaled before the training in order to improve the efficiency of the RNN. The scaling approach consists of normalizing the inputs and targets so that they will have a mean and standard deviation equal to zero and one, respectively (Demuth & Beale, 2005).

The training is performed by a back-propagation algorithm which has been successfully applied to water resources systems (Haykin, 1999). In this approach, the Scaled Conjugate Gradient (SCG) method is used for the back-propagation. A detailed explanation of the SCG method is provided by Moller (1993). The network training is supervised, i.e. the series of weights between the neurons and the bias are adjusted through the iterations (epochs) in order to fit the series of inputs to another series of known outputs. The training also occurs in the batch mode. In this mode, the weights and biases are updated only after the entire training set has been applied to the network. In order to improve generalization, the training is stopped by the Early Stopping Method (Demuth & Beale, 2005). This technique avoids a problem called overfitting that occurs during the neural network training. The network seems to be very well trained by showing very small errors from the training data set, but when new inputs are used the error is large.

APPLICATION AND RESULTS

Recurrent neural network model

The RNN model related the current-period forecasted rainfall and the previous reservoir inflow with the current inflow. The historical data utilized in the procedure contain 20 years of monthly inflows. The RNN was calibrated using the monthly inflows of the first 12 years and validation was carried out over the last eight years. Figure 3 shows the scatter graph between historical and RNN-forecast inflows for the last eight years of the data set. The relationships between the inflows from historical data and RNN are displayed in Fig. 4.

The correlation between historical and RNN-forecasted inflows was 96%. Observing Figs 3 and 4, the excellent accuracy obtained by the RNN suggests that it is very effective for forecasting of reservoir inflows.

Implicit stochastic optimization procedure and recurrent neural networks

The ISO procedure was applied to the Ishitegawa Dam reservoir which supplies the city of Matsuyama, located in Ehime Prefecture, Japan. The reservoir is also used for irrigation and flood control. The maximum reservoir storage ($S_{\max}$) was assumed to be only 8 500 000 m^3, different from the actual capacity of 12 800 000 m^3, because it was desired to observe many shortage situations and then compare how they are handled by the models.

The ISO process was run under an operating horizon of 288 months (24 years). 100 sequences of synthetic monthly inflow data were generated by the autoregressive model of Thomas-Fiering (Celeste *et al.*, 2004), which successfully incorporated the statistical features of the historical data into the generated values, as seen in Table 1. The initial storage was set to $S_{\max}$. The first and last two years of data were rejected to avoid problems with boundary conditions. That means the inflow series became the same as the one used for calibrating and validating the RNN. This provided 24 000 optimal monthly releases.

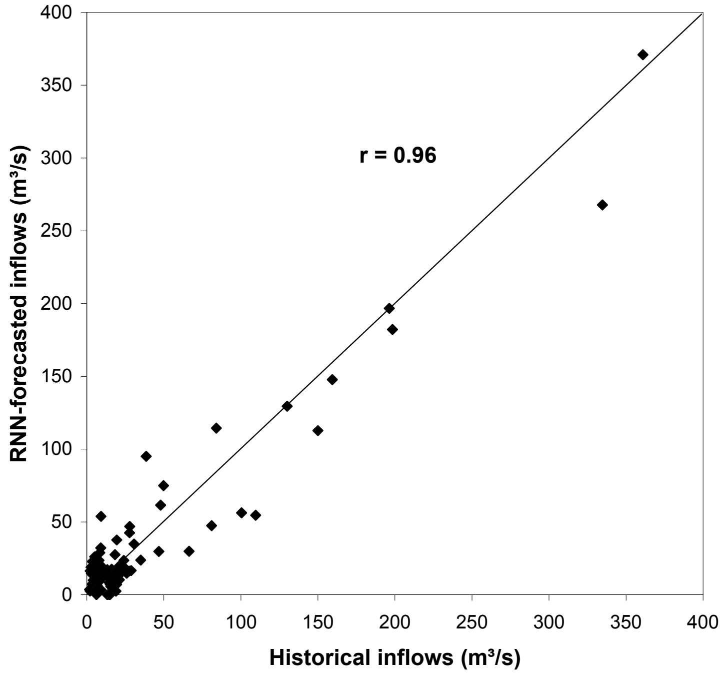

Fig. 3 Scatter graph of historical inflows and RNN-forecasted inflows for the last eight years of the data set.

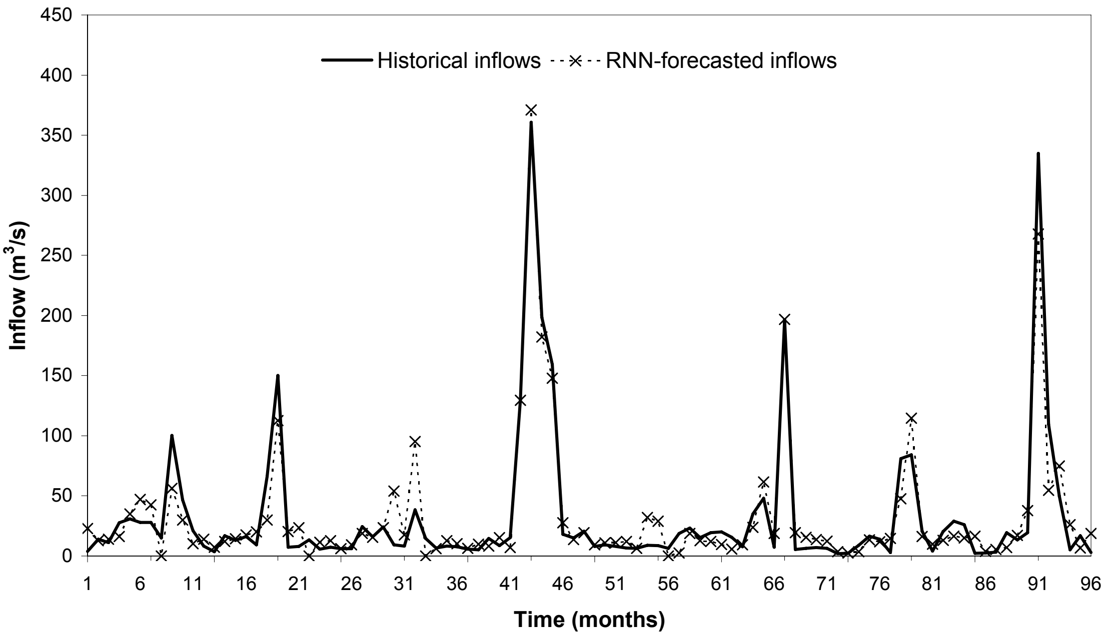

Fig. 4 Comparison between historical inflows and RNN-forecasted inflows for the last eight years of the data set.

The data of releases, initial storages and inflows for the months of January through December were grouped and plotted. For each month, a mesh of 100 × 100 nodes between the minimum and maximum values of storage and inflow was established and the corresponding values of releases were computed by numerical interpolation. This

Table 1 Monthly and annual statistics for the historical and synthetic monthly inflow series.

	Historical series: Mean	Standard deviation	Skewness coefficient	Synthetic series (100 independent processes): Mean	Standard deviation	Skewness coefficient
Jan	6.2	5.7	1.7	6.3	5.7	1.6
Feb	8.4	7.3	0.8	8.4	6.9	0.9
Mar	10.2	6.8	1.4	10.2	7.0	1.3
Apr	12.9	8.7	1.1	13.0	8.8	0.9
May	16.9	16.1	1.6	16.5	14.9	1.3
Jun	58.2	50.2	0.8	60.2	48.3	1.0
Jul	102.1	109.8	1.4	107.7	103.9	1.3
Aug	35.5	59.7	2.4	41.4	56.3	2.5
Sep	39.9	48.4	1.5	43.9	45.9	1.6
Oct	20.5	25.0	3.0	20.8	25.4	2.5
Nov	11.9	11.7	1.8	11.7	11.0	1.6
Dec	9.7	10.5	1.8	10.0	10.3	1.8
Annual	332.5	222.6	1.8	350.1	211.2	2.0

process generated 12 surfaces, one for each month. Figures 5–7 show examples of surfaces generated for January, April and November, respectively. For each surface two boundary conditions were added: (1) minimum storage and minimum inflow implies no release; and (2) maximum storage and maximum inflow implies maximum release. These graphs reveal the highly nonlinear correlations of the variables.

A simulation of the reservoir operation for the last eight years of the data set (data used for the test of the RNN model) was carried out by using the RNN-forecasted inflow as input to the release rules defined by the ISO procedure. Results obtained from the utilization of the deterministic optimization model, assuming the whole horizon of inflows as perfect forecasts, were used for comparison. The operation of the system using the perfect-forecast situation gives us the "ideal" releases that should be

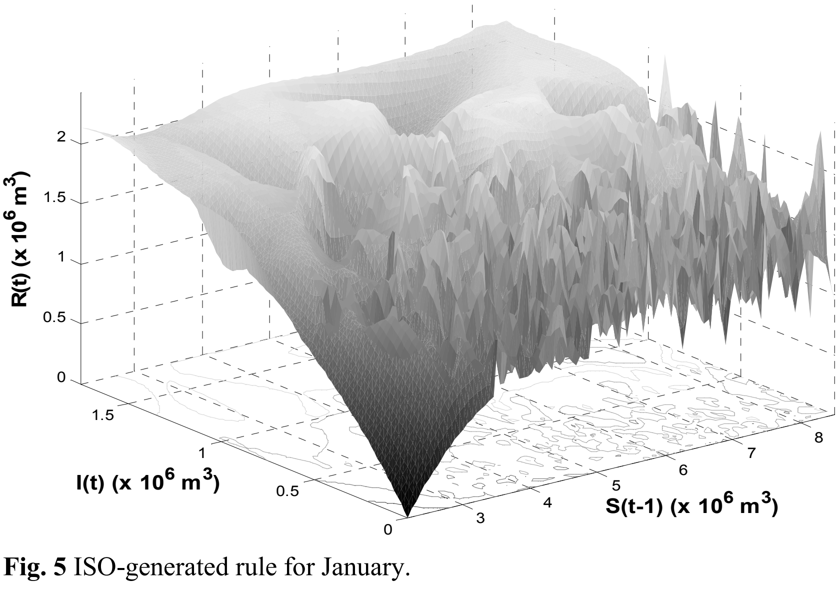

Fig. 5 ISO-generated rule for January.

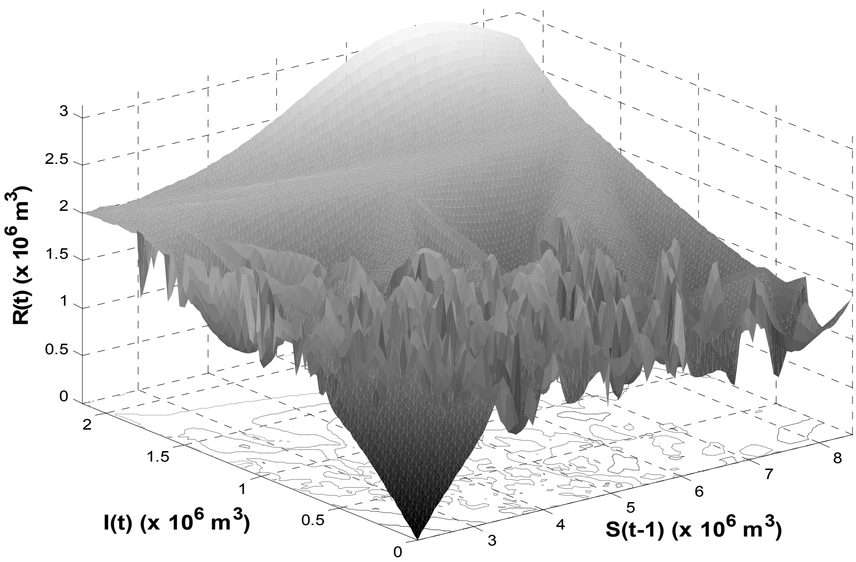

Fig. 6 ISO-generated rule for April.

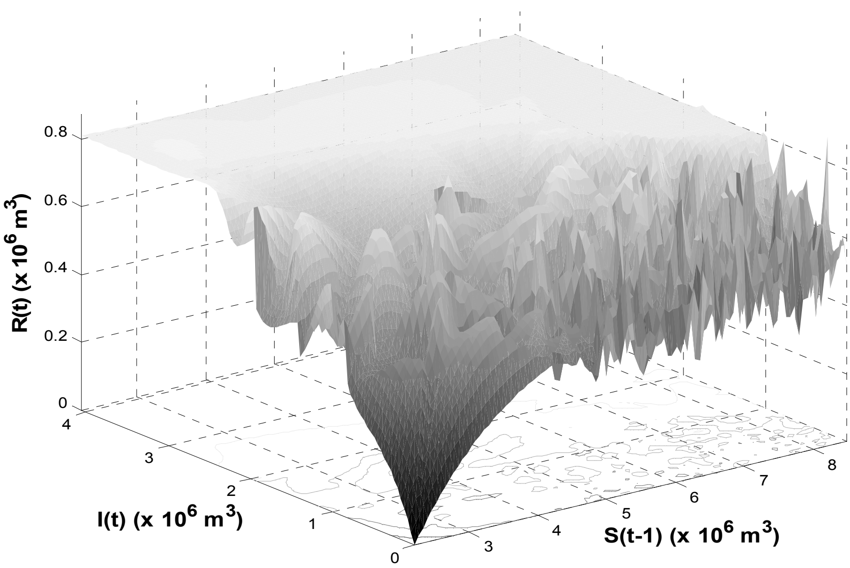

Fig. 7 ISO-generated rule for November.

employed for all eight years since it has knowledge of all future inflow values. In addition, simulations based on the ISO-generated policies using the perfect forecast (assuming the current inflow input needed by the rules to be the perfect forecast) and on the so-called Standard Linear Operating Policy (Loucks *et al.*, 1981), or SLOP, were used for comparison. The SLOP states that when the available water is equal to or less than the demands, all storage water is released; and when the available water exceeds the demands, the excess is stored in the reservoir until its maximum capacity is reached and spillage starts to occur.

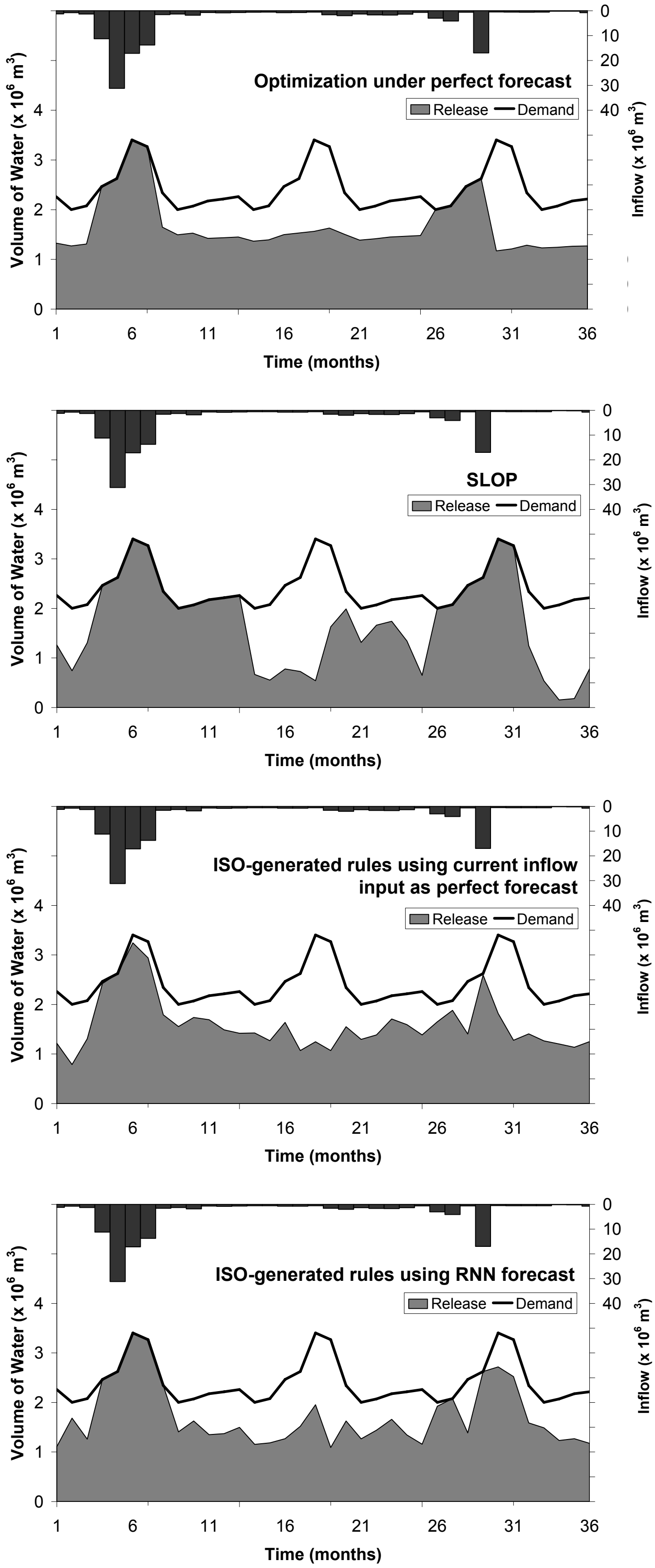

Fig. 8 Results for the period between the fourth and sixth year within the 8-year series.

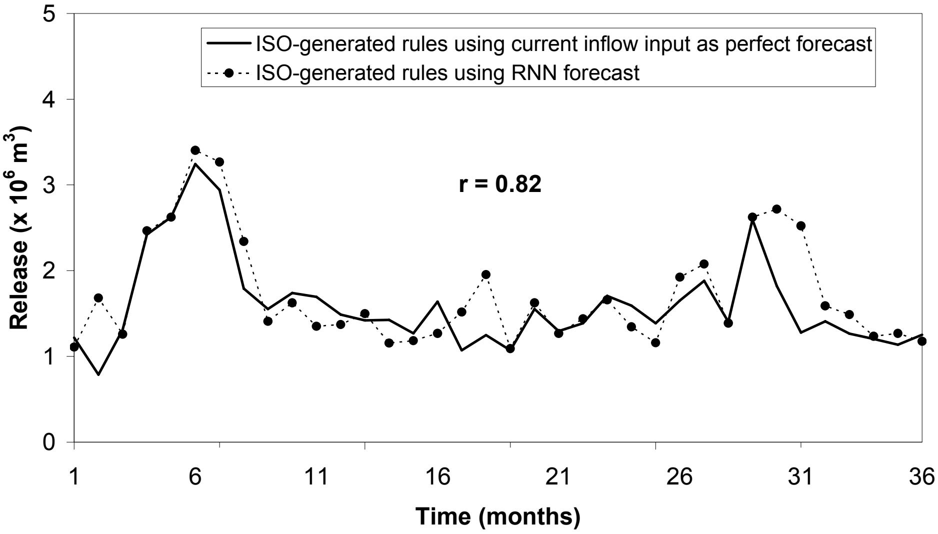

Fig. 9 Correlations of releases obtained by ISO-generated rules using current inflow input as perfect forecast and ISO-generated rules using RNN-based forecast.

Figures 8 and 9 show the results for the period between the fourth and sixth years within the 8-year series. The correlation regarding water allocation between the results obtained by the ISO-generated rules using the RNN-based forecast and optimization under a perfect forecast was 73%. The correlations of ISO-generated rules using the perfect forecast and SLOP with optimization under perfect forecast were 87% and 59%, respectively. The relationship between releases obtained by ISO-generated rules using the RNN-based forecast and ISO-generated rules using the perfect forecast was 82%.

Comparing the results from the optimization under perfect forecast with those from the SLOP, it can be seen that the optimization model tries to mitigate the great concentrated deficits that happen with the SLOP by decreasing the releases prior to shortages periods so that the overall deficit also diminishes. Examination of Fig. 8 shows that the simulation with the ISO-generated rules using the perfect forecast tries to allocate water in a way very similar to the optimization under the perfect forecast. This information indicates that the results from the derived release policies are quite satisfactory because they have information only on the previous reservoir storage and current inflow, whereas the optimization model has knowledge of inflows for the whole operating horizon and thus better means to define superior policies.

Analysis of Figs 8 and 9 shows that the releases generated by the ISO-generated rules using the RNN-based forecast were similar to those obtained by ISO-generated rules using the perfect forecast and better than those obtained by the standard rules of simulation (SLOP).

The RNN model utilized in this study was capable of learning and generating interesting temporal representations. Consequently, this model may be appropriate for providing consistent monthly inflow forecasts to be used by the interpolation-based ISO procedure.

CONCLUSION

In this paper, RNN-based forecasts were used to assist reservoir operations carried out by ISO-derived operating policies. The methodology was applied to define monthly optimal releases to the reservoir that supplies the city of Matsuyama in Japan.

The basic principle of the RNN model was to relate current-period forecasted rainfall and previous reservoir inflow in order to predict the current inflow. The excellent accuracy obtained by the RNN indicated that it is very competent for one-month-ahead forecasting of reservoir inflows.

Once the efficiency of the RNN model was verified, a simulation of the reservoir operation could be done by using its forecasts as input to the operating rules defined by ISO. The optimal reservoir releases obtained by the ISO using the RNN-based forecasts were shown to be highly correlated with those using perfect forecasts and superior to the ones obtained by the standard rules of operation (SLOP). Thus, this suggests that temporal neural networks such as RNNs may provide reliable supports for sustainable water resources management.

REFERENCES

Celeste, A. B., Suzuki, K., Kadota, A. & Farias, C. A. S. (2004) Stochastic generation of inflow scenarios to be used by optimal reservoir operation models. *Annual J. Hydraulic Engng JSCE* **48**(1), 451–456.

Celeste, A. B., Suzuki, K., Kadota, A. & Santos, V. S. (2005) Derivation of reservoir operating rules by implicit stochastic optimization. *Annual J. Hydraulic Engng JSCE* **49**(2), 1111–1116.

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

Demuth, H. & Beale, M. (2005) *Neural Network Toolbox: For Use with Matlab*. The MathWorks, Inc., Natick, Massachusetts, USA.

Elman, J. L. (1990) Finding structure in time. *Cognitive Science* **14**, 179–211.

Farias, C. A. S., Celeste, A. B., Sakata, Y., Kadota, A. & Suzuki, K. (2006) Use of Monte Carlo optimization and artificial neural networks for deriving reservoir operating rules. *Annual J. Hydraulic Engng JSCE* **50**(1), 25–30.

Farias, C. A. S., Celeste, A. B., Kadota, A. & Suzuki, K. (2006) ANN against interpolation-based implicit stochastic optimization for deriving reservoir operating rules. In: *Proceedings of the 15th Congress of APD-IAHR* (ed. by V. Sundar *et al.*), 633–639. Allied Publishers Pvt. Ltd, ITT Madras, Chennai, India.

Haykin, S. (1999) *Neural Networks: A Comprehensive Foundation*. Prentice-Hall, Upper Saddle River, New Jersey, USA.

Loucks, D. P., Stedinger, J. R. & Haith, D. A. (1981) Water resources systems planning and analysis. Prentice Hall, Englewood Cliffs, New Jersey, USA.

Moller, M. F. (1993) Scaled conjugate gradient algorithm for fast supervised learning. *Neural Networks* **6**(4), 525–533.

The MathWorks, Inc. (2000) *Optimization Toolbox User's Guide.* The MathWorks, Inc., Natick, Massachusetts, USA.