An integrated neural network stochastic dynamic programming model for optimizing the operation policy of Aswan High Dam

A. H. El-Shafie and M. S. El-Manadely

ABSTRACT

Developing optimal release policies of multipurpose reservoirs is very complex, especially for reservoirs within a stochastic environment. Existing techniques are limited in their ability to represent risks associated with deciding a release policy. The risk aspect of the decisions affects the design and operation of reservoirs. A decision-making model is presented that is capable of replicating the manner in which risks associated with reservoir release decisions are perceived, interpreted and compared by a decision-maker. The model is based on Neural Network (NN) theory. This decision-making model can be used with a Stochastic Dynamic Programming (SDP) approach to produce a NN-SDP model. The resulting integrated model allows the attitudes towards risk of a decision-maker to be considered explicitly in defining the optimal release policy. Clear differences in the policies generated from the basic SDP and the NN-SDP models are observed when examining the operation of Aswan High Dam (AHD). The NN-SDP model yields policies that are more reliable and resilient and less vulnerable than those obtained using the SDP model.

Key words | Aswan High Dam, dam operation, neural network, risk analysis, stochastic dynamic program

A. H. El-Shafie (corresponding author)
Civil and Structural Engineering Department,
University Kebangsaan Malaysia,
43600 UKM Bangi,
Selangor Darul Ehsan,
Malaysia
E-mail: *elshafie@eng.ukm.my*

M. S. El-Manadely
Hydraulic and Irrigation Department,
Faculty of Engineering,
Cairo University,
P.O. Box 12613,
Giza,
Eygpt

INTRODUCTION

Background

The development of management models for identification of optimal operating policies for reservoirs spans more than four decades of research. In an uncertain environment, where climatic factors such as streamflow are stochastic, the returns from reservoir releases defined by the optimal policy are uncertain. Furthermore, the consequences of release decisions cannot be fully realized until future unknown events occur. Existing optimization models have tended to address these uncertainties through identification of release policies that optimize a summative metric of the risk associated with the release decisions, rather than explicitly recognizing the full spectrum of consequences possible within an uncertain environment. Such a reduction in model objective provides a rational definition of optimal release policies.

The use of more complete representation of the risks associated with the decision in a release policy has been limited by existing optimization frameworks which require summative metrics of performance as their objective functions. A model capable of replicating the manner in which the complete spectrum of risks associated with reservoir release decisions are perceived, interpreted and compared by a decision-maker is required. The model is based upon Neural Network theory, and enables a more complete representation of the risks of a particular decision to be considered in making decisions on reservoir releases. The risk selection process, which is in general internal to the decision-maker, is

doi: 10.2166/nh.2010.043

duplicated within the neural network thereby allowing the concerns of the users to influence the release policy. The neural network proposed to perform the decision-making process is a feed-forward back-propagation network (Chang & Chang 2001; Chang *et al.* 2005).

The neural network decision-making model can be used in a stochastic dynamic programming approach to give a Neural Network–Stochastic Dynamic Programming (NN-SDP) model. The SDP framework, already used extensively in the literature (e.g. Bellman 1959; Butcher 1971; Goulter & Tai 1985; Chandramouli & Raman 2001; Chaves *et al.* 2003; Chaves & Kojiri 2004) to identify optimal reservoir release policies within a stochastic environment, has the flexibility necessary to enable the replacement of the expectation function by the proposed neural network. The resulting integrated model allows the attitudes towards risk of a single decision-maker to be considered explicitly in defining the optimal release policy for a multipurpose reservoir.

Risk is defined as the possibility of a gain or loss that affects the group or individual making the decision. Risk is represented mathematically by the cumulative distribution of consequences, and is comprised of the probability and magnitude of the gain or loss (Xu *et al.* 1998; Kaplan 2000; Kaplan *et al.* 2001; Haimes *et al.* 2002).

Garrick & Kaplan (1981) proposed a quantitative definition of risk based on the concept that risk must be represented by the complete distribution of the possible outcomes. This can be described in other words by the complete risk curve, defined as the cumulative exceedance probability distribution of losses.

Problem statement

Under this interpretation of risk, the only way to properly communicate all the subtleties of the risk associated with a particular decision is to present the complete risk curve. The selection of the best strategy is then accomplished by comparing the complete risk curve of the possible alternatives. The alternative to the preferred risk curve is the optimal solution.

The shortcoming of the approach relates to the difficulties of incorporating the complete risk curve within an analytical framework. Under current technologies, risk curve comparison for decision-making must be accomplished visually, normally by presenting the curves to the decision-maker for selection. Visual comparison and evaluation of alternative risks are extremely impractical when large numbers of alternatives must be examined. Furthermore, the visual assessment by the decision-maker may not be consistent over time. These problems suggest the need for a mathematical framework capable of making the necessary solution.

The NN-SDP model, with its emphasis on the attitudes towards risk of decision-makers, complements current management models. If the value of management models is to inform decision-makers during the planning for the operation of a reservoir, the proposed methodology allows the views and concerns of individual water users to be considered in the generation of the release policy of the reservoir.

The optimization model proposed in this study employs mathematical neural network architecture to present the quantitative definition of risk. The framework allows for the pair-wise comparison of complete risk curves, without the need for surrogate measures of either risk or risk attitudes. The key to the proposed approach is that the risk curves, when presented as visual patterns, can be analyzed by the powerful class of pattern recognition techniques known as neural networks. A neural network can be developed to compare two risk curves, the comparison being based on the previously articulated risk preference of the decision-maker. The risk curve neural network mimics the selection process used by the decision-maker. This mimicking is accomplished by observing the responses of the decision-maker to a series of pair-wise comparisons, and training the neural network to respond in an identical manner. The methodology is demonstrated by application to the AHD and its reservoir (Lake Nasser) in Egypt.

Objective

The primary objective of the present study is to use a more comprehensive representation of the risks associated in determining optimal release policy from the point of view of the decision-maker. The risk selection process, which is in general internal to the decision-maker, allows the concerns of the users to be considered and influence the release policy.

In addition, the objective of the present study is to develop an improved technique whereby the attitudes towards risks probabilities and magnitudes of consequences of a decision-maker could be comprehensively incorporated.

The NN-SDP formulation is proposed to address this problem. The ability of the model to identify the optimal release policy for a reservoir, as defined by the attitudes of the decision-maker and learnt by the neural network, is tested.

METHODOLOGY

Neural network

Artificial Neural Networks (ANNs) are densely interconnected processing units that utilize parallel computation algorithms. The basic advantage of ANNs is that they can learn from representative examples without providing special programming modules to simulate special patterns in the data set (Gibson & Cowan 1990). This allows ANNs to learn and adapt to a continuously changing environment. Therefore, ANNs can be trained to perform a particular function by tuning the values of the weights (connections) between these elements.

Multilayer ANN has been reported as a powerful modelling tool (Gibson & Cowan 1990; Ooyen & Nienhuis 1992). The input and output layers of any network have numbers of neurons equal to the number of the inputs and outputs of the system, respectively. The architecture of a multilayer feed-forward neural network can have many layers between the input and the output layers, where a layer represents a set of parallel processing units (or nodes). These intermediate layers are known as hidden layers. The main function of the hidden layer is to allow the network to detect and capture the relevant patterns in the data and to perform complex non-linear mapping between the input and the output variables. The sole role of the input layer of nodes is to relate the external inputs to the neurons of the hidden layer. The number of input nodes therefore corresponds to the number of input variables. The outputs of the hidden layer are passed to the last (or output) layer, which provides the final output of the network. Finding a parsimonious model for accurate prediction is critical since there is no formal method for determining the appropriate number of hidden nodes prior to training. We therefore resort to a trial-and-error method commonly used for network design (Bishop 1996; Haykin 1999).

In the prediction context, multilayer feed-forward neural network training consists of providing input-output examples to the network and minimizing the objective function (i.e. error function) using either a first-order or a second-order optimization method. Training can be formulated as one of minimizing a function of the weight, the sum of the non-linear least squares between the observed and the predicted outputs defined by:

$$E = \frac{1}{2}\sum_{P=1}^{n}(Y_O - Y_P)^2 \tag{1}$$

where n is the number of patterns (observations) and Y_O represents the observed response (target output) and Y_P the model response (predicted output). In the back-propagation training, minimization of the error function (Equation (1)) is attempted using the steepest descent method and computing the gradient of the error function by applying the chain rule on the hidden layers of the feed-forward neural network (Bishop 1996; Haykin 1999).

Consider a typical multilayer feed-forward neural network whose hidden layer contains M neurons. The network is based on the equations:

$$net_{PJ} = \sum_{I=1}^{N} W_{ji}x_{pi} + W_{jo} \tag{2}$$

$$g(net_{PJ}) = \frac{1}{1 + e^{-net_{PJ}}} \tag{3}$$

where net_{PJ} is the weighted inputs to the jth hidden unit, N is the total number of input nodes, W_{ji} is the weight from input unit i to the hidden unit j, x_{pi} is a value of the ith input for pattern P, W_{jo} is the threshold (or bias) for neuron j, and $g(net_{PJ})$ is the jth neuron's activation function assuming that g is a logistic function. Note that the input units do not perform an operation on the information, but simply pass it onto the hidden nodes. The output unit receives a net input of:

$$net_{pk} = \sum_{J=1}^{M} W_{kj}.g(net_{PJ}) + W_{ko} \tag{4a}$$

$$y_{pk} = g(net_{pk}) \tag{4b}$$

where M is the number of hidden units, W_{kj} represents the weight connecting the hidden node j to the output k, W_{ko} is the threshold value for neuron k and y_{pk} is the kth predicted output. Recalling that the ultimate goal of the network training is to find the set of weightsW_{ji} (connecting input units i to

the hidden units j) and W_{kj} (connecting the hidden units j to output k) that minimize the objective function (Equation (1)). Since Equation (1) is not an explicit function of the weight in the hidden layer, the first partial derivatives of E in Equation (1) are evaluated with respect to the weights using the chain rule, and the weights are moved in the steepest-descent direction. This can be formulated mathematically as:

$$\Delta W_{kj} = -\eta \frac{\partial E}{\partial W_{kj}} \tag{5}$$

where η is the learning rate, which scales the step size. The usual approach in back-propagation training consists of choosing η according to the relation $0<\eta<1$ (Ooyen & Nienhuis 1992).

Stochastic dynamic programming

Dynamic Programming (DP) is an optimization framework that has been used extensively for identifying optimal release policies for reservoirs (Yeh 1985). The basic DP technique, which was initially developed by Bellman (1959), is an optimization procedure that decomposes complex problems which require large number of decisions into a series of independent but interrelated subsystems. The subsystems, or stages resulting from this decomposition, are solved sequentially to obtain the optimal solution to the complete model (Howard 1960; Kelman & Stendinger 1990; Labadie 2004).

The objective function of the SDP model, i.e. minimization deviations from demand targets, is represented by the following recursive equation:

$$f_N(s,i) = \underset{k}{Min}\left(\delta_N^k + \sum_{j=1}^{j} P_N^{ij} f_{N-1}(k,j)\right), \tag{6}$$

where $s=1, 2, \ldots, s$; $i=1, 2, \ldots, i$; $f_N(s,i)$ is the optimal objective function value when there is a storage level s at the beginning of time interval t (stage n) and a reservoir inflow i during time interval t (stage N); N is stage index ($N=1$ corresponds to the last time interval in the time horizon of the model and $N=N$ corresponds to the first time interval); δ_N^k is the deviation from target demand resulting from policy decision k during time interval t (stage N); P_N^{ij} is transition probability of observing an inflow j during time interval $t+1$ (stage N–1) if an inflow of i has been observed during time interval t (stage N); K is the release decision resulting in a storage level k at the end of time interval t (stage N) or the beginning of time interval $t+1$ (stage N–1); S is the number of discrete storage levels at the beginning of time interval t (stage N); i is the number of discrete inflows during time interval t (stage N); and j is the number of discrete inflows during time interval $t+1$ (stage N–1).

It should be noted that the only deviations from target demands δ of concern are the supply deficits. The administrative structure currently in place for the AHD on which the proposed model is demonstrated allows for increases in water allocations if additional water supplies become available. Surplus releases, which simply indicate a spillage of water downstream, are therefore not penalized in terms of the objective of this model. However, in situations where surplus releases are to be minimized (e.g. Simonovic & Burn 1989), the proposed methodology can be employed simply by redefining the deviations from target demands such that surplus releases are penalized.

The deviation from the target demand and its penalty over a time interval can be expressed mathematically in two steps:

$$\delta_N^k = Max(0, (d_N - r_N^k)) \tag{7a}$$

$$\text{Penalized deviation} = (\delta_N^k)^2 \tag{7b}$$

where d_N is the target demand during time interval t (stage N) and $r_N{}^k$ is the reservoir release associated with policy decision k during time interval t (stage N). The water released to meet the demand associated with policy decision k is calculated from:

$$r_N^k = \phi_N^s + \psi_N^i - \phi_{N-1}^k - e_N \tag{8}$$

where ϕ_N^s is storage level s at the beginning of time interval t (stage N); ψ_N^i is reservoir inflow i during time interval t (stage N); and e_N is water losses during time interval t (stage N) as a function of the surface area and water level of the reservoir.

Equations (6)–(8) define the basic SDP formulation for minimizing expected deviation from demand targets, subject to the operational constraints of the system.

SDP model with risk curves

In the model proposed in this study, a neural network is incorporated into the basic SDP formulation defined by

Equations (6)–(8). In particular, the recursive equation relationship that derives the optimization search is modified and extended from that in Equation (6), such that the evaluation of the alternate release decisions is accomplished by the neural network instead of the expectation function. This change removes the need for Equation (7) since the risk curves reflect the risk attitudes of the decision-maker. This eliminates the need for a surrogate measure of attitudes towards risk such as that encapsulated within the weighted squared deviations from target demands.

In the modified structure of formulation, rather than calculating the expected value of the objective function for each possible alternative decision and using these values to identify the optimum, the methodology compares the risks associated with alternate release policy. The risk curves upon which the risks are based and subsequently compared are the input for the neural network, which in turn identifies the preferred risk alternative. This modification in the SDP framework does not violate Bellman's underlying principle of optimality, which is critical to the use of both DP and SDP. The release decisions at each stage in the time horizon remain independent of the state of the system in subsequent stages.

As described above, the process of identification of the preferred risk curve is achieved by pair-wise comparison of risk curves, resulting in the following new recursive function for the NN-SDP formulation:

$$f_n(s,i) = \underset{k}{NN}(g_n(s,i,k)), \tag{9}$$

where $i = 1, 2, \ldots, i$; $s = 1, 2, \ldots,$ s; $f_n(s,i)$is the optimal objective function value when there is a storage level s at the beginning of time interval t (stage n) and a reservoir inflow i during time interval t (stage n); $\underset{k}{NN}$ is the pair-wise comparison operator that consists of comparing k risk curves to identify the policy decision with the preferred risk; and $g_n(s,i,k)$ is the risk curve associated with a storage level s at the beginning of time interval t (stage n), an inflow i during time interval t (stage n) and a final storage level k at the end of time interval t (stage n). Equation (9) replaces the original recursive Equation (6) in the SDP model.

Recall that the pair-wise comparisons, which are the basis of Equation (9), are accomplished using the risk curve neural network using two risk curves for each comparison. Conceptually, the comparisons may be performed on all possible combinations of two risk curves from the population of K curves (corresponding to the K possible decisions). The actual implementation of the methodology in this study assumes that the transposition rule applies for curve preference; that is, if risk curve A is better than risk curve B and curve B is better than curve C, then curve A is better than curve C. Therefore, any curve identified as undesirable in relation to another curve during the comparison process for determination of the optimal decision for a particular storage and inflow combination at a given stage is eliminated from any further comparisons. This process is shown in Figure 1, where undesirable risk curves are progressively eliminated from the comparison process.

It should be noted that the terms $f_n(s,i)$and $g_n(s,i,k)$ in Equation (9) denote complete risk curves specified in terms of sets of ordered pairs, each of which define a loss R_α and its associated cumulative probability P_α, i.e.

$$g_n(s,i,k) = \{(R_1,P_1),(R_2,P_2),\ldots,(R_\alpha,P_\alpha),\ldots,(R_G,P_G)\}, \tag{10}$$

$$f_n(s,i) = \{(R_1,P_1),(R_2,P_2),\ldots,(R_\alpha,P_\alpha),\ldots,(R_G,P_G)\}, \tag{11}$$

$$R_1 \le R_2 \le \ldots \le R_\alpha \le \ldots. \le R_G \tag{12}$$

$$P_1 \le P_2 \le \ldots \le P_\alpha \le \ldots. \le P_G \tag{13}$$

where R_α is the possible outcome α, defined by the deviations from target demands; P_α is the cumulative probability for outcome R_α; and G is the number of ordered pairs used to define the risk curve. These ordered pairs represent discrete points along the cumulative exceedance probability distribution of water deficits.

The ordered pairs that comprise the risk curve g_n are calculated using:

$$R_\alpha(g_n(s,i,k)) = \delta_n^k + R_\alpha(f_{n-1}(k,j)), \tag{14}$$

where $\alpha = 1, 2, \ldots, \Gamma\ (= GJ) \forall j, \beta,$

$$P_\alpha(g_n(s,i,k)) = \left\{P_n^{ij} * \left[P_\beta(f_{n-1}(k,j)) - P_{\beta+1}(f_{n-1}(k,j))\right] + P_{\alpha+1}(g_n(s,i,k))\right\} \tag{15}$$

where $\alpha = \Gamma, \Gamma-1, \ldots, 1\ \forall\ i, \beta$ and

$$P_{\Gamma+1}\ g_n(s,i,k) = 0.0. \tag{16}$$

The notation used in Equations (14)–(16) has been slightly altered from the notation used in Equations (9)–(13)

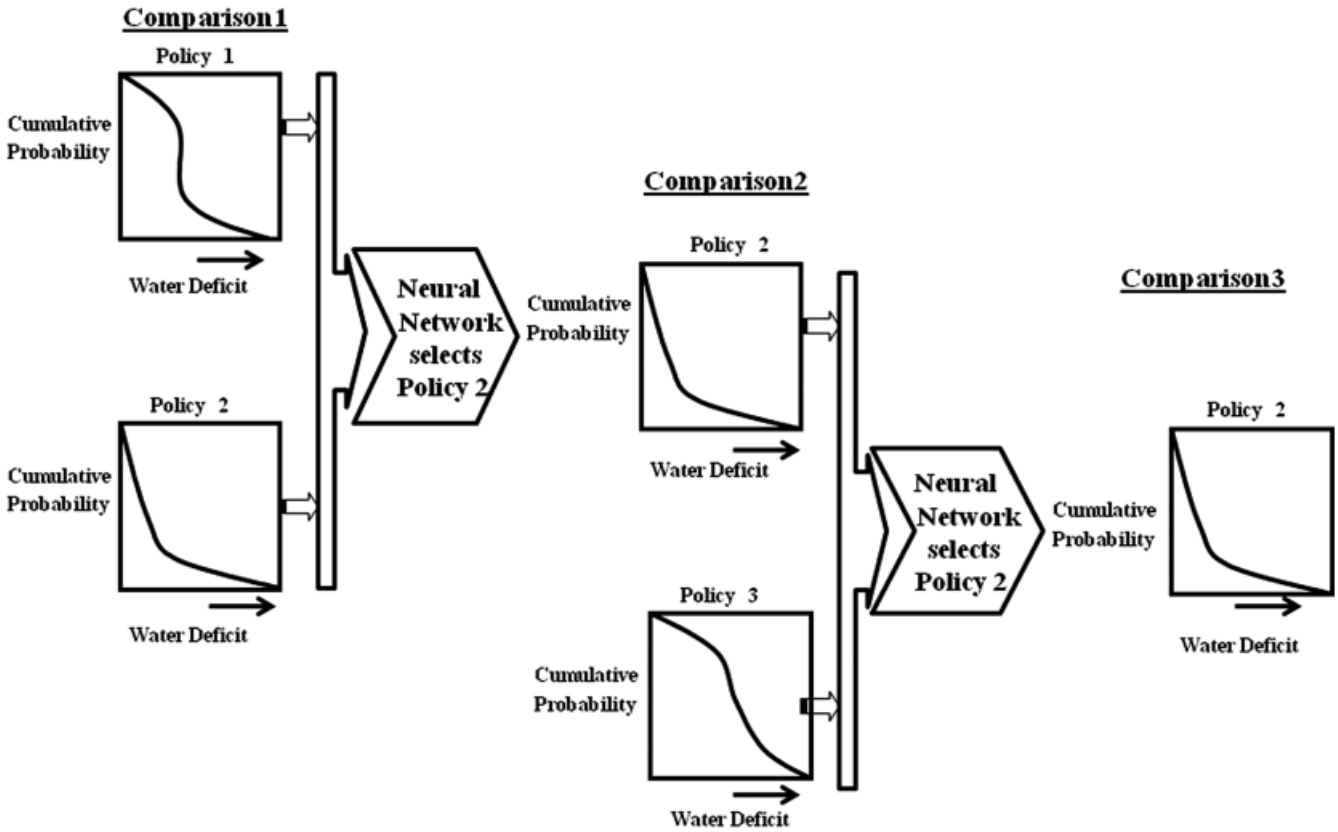

Figure 1 | Pair-wise comparison process.

to differentiate between ordered pairs taken from different risk curves. The process defined by Equations (9)–(11), and depicted in Figures 2 and 3, combines the immediate risks of release decisions with the possible risks in future stages. This identifies the cumulative consequences of a decision as the consequences in the current stage combined with the 'optimal' consequences in subsequent stage. The risk curve associated with a particular release decision therefore contains all risk outcomes, both in the immediate future and the long term.

This process of identifying the distribution of cumulative outcomes is recursive, starting with the largest of the cumulative outcomes R_α defined by Equation (9). This largest cumulative outcome is therefore the combination of immediate deficit δ_n^k and the long-term cumulative deficit R_β that yields the greatest loss. The recursive process defined by Equation (10) then identifies the cumulative probability of progressively smaller cumulative deficits R_α, terminating when the cumulative probability for each combination of short-term and long-term deficits has been evaluated. It should be noted that Bellman's principle of optimality is not violated by this recursive process as the release decision at stage n remains independent of the decisions made in future time periods.

It must be stressed that the resulting risk curve $g_n(s,i,k)$ represents the distribution of the cumulative deficits associated with the decisions made in the n stages (time periods t to the end of the time horizon T). The deficits experienced during individual time periods (or stages) do not remain distinct within the proposed risk representation. Distinct cumulative deficits are however represented, and indicate the total water deficit experienced over the n stages.

The new risk curve, $g_n(s,i,k)$, contains Γ ordered pairs (which are actually the number of inflow categories J multiplied by the number of ordered pairs that define the risk curve $f_n(s,i)$, i.e. G). Since this new risk curve is used to define risks in future stages of the model, it is necessary to implement a strategy to arrest the growth of ordered pairs. Consider that

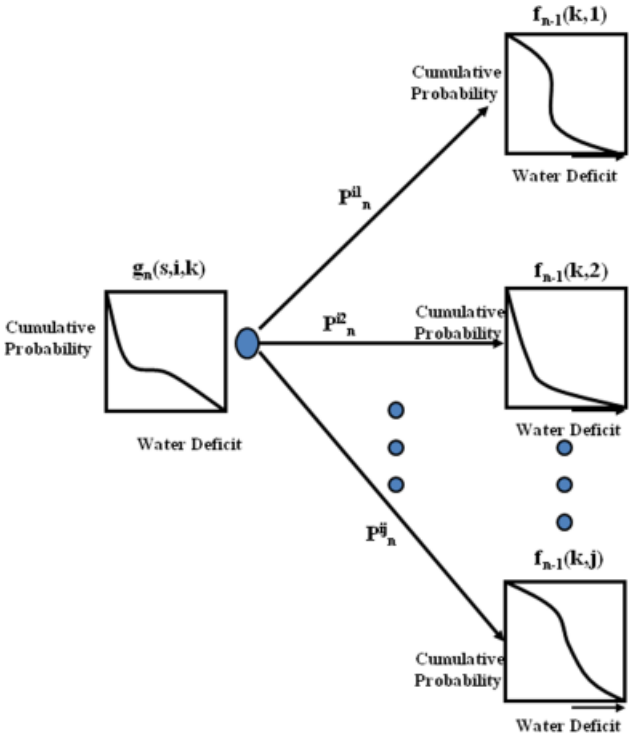

Figure 2 | Convolution of risk curve.

the risk curve $g_n(s,i,k)$ is based on the immediate risk associated with the decision k and the long-term risks represented by the set of f_{n-1} curves (each containing G ordered pairs). Recursive Equation (6) then selects the $f_n(s,i)$ risk curve from the set of possible g_n curves (with each curve representing a different release decision k). Since the g_n curves are defined by Γ ordered pairs, Γ (rather than G) ordered pairs define the $f_n(s,i)$risk curve.

Subsequent evaluations of risk, including the determination of curve $g_{n+1}(s,i,k)$, would require an ever-increasing

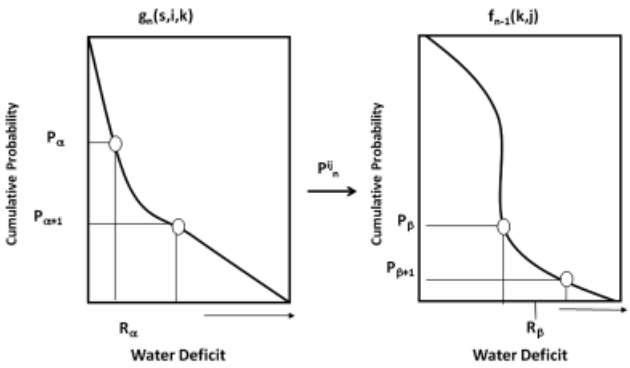

Figure 3 | Single calculation in the convolution of risk curves.

number of ordered pairs. In the case of risk curve $g_{n+1}(s,i,k)$, the number of ordered pairs would be Γ^*G rather than J^*G ordered pairs obtained for the risk curve $g_n(s,i,k)$. If this growth in ordered pairs is not addressed, the curves needed to define the risk in subsequent stages may grow to unmanageable proportions as the recursive process progresses.

One approach which can be used to arrest the growth in ordered pairs is to redefine the risk curve $g_n(s,i,k)$ such that the number of ordered pairs used to define the curve is the same as for f_{n-1}. Note that this redefinition must be performed in a way that preserves the shape of the curve. These two requirements can be met by sampling the risk curve $g_n(s,i,k)$at G pre-determined equidistant points along the deficit axis. The sampling process, which was selected for use in this study, is described in the following section.

The NN-SDP model formulation proposed in this paper is now completely described by Equations (9)–(16) and Equations (7) and (8). The model identifies the optimal operating policy based on the complete risk curve preferences of the decision-maker rather than on the basis of simplistic surrogates or representations of these preferences. The following section describes the neural network model actually employed to implement Equation (9) in the recursive process.

Neural network within stochastic dynamic program

The architecture of the neural network model chosen for the pair-wise comparisons of risk curves consists of three layers arranged in feed-forward architecture. The input layer contains the descriptions of the two risk curves to be compared, with each curve defined by a discrete step function of the cumulative probability distribution. An input neuron is created for each discrete point of each of the two risk curves. Figure 4 depicts an example input process where the risk curves are defined by four discrete points and the input layer contains a total of 8 neurons (4 for each risk curve). In this example network, the combined stimulus is processed by remaining network layers to provide an optimal response contained in the output layer which consists of two neurons.

Each of these two output neurons is trained to respond to the preferred risk curve. Hence if curve 1 is preferred over curve 2, neuron 1 has a greater output signal than neuron 2, and vice versa. Ideally, the response to a pair-wise comparison consists of an output signal of +1.00 for the neuron

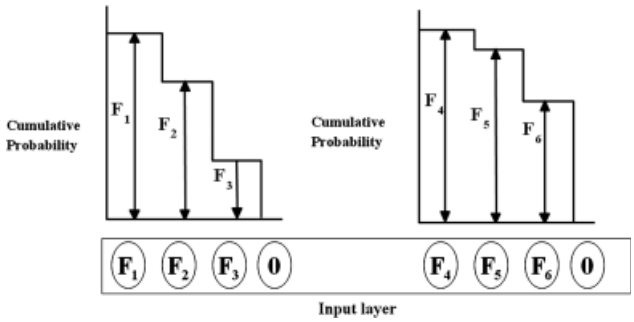

Figure 4 | Example representation of the two risk curves in the input layer of the network.

associated with the preferred curve and an output of 0.00 for the other neuron. Furthermore, curves identified as equivalent by the decision-maker result in signals of +1.00 from both neurons. Unfortunately, the use of continuous values for the output signals, and the small errors between desired and actual responses allowed during training, mean that the output neurons can emit signal deviation in the ideal values of +1.00 and 0.00. A clear differentiation between the two output signals is therefore achieved only as long as (1) the relative difference between the two signals is sufficiently large in the case of one risk curve being preferred over the other, and (2) the relative difference between the two signals is sufficiently small in the case of two curves being equivalent. Hence there exists a threshold in the difference between the two output signals beyond which one risk curve is identified as being preferred. This threshold was set at 20% of the larger signal value, i.e. risk curves are identified as being equivalent if the relative difference in their respective output signals is less than 20% of the larger of the two signal values. If the relative difference in the signal exceeds this threshold, the curve with the greatest output signal is deemed to be preferred and is therefore selected. Preliminary investigation indicated that the overall performance of the neural network is relatively insensitive to the selection of this threshold value.

The exact number of neurons in the hidden layer is obtained through trial and error. Since knowledge is represented in a neural network by both the network architecture and its strength of the neural connections, the final architecture is specific to the set of risk preferences used and is therefore unique to the decision-maker.

The input layer of the network used in this study contains 22 neurons, which corresponds to risk curves defined using 11 discrete points. The decision to use 11 discrete points to represent each risk curve is based on trade-offs between the size and computational burden of a large network and the need to express the preferences of the decision-maker at an appropriately accurate level. In fact, the number of significant features used to define risk preferences controls the number of neurons required in the input layer. Use of a small number of discrete points to define the risk curves means that each set of points has the potential to represent a large number of possibly quite different curves. If it is necessary to differentiate between these two curves, then a larger set of points must be employed to define the curves.

While accuracy of the representation of the risk curves is critical to the validity of the comparisons, there nevertheless exists a practical limit on the number of discrete points necessary to adequately define a complete risk curve. Firstly, the risk curves are monotonically decreasing with all probability values constrained between 0.00 and 1.00. This observation suggests that all curves passing through a given set of ordered pairs are likely to be similar in shape. Secondly, there is a practical limit to the ability of a decision-maker to truly differentiate between risk curves having only relatively small differences in risks. For example, decision-makers involved in this study ranked the two curves as equivalent in terms of the associated risk. Therefore, practical limits exist for both the minimum and maximum number of discrete points necessary to effectively define risk curves. More research is needed to identify the optimal number of points used to define complete risk curves, and to evaluate the sensitivity of the neural network methodology to this selection of discrete points. However, it must be stressed that, in theory, the methodology has the flexibility to use any level of precision in the specification or representation of the risk considered appropriate. For the purpose of demonstrating the NN-SDP approach, a curve representation with 11 discrete points (with each point representing approximately 10% of the range of possible losses) was therefore selected for this study.

The size of the neural network, and the computational burden associated with a large network, must also be considered along with the need for an adequate representation of the risk. Each addition of an ordered pair to the definition of the risk curves, i.e. using one more discrete point to define the curve, requires two additional neurons in the input layer and most probably additional neurons in the hidden layers. While the size of the neural network should not be a critical

factor in determining the number of ordered pairs, it is important to know that there is no benefit in going beyond a practical level of precision at which small variations in risk become inconsequential to the decision-maker.

In order to examine the reservoir performance, three different performance indices are utilized in this research: reliability, resilience and vulnerability. Reliability is the probability that the reservoir can supply the volume of water specified by the demand target in any given year. Resilience is taken to be the conditional probability of experiencing a period of no failure (no water deficit) given that a failure occurred in the previous time period. A second measure of resilience (Moy *et al.* 1986), defined as the maximum number of consecutive periods of failure, is also employed. The use of two resiliency measures allows the concept of resilience, which is the ability of the system to recover from a failure, to be more fully expressed. The conditional probability definition of resilience gauges the ability of the system to recover, while the maximum number of consecutive periods of failure provides information on the length of time to that recovery. The final measure of performance is the vulnerability of the system; vulnerability is the maximum deficit experienced in any one year.

APPLICATION OF METHODOLOGY

The NN-SDP model presented previously was applied to the AHD in Egypt. The aim of this application is to demonstrate the effects of using different risk preferences on the resulting water release policies and to compare these with the original SDP formulation, which employs the expectation function.

Description of the Aswan High Dam reservoir

The AHD, which is the major irrigation structure in Egypt, is located on the Nile River near Aswan city. Lake Nasser is supplied by the River Nile with an average annual inflow of 84×10^9 m^3. The reservoir supplies water for irrigation, municipal and industrial use, energy production and navigation purposes. Water allocations to these groups of users are prioritized, with the highest priority given to irrigation.

The operational policy of the reservoir is based on dividing Lake Nasser storage into six zones, as illustrated in Figure 5. The dead storage zone, which is allocated to receive sediment inflow during floods, has a top elevation of 147 m with total volume of about 31×10^9 m^3. The operation rule for this zone is to release no flow regardless of the downstream requirements. The second zone is the live storage zone, which amounts to 90×10^9 m^3. This zone is divided into two parts. The first part, referred to as the buffer zone, lies between elevation 147 m and elevation 150 m. Within this zone, the dam operators make their releases to meet the downstream requirements. The total annual release should not overshoot Egypt's quota (55.5×10^9 m^3). The remaining storage between elevations 175 m and 183 m is divided between a flood buffer and flood control zone.

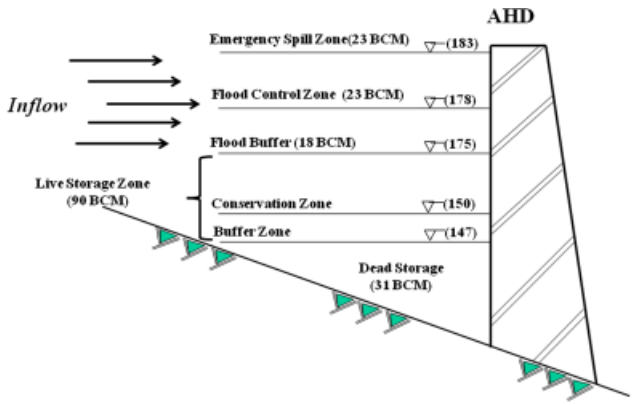

Figure 5 | Different operation zones of Aswan High Dam reservoir.

To reduce the complexity of the water allocation model proposed in the present study, the risks considered shall be limited to the presence of uncertain streamflows. The risk then becomes solely a result of incomplete information regarding future streamflow events. The economic and social variables, as well as all other environmental factors, are assumed to be deterministic or able to be predicted with a perfect forecast. This reduction in scope in no way affects the generality of the methodology proposed. It merely limits the amount of information about the stochastic variables required to measure or reflect the risk.

To study the problem of the optimal operation of AHD reservoir, it is first necessary to develop a well-calibrated model for the reservoir. The operation policies to be sought should then incorporate this model, the probabilistic nature of the Nile River inflows and the over-year storages as well as reservoir spills and losses. Fortunately, previous

investigations have led to the development of a reliable model for the AHD reservoir (Master Water Plan, Ministry of Water Resources and Irrigation, Egypt 1981). However, this model was modified due to the changing of some physical characteristic of the reservoir and additional inflow data to the year 2000. Before applying the proposed optimization model for the AHD reservoir, the data invoked need to be investigated. These may be summarized in the following.

The SDP formulation described above requires discrete representation of storage volume and inflows to the reservoir. Using the available naturalized inflow data at Aswan, five monthly inflow states have been selected as shown in Table 1. The discretization criterion for the inflows is, for both extremely high and low inflow ranges, equal probabilities of 0.05. The three intermediate discretization ranges have equal probabilities of 0.30.

The discrete representation of the state variable, namely inflow to the reservoir, is required. Inflow data are taken as the naturalized monthly streamflow at Aswan in the period 1882–2000. A particular distribution (normal distribution) can be associated with the random variable 'inflow during period t'. It will be discretized in N values in the following way.

1. The domain of the random variable is divided into N intervals in such a form that the area under the PDF (probability density function) for the interval corresponding to the largest values of the inflow is β. The remaining portion of the domain is divided into $N-1$ 'equiprobable' intervals.
2. For each interval, its median is chosen as discrete representative values.

Analysis of the 118 years of streamflow data revealed a strong dependence, as measured by cross-correlation coefficients, between the inflows of consecutive months. The SDP model therefore requires a Lag-one Markov process and the transition probability between any consecutive monthly inflows (at time t and $t+1$). To compute the transition probabilities the procedure described below is followed.

1. Given one discrete value for inflow in period t (say), compute the parameters of the conditional distribution:

$$f_{Qt+1|Qt}(q_t|q_{t+1}) = \frac{f_{Qt+1,Q}(q_{t+1},q_t)}{f_{Qt+1}(q_{t+1})} \tag{17}$$

2. Given the conditional PDF in step 1, the jth row can be computed as

$$P_{j1}^t = \int_{x_1^t}^{\infty} f_{Qt|Qt+1}(q_t|Q_j^{t+1})dq_t \quad (k=1);$$
$$P_{jk}^t = \int_{x_1^t}^{x_{k-1}^t} f_{Qt|Qt+1}(q_t|Q_j^{t+1})dq_t \quad (1<k<N);$$
$$P_{jk}^t = \int_{-\infty}^{x_{N-1}^t} f_{Qt|Qt+1}(q_t|Q_j^{t+1})dq_t \quad (k=N), \tag{18}$$

Table 1 | Five monthly inflow states (10^9 m^3 per month)

	Inflow states				
Month	**Low**	**Medium-low**	**Medium**	**Medium-high**	**High**
January	1.6	2.2	2.8	3.5	4.8
February	0.5	1.1	1.6	2.3	3.7
March	0.3	0.8	1.3	2.1	3.5
April	0.1	0.5	0.8	1.5	2.7
May	0.5	0.8	1.1	1.6	2.5
June	0.7	1.1	1.4	1.9	2.8
July	2.3	3.3	4.1	5.4	7.7
August	13.7	16.4	18.8	22	27.5
September	17.1	20	22.4	25.7	31
October	10.2	12.4	14.3	16.9	21.2
November	4.1	5.4	6.5	8.1	10.9
December	2.3	3.1	3.8	4.8	6.5

$$f_{Qt+1|Qt}(q_t|Q_j^{t+1}) = \begin{cases} \int_{x_1^{t+1}}^{\infty} f_{Qt|Qt+1}(q_t|q_{t+1})dq_{t+1} & (j=1) \\ \int_{x_1^{t+1}}^{x_{j-1}^{t+1}} f_{Qt|Qt+1}(q_t|q_{t+1})dq_{t+1} & (1<j<N) \\ \int_{-\infty}^{x_{N-1}^{t+1}} f_{Qt|Qt+1}(q_t|q_{t+1})dq_{t+1} & (j=N). \end{cases} \quad (19)$$

3. Repeat for all j ($j = 1, \ldots, N$). The entire procedure is repeated for each $t = 1, \ldots, 12$.

The discretization of the reservoir storage volumes is based on the study conducted by Klemes (1977) of the most commonly used schemes to discretize storage volumes. Storage discretization is modified due to the accumulated deposited sediments of the last 20 years totalling 2.5 $\times 10^9$ m^3 (Hydraulic Research Institute 1998).

Before applying the SDP, some of the AHD reservoir characteristics should be presented. The reservoir characteristics describing area, storage and elevation can be determined from ground and air survey contour maps (Master Water Plan 1981):

$$H = 79.97 + 0.0369\,V + 18.87\ln(V) \quad (20)$$

$$A = -3164.28 + 25.49\,V + 1092.92\ln(v) \quad (21)$$

where H is the reservoir elevation (m), A is the surface area (km^2) and V is the reservoir storage (billions of m^3 or BCM).

In fact, this non-linearity of the height-volume-surface area relationship influences the process of determining accurate water balances for the reservoir using Equation (8). This is due to the estimation of the reservoir water losses e_N (the water losses during the month), derived from surface area water level of the reservoir. In order to overcome this complexity, an iteration procedure has been introduced to calculate the water balance at each time-step within the SDP model. This iteration procedure is developed for each release option by assuming the final storage at each time-step then calculating the water losses considering the average values for the surface area and water level at the start and at the end.

Differences between the assumed final storage and the calculated storage are observed and the procedure is then repeated until an acceptable error between two consecutive calculations of final storage is reached. With reference to water balance Equation (8), it should be considered that for both flood and drought operations, two elevation constraints are mandatory. These are the maximum and minimum permissible reservoir elevation and were selected as 183 m and 150 m, respectively. Another consideration suggested by the Dam Authority is to maintain the level below 175 m on 31st July of each year, to allow for the next flood (Cairo University-MIT 1977).

HAD reservoir water losses originate from three main sources: evaporation, seepage and rock saturation or absorption losses. Some assumptions are made to represent the values of these losses either as a function of time and level or as a constant value. The evaporation losses can be expressed as a function of reservoir water area for a certain month n of a year t, written as:

$$G_N(t,N) = \alpha Vs(N)A(t,N) \quad (23)$$

where G_N is the evaporation ($\times 10^9$ m^3), α is a unit conversion coefficient, $Vs(N)$ denotes the monthly evaporation depth (mm), and $A(t,N)$ is the average reservoir surface area during month n of year t (km^2). The evaporation depth $Vs(N)$ is observed at both Aswan and Wadi Halfa metrological stations (south of Aswan) by the Piche evaporation method, and a weighted average value is then determined. The monthly distributions of Vs are as illustrated in Figure 6. It is assumed that Vs varies only according to the month in any year (Master Water Plan 1981).

For the seepage loss, an annual value of 0.10×10^9 m^3 is assigned. This is equivalent to 0.08×10^9 m^3 per month. The absorption losses depend on several factors such as the type

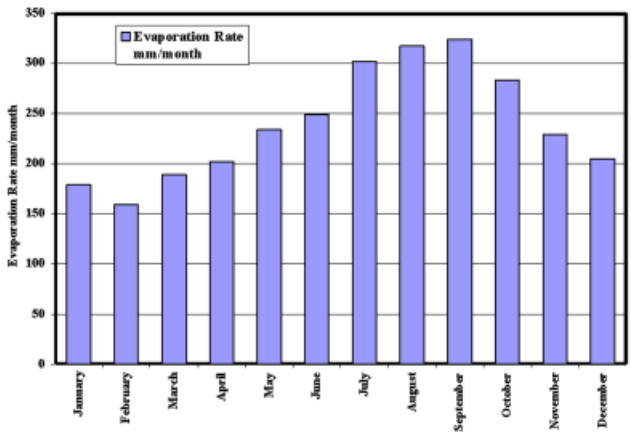

Figure 6 | Monthly evaporation rates for Lake Nasser.

of rocks covering the reservoir bed, the period of water covering the sides and if such covering takes place for the first time or not. Based on the mass balance principle of the lake, the measurement indicated that the amount 0.125×10^9 m^3 is a reasonable monthly average value for absorption losses (Master Water Plan 1981).

Model implementation

The development of an appropriate training set is critical to the success of any neural network. In the present study, the neural network must learn to make the same risk selections as the decision-maker, not only for the risk comparisons presented during the training session but also for risk comparisons generated within the SDP formulation. Without an adequate representation of the critical characteristics that define the risk curves, the resulting neural network would be unable to adequately duplicate the response of the decision-maker.

The preferences of the decision-makers to risk scenarios were developed in the following manner. Each of the two decision-makers were presented separately with a series of pairs of risk curves and asked to identify which of the two risk situations they preferred (or to indicate that they believed the two risk situations to be equivalent). The comparisons were actually performed by presenting each decision-maker with two risk curves in graphical rather than tabular form. The decision-maker was therefore asked to indicate risk preferences on the basis of a visual comparison of the two curves. The responses to the comparisons were then used to establish a risk profile or preference ranking for that decision-maker.

Ideally an infinite number of possible risk curves must be considered in the selection of those curves to be included in the training set for the neural network. In the present study, a total of 26 distinct risk curves were used during the sessions with the two decision-makers, resulting in training sets for the neural network with 676 (26×26) pair-wise comparisons. The 26 curves chosen for the training set represent a near uniform sampling of the risk space. To validate the choice of these 26 risk curves selected for use in training the neural network, the curves were compared to risk curves obtained when running the NN-SDP model with an untrained neural network. An untrained neural network can respond to external stimuli but, since the connections weighted are set to random values in such an untrained network, the output responses do not reflect the preferences of a decision-maker. The curves produced by running an untrained NN-SDP help to specify the range of curves that the NN-SDP model might encounter. The 26 risk curves proposed for training were then examined to determine if they adequately represented or covered the range of curves which the neural network might reasonably encounter in an actual application of the NN-SDP model.

It should be noted that the curves derived from the untrained NN-SDP are only indicative of the possible curves. They should not be mistaken for the curves that would be evaluated by a fully trained neural network. The curves generated by the untrained NN-SDP model were therefore not specifically included in the training set, and were only used to guide the selection of the final training set.

It might be observed that the 26 risk curves selected for the training set are inappropriately simplistic representations of the potential risk scenarios. However, it should be recognized that the curves used are only for the demonstration of the methodology and do not constitute a restriction on the full implementation of the model. There are in fact no limits to the number or shape of risk curves that may be included in the training set. However, the use of an extended training set to identify the optimal neural network architecture (the number of hidden neurons) and the optimal set of connection weights has resource implementation, namely time and computing power.

Development of the 676 risk comparisons for the two training sets for the neural network requires knowledge about how each of the two decision-makers responds to each and every pair-wise comparisons. With such a large number of selections having to be explicitly known through visual interpretation, the approach may initially appear to be overly cumbersome for practical use. However, on closer examination, a significant number of comparisons may be eliminated from the set to be presented to the decision-maker for consideration. For example, a comparison of curves 531 and 561 (Figure 7) would be consistent for all decision-makers, with curve 531 clearly dominant. Thus a reduced set of curves required for visual comparisons by the decision-maker can be identified. This approach resulted in 101 of the original 676 risk comparisons to be performed by visual investigation.

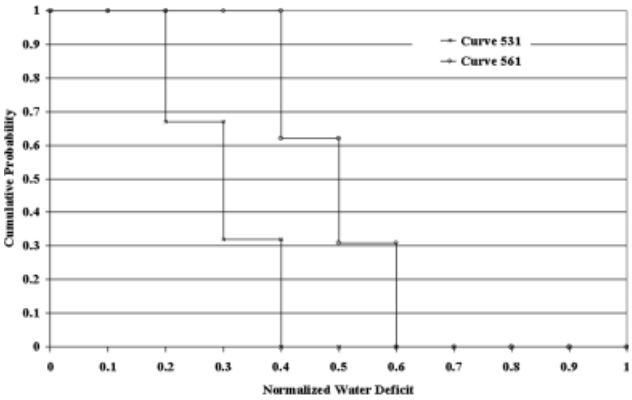

Figure 7 | Risk curve comparison involving clear preference.

Table 2 shows the risk rankings for the 26 curves derived from the risk preferences of decision-makers S1 and S2, as articulated in their responses to the visual comparisons of the 101 pairs of risk curves (second and third column, respectively). These rankings exhibit many similarities to the curve ranking in the fourth column of Table 2, which was based on ranking the curves from smallest to largest expected deficit (the curve with the lowest variance is preferred in cases where the risk curves have equal expectations). Similarities also exist between the two rankings derived from the two decision-makers. There appears to be a general consensus between the two decision-makers that the average water loss plays an important role in the ranking of the risk curves. Thus curve 500 (smallest average loss) is placed at the top of the ranking list and curve 600 (largest average loss) is placed at the bottom. Although the curves between these two extremes are not ranked strictly on average losses, there appears to be a trend to order the curves from smallest to largest average water loss in the two rankings.

The complete training set for each of the two decision-makers is generated directly from the preference rankings provided in Table 2. It is important to note that the curve preferences obtained during the visual analysis of the 101 pairs of curves were maintained in this set of 676 training pairs. This is an important safeguard on the consistency of the pair-wise preferences because not all responses obtained during the sessions with the decision-makers could be incorporated in a simple ranking scheme. Therefore, the ranking of curves is only used to obtain responses for comparisons not included in the visual analysis. Since only obvious comparisons are eliminated from the set of pairs of curves considered in the visual comparison, the use of the preference ranking is believed to adequately reflect the preferences of the decision-maker (the only way to establish the preferences of the decision-maker with absolute certainty is to perform all 676 comparison selections visually).

A final consideration for the establishment of a training set for the neural network is the adherence to the transproposition rule (recall that the transproposition rule states that if curve A is better than curve B and curve B is better than curve C, curve A is better than curve C). The validity of the sequential pair-wise comparison performed within the

Table 2 | Ranking of training risk curves for decision-makers SI and SII and expectation function

Rank	Decision-maker I	Decision-maker II	Expectation function
1	500	500	500
2	510	510	510
3	511	511	511
4	520	520	520
5	521	521	521
6	530	530	530
7	531	532	531
8	540	531	532
9	532	542	540
10	541	540	541
10	550	541	542
12	551	553	550
13	542	552	551
13	560	551	552
15	552	550	553
16	561	562	560
17	570	561	561
18	571	560	562
19	553	571	570
20	562	572	571
20	580	570	572
22	572	581	580
23	590	580	581
24	581	590	590
25	591	591	591
26	600	600	600

NN-SDP framework requires achieving the same final decision regardless of the order of the comparisons. This outcome can only be guaranteed if the risk preference ranking outlined above provides consistent decisions; the rankings are used only to obtain risk selections not established during the visual analysis.

Training of the network was performed using the Levenberg–Marquardt (LM) feed-forward back-propagation algorithms. A computer program was written using Matlab 6.5 software. Hyperbolic Tangent Sigmoid Function (HTSF) and Purelin Function (PF) were used as the transfer function in the hidden and output layers, respectively:

$$y = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}, \tag{24}$$

$$y = ax + b. \tag{25}$$

Model validation

The performance of the two neural networks was evaluated on the basis of their ability to duplicate the decisions of the two decision-makers. Thirty-five risk curves, in addition to the original 26 curves used in the training set, defined the set of pair-wise comparisons used for this analysis of performance. This addition of new risk curves to the validation set requires the network to react to, and make comparative evaluations about, curves it has not previously encountered. Such new risk situations provide a more comprehensive and unbiased evaluation of network performance that does not rely on the performance information obtained during training (e.g. average and maximum error in Table 3). The validation set of risk curve comparisons for each decision-maker arising from all 61 $(26+35)$ curves is made up of 3721 (61×61) pairs of curves. Each of these pair-wise comparisons has a known response or preference solicited from both decision-makers.

These responses or preferences were derived in the same fashion as for the original 26 curves used to train the networks, i.e. pairs of curves for which clear preferences were evident were identified and set aside. The remaining 191 pairs of curves were then evaluated by both decision-makers by visual comparison, and a ranking of the total 61 curves established. Responses to the remaining 3530 pair-wise comparisons were then derived from these rankings.

Recall that the validation set is used only to measure the ability of the neural network to duplicate the decision-making process. The validation set is not used to train the neural network and therefore does not incorporate the inconsistencies in that set within the neural network logic. Furthermore, the testing is directed towards the performance of the neural network in isolation rather than the complete NN-SDP model. Only neural networks trained with a minimal number of inconsistencies are used in the NN-SDP model itself. It should be noted that the 676 comparisons in the validation set repeated from the training set represent 16% of the complete test set of comparisons. Responses to these 676 comparisons should be identical to the responses elicited from the decision-maker which the network learns to mimic. This leaves 3045 (3721–676) risk scenarios for which the network has not learned the desired response and for which it must generalize. The network must be able to generalize if it is to successfully respond to these remaining risk comparisons. A network unable to generalize the knowledge gained during training cannot duplicate the attitudes of the decision-maker within the larger context of the NN-SDP framework, or the wide range of previously unseen risk curves the neural network is likely to encounter in that formulation. As such, the validation set provides the basis for a comprehensive measure of the performances of the networks.

An analysis of the responses of the two neural networks (N1, N2) to the set of validation curves yields the

Table 3 | Configurations of the neural networks N1 and N2

		Neural Networks	
		N1	N2
Number of neurons	Input layer	22	22
	Hidden layer	16	12
	Output layer	2	2
Learning parameters	α	0.01	0.01
	ß	0.005	0.005
Training performance	Average error	0.00018	0.0001
	Maximum error	0.011	0.0137
Number of iterations required for training		5000	5890

performance metrics presented in Table 4. Recall that the maximum error is defined by:

$$D(\omega_{\lambda ij}, \vartheta_{\lambda i}) = 0.5 \sum_{\mu=1}^{M} \sum_{\mu=1}^{N_L} (S_{Li}^{\mu} - \zeta_i^{\mu})^2 \quad (26)$$

where $D(\omega_{\lambda ij}, \vartheta_{\lambda i})$ is the sum of the squared difference between the desired and actual neural network responses, based on the synaptic weights and bias terms (referred to as D_t for iteration t); $S_{Li}^{\mu}$is the input signal to neuron i in the input layer L, resulting from input pattern μ; and $\zeta_i^{\mu}$is the input signal to neuron i in the input layer 0, resulting from input pattern μ.

On the other hand, the average error is the average of these squared deviations from the desired responses. While the small average errors support the use of neural networks to identify the desirable risk options, it can be seen that the networks make completely erroneous decisions when presented with certain risk combinations, as indicated by a maximum error of 1.00 for both networks.

In fact, the ability of the neural network to make the correct risk selection depends on the extent to which the two risk curves in the particular risk comparison resemble curves contained in the 26 curves included in the training set.

Risk comparisons containing one (or two) risk curve(s) completely dissimilar to any curves in the training set may require knowledge not internalized by the network during the training phase before a network response consistent with the attitudes of the decision-maker can be obtained. In this case, the particular characteristics of the new curve(s) may invoke an actual decision from the decision-maker which is in conflict with the decision-making process learnt by the network. This fact highlights the importance of selecting risk curves for the training set which are representative of the curves expected to be considered within the NN-SDP formulation. To further improve the performance of the neural network, the training set can be extended to include a larger number of risk comparisons.

Table 4 | Performance of the neural networks N1 and N2

		Neural networks	
		N1	N2
Testing performance	Average error	0.049	0.025
	Maximum error	1.0	1.0
Incorrect responses	Number out of 3721	334	201
	Percentage	9.0	5.2

RESULTS AND DISCUSSION

Water release policies were obtained from both the basic SDP formulation with its objective function of minimization of the expected deficits, and from the proposed NN-SDP models associated with the two decision-makers. NN-SDP release policies reveal that differences in risk attitudes of the two decision-makers fit with the same management practices of the reservoir as shown in Figures 8 and 9. The differences in preference rankings of the 26 risk curves, and the corresponding differences in the neural network responses to these risks, translate into differences in water release policies. The results obtained from NN-SDP imply an ability to identify specifically the attitudes towards risk of a decision-maker.

It should be noted that the curves are not ordered by the two decision-makers strictly according to average losses. For example, while decision-maker SI prefers curve 531 over curve 540 because of the smaller average loss of the former curve, curve 540 is preferred over curve 531 despite the contradiction that this preference imposes on keeping the average losses to a minimum. These contradictions to the general trend of ranking the curves according to average losses are not the same for both decision-makers. However, they do demonstrate that a tendency towards allowing other issues to influence the risk selection process exists for both decision-makers.

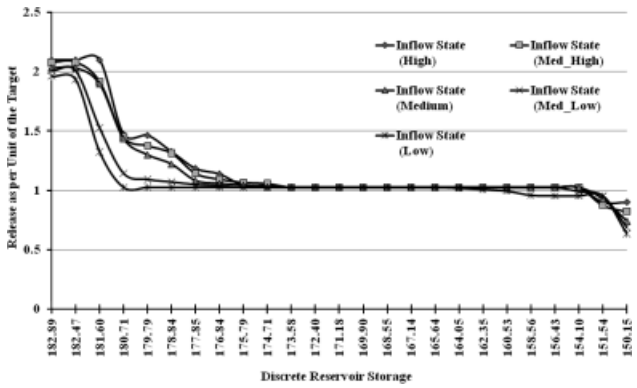

Figure 8 | Operation policy for AHD reservoir for the month of January (SDP).

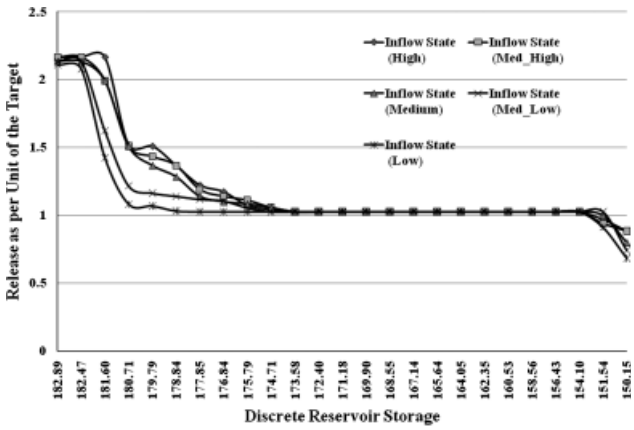

Figure 9 | Operation policy for AHD reservoir for the month of January (SDP_DM1).

Comparison of the results for each of the NN-SDP models and SDP formulation reveals that, in contrast to the policy defined by the expectation function, the NN-SDP models yield policies incorporating hedging and higher storage levels. Hedging refers to the practice of withholding water so that future water demands can be met (Bayazit & Unal 1990).

Clear differences in the policies generated from the basic SDP model and the NN-SDP models extend beyond the observed hedging. The performance metrics of reliability, resilience and vulnerability further reinforce the influence of the neural network approach on the selection of the optimal release decisions. Definitions for each of these three metrics were taken from a study conducted by Moy *et al.* (1986).

It can be observed that the criteria of performance (reliability and resilience as a conditional probability) are related to the frequency of water shortages, while the remaining criteria (resilience as duration of failure and vulnerability) describe the severity of these water shortages. The operation of AHD under the optimal policies derived from both the basic SDP formulation, as well as the policy common to the two NN-SDP models, was simulated using 20 years (1980–2000) of historical data. The simulation of the reservoir for the annual demands scenarios of 55.5×10^9 m^3 yields the performance metrics shown in Table 5 for the annual demand targets.

The policy derived from NN-SDP models created an operating environment that is more reliable and resilient than the reservoir controlled by the optimal policy generated by the basic SDP model with its expectation objective function. This improvement in the frequency of water shortage is however achieved at the expense of the vulnerability of the reservoir. The reservoir release policy generated from the NN-SDP (decision-maker) model clearly results in larger water deficits. For example, the average water deficit for the NN-SDP model is 0.85×10^9 m^3 compared to 0.5×10^9 m^3 under the optimal policy generated from the basic SDP. It is interesting to note that these findings are at odds with the research by Bayazit & Unal (1990) who concluded that policies incorporating hedging provide less reliable systems while reducing their vulnerability.

It is not clear why the hedging strategy observed did not affect the performance as predicted. However, a possible hypothesis for the increase in vulnerability is that the hedging strategy of withholding water for future use is maintained even in situations where large shortfalls are being experienced due to the low availability of water in the reservoir. Considering this observation it could be noted that, when evaluating the volumetric reliability during the 20 year period, simulation is higher than the time-based reliability. It can be

Table 5 | Reservoir performance index of release policies associated with an annual demand of 55.5×10^9 m^3

		Release policy	
		Expectation objective function	Neural network
Reliability	Time-based	92%	94%
	Volumetric	96%	96%
Resilience (condition probability)		25%	50%
Resilience (max. no. of months)		4	2
Vulnerability		38% of the demand	46% of the demand
Average deficit (10^9 m^3)		0.5	0.85

seen from Table 5 that the volumetric reliability for both models are 96% which is higher than the time-based reliability. This is consistent with the decision-makers choosing to withhold water for future uses, although water is available in the reservoir. This observation is also noticed when applying the SDP with the same level of volumetric reliability.

In other words, the release strategy does not recommend adequate water releases which are necessary to safeguard against the maximum water deficits. Verification of this hypothesis requires further work. Nevertheless, the relationships between reliability and vulnerability and between resilience and vulnerability were observed in the results shown in Table 5. Reliable and resilient reservoirs are therefore vulnerable to large deficits.

The final performance criterion provided in Table 5 is the average deficit experienced during the 20 years of the simulation. The average deficit experienced under the NN- SDP releases is up to 1.5 times greater than that experienced under the SDP releases. However, it must be recognized that the expectation objective function drives the solution process such that this average deficit is minimized. The fact that the NN-SDP formulation encapsulates a more complete representation of the risks associated with the release decisions alters the search for the optimal policy with correspondingly less emphasis on the size of the deficit in isolation. The more complete representation of the risks simply allows important concerns other than long-term performance to influence the operation of the reservoir.

CONCLUSIONS

The present study has introduced an improved technique whereby the attitudes towards risks (the uncertainty of inflow) of a single decision-maker could be comprehensively incorporated to operate the AHD reservoir. The NN-SDP formulation proposed to address this problem is able to identify the optimal release policy for a reservoir, as defined by the attitudes of the decision-maker and learnt by the neural network. The resulting integrated model allows the attitudes towards risk of a decision-maker to be considered explicitly in defining the optimal release policy.

The approach was demonstrated by application to the operation of the AHD reservoir. The release policies derived from the NN-SDP model incorporated the neural networks training using each of the two decision-maker's preferences, which exhibited hedging. The storage levels for the policies were consistently higher than those defined by SDP formulation, which used a traditional expectation-based objective function, i.e. the expected deviations from target demands. These results suggest that the decision-makers are willing to forego water in the short term to guard against the possibility of large water deficits in the future. However, while such hedging strategies normally tend to reduce the vulnerability of the reservoir to large deficits at the cost of lower reliability and resilience, the hedging observed in the NN-SDP policy actually resulted in a more reliable and resilient (but also more vulnerable) system compared to the policy generated from the SDP model with its expectation objective.

ACKNOWLEDGEMENTS

This research is supported by the first author's grant from Smart Engineering Research Group, University Kebangsaan Malaysia (UKM), and eScience Fund 01-01-02-SF0581 from Ministry of Science, Technology and Innovation (MOSTI), Malaysia. Nile Sector, Ministry of Water Resources and Irrigation, Egypt is also acknowledged for providing the natural inflow for monitoring station datasets used in this research.

REFERENCES

Bayazit, M. & Unal, N. E. 1990 Effects of hedging on reservoir performance. *Water Resources Research* **26**(4), 713–719.

Bellman, R. 1959 *Dynamic Programming*. Princeton University Press, Princeton, N.J.

Bishop C. M. 1996 *Neural Networks for Pattern Recognition*, 1st edition. Oxford University Press, Oxford.

Butcher, W. 1971 Stochastic dynamic programming for optimum reservoir operation. *Water Resour. Bull.* **7**(1), 115–123.

Cairo University and Massachusetts Institute of Technology Technological Planning Program 1977 *Stochastic Modeling of Nile Inflows to Lake Nasser*. Cairo University, Cairo.

Chandramouli, V. & Raman, H. 2001 Multireservoir modeling with dynamic programming and neural networks. *J. Water Resour. Plann. Manage.* **127**(2), 89–98.

Chang, L. C. & Chang, F. J. 2001 Intelligent control for modeling of real-time reservoir operation. *Hydrolog. Process* **15**(9), 1621–1634.

Chang, Y. T., Chang, L. C. & Chang, F. J. 2005 Intelligent control for modeling of real-time reservoir operation. II: Artificial neural network with operating rule curves. *Hydrolog. Process* **19**(7), 1431–1444.

Chaves, P. & Kojiri, T. 2004 Reservoir optimization through stochastic dynamic programming with fuzzy conditional probability. In *Proceedings of 2004 Annual Conference of the Japanese Society of Hydrology and Water Resources*, Hokkaido, Japan, pp. 32–33.

Chaves, P., Kojiri, T. & Yamashiki, Y. 2003 Optimization of storage reservoir considering water quantity and quality. *Hydrolog. Process* **17**(14), 2769–2793.

Garrick B. J. & Kaplan S. 1981 On the quantitative definition of risk. *Risk Analysis* **1**(1), 11–27.

Gibson G. J. & Cowan C. F. N. 1990 On the decision regions of multilayer perceptrons. *Proceedings of the IEEE* **78**(10), 1590–1594.

Goulter, I. C. & Tai, F. K. 1985 Practical implication in the use of stochastic dynamic programming for reservoir operation. *Water Resour. Bull.* **2**(1), 65–74.

Haimes, Y. Y., Kaplan, S. & Lambert, J. H. 2002 Risk filtering, ranking, and management framework using hierarchical holographic modelling. *Risk Analysis* **22**(2), 383–397.

Haykin S. 1999 *Neural Networks: A Comprehensive Foundation*. Prentice Hall, New York.

Howard, R. A. 1960 *Dynamic Programming and Markov Process*. MIT Press, *Cambridge, Mass*.

Hydraulic Research Institute 1998 Calibration of Toshka Spillway, Technical Report.

Kaplan, S. 2000 Combining probability distributions from experts in risk analysis. *Risk Analysis* **20**(2), 155–156.

Kaplan, S., Haimes, Y. Y. & Garrick, B. J. 2001 Fitting hierarchical holographic modeling into the theory of scenario structuring and a resulting refinement to the quantitative definition of risk. *Risk Analysis* **21**(5), 807–819.

Kelman, J. & Stendinger, J. R. 1990 Sampling stochastic dynamic programming applied to reservoir operation. *Water Resour. Res.* **26**(3), 447–454.

Klemes, V. 1977 Discrete representation of storage for stochastic reservoir optimization. *Water Resources Research* **13**(1), 149–158.

Labadie, J. W. 2004 Optimal operation of multireservoir systems: State-of-the-art review. *J. Water Resour. Plann. Manage.* **130**(2), 93–111.

Master Water Plan 1981 *Hydrological Simulation of Lake Nasser*, technical report No. 14 (UNDP/EGY/73/024), Ministry of Public Works and Water Resources, Cairo Egypt.

Moy, W.-S, Cohon, J. L. & ReVelle, C. S. 1986 A programming model for analysis of the reliability, resilience, and vulnerability of water supply reservoir. *Water Resources Research* **22**(4), 489–498.

Ooyen, V. & Nienhuis, B. 1992 Improving the convergence of the backpropagation algorithm. *Neural Networks* **5**, 465–471.

Simonovic, S. P. & Burn, D. H. 1989 An improved methodology for short-term operation of a single multipurpose reservoir. *Water Resources Research* **25**(1), 1–8.

Xu Z., Jinno, K., Kawamura, A., Takesaki, S. & Ito, K. 1998 Performance risk analysis for Fukuoka water supply system. *Water Resources Management* **12**, 13–30.

Yeh, W. W.-G. 1985 Reservoir management and operations models: a state-of-the-art review. *Water Resources Research* **21**(12), 1797–1818.

First received 5 April 2009; accepted in revised form 5 February 2010. Available online December 2010