Journal of Hydroinformatics | 10.4 | 2008

Neural-network-based simulation-optimization model for water allocation planning at basin scale

M. Shourian, S. Jamshid Mousavi, M. B. Menhaj and E. Jabbari

ABSTRACT

Heuristic search techniques are highly flexible, though they represent computationally intensive optimization methods that may require thousands of evaluations of expensive objective functions. This paper integrates MODSIM, a generalized river basin network flow model, a particle swarm optimization (PSO) algorithm and artificial neural networks into a modeling framework for optimum water allocations at basin scale. MODSIM is called in the PSO model to simulate a river basin system operation and to evaluate the fitness of each set of selected design and operational variables with respect to the model's objective function, which is the minimization of the system's design and operational cost. Since the direct incorporation of MODSIM into a PSO algorithm is computationally prohibitive, an ANN model as a meta-model is trained to approximate the MODSIM modeling tool. The resulting model is used in the problem of optimal design and operation of the upstream Sirvan river basin in Iran as a case study. The computational efficiency of the model makes it possible to analyze the model performance through changing its parameters so that better solutions are obtained compared to those of the original PSO–MODSIM model.

Key words | ANNs, basin-wide water management, MODSIM, optimization, PSO, simulation

M. Shourian
S. Jamshid Mousavi (corresponding author)
School of Civil Engineering,
Amirkabir University of Technology
(Tehran Polytechnic), Tehran,
Iran
E-mail: *jmosavi@aut.ac.ir*

M. B. Menhaj
School of Electrical Engineering,
Amirkabir University of Technology
(Tehran Polytechnic), Tehran,
Iran

E. Jabbari
Department of Civil Engineering,
Iran University of Science and Technology,
Tehran,
Iran

INTRODUCTION

To bring the concept of integrated water resources management into an analytical framework, modeling techniques for integrating hydrologic, agronomic, economic and institutional components have been studied and introduced to provide opportunities for the advance of water resources management (McKinney *et al.* 1999). Simulation models have been used for river basin systems modeling since they allow a detailed representation of the system's characteristics; however, they do not identify the optimal design and operating policies.

The application of traditional constrained optimization algorithms to river basin systems' analysis may be limited. This is due to the complexity of the systems including several components like reservoirs, aquifers, pumping systems, hydroelectric power plants, demand sites, etc. Besides, there may be various phenomena and relationships represented by highly nonlinear, nonconvex or discontinuous equations involving hydrologic, economic, social and institutional aspects regarding water quantity and quality, surface and groundwater, and land resources. Multi-period linear optimization, especially network flow programs (McBride 1985; Kuczera & Diment 1988; Kuczera 1993; Sun *et al.* 1995; Hsu & Cheng 2002; Jenkins *et al.* 2004) and in some cases nonlinear programming or evolutionary algorithms (McKinney *et al.* 1999; Cai *et al.* 2002, 2003) have been used for optimizing river basin systems operation and design.

Simulation-optimization methods linking a detailed simulation model with a heuristic or population-based evolutionary algorithm (EA) are becoming increasingly

doi: 10.2166/hydro.2008.057

attractive for solving optimization models. The advantage of EAs lies in their ability to locate good solutions to combinatorial optimization problems with greater efficiency than implicit enumeration techniques (Wagner 1995*b*). They are also advantageous because they accommodate the discontinuities and nonlinearities of real functions more easily than gradient-based techniques do in which constraints and functions must be represented as algebraic equalities or inequalities (McKinney & Lin 1994). Another advantage of gradient-free and evolutionary optimization techniques is the possibility of using any kind of built-up simulation package without having access to its embedded source codes or detailed equations.

Although the approaches are advantageous, their computational cost may be very high when a time-consuming simulation model is performed for objective function evaluations. This difficulty may be dealt with by either simplifying the original problem by analyzing smaller-scale situations and using simpler models (Wagner 1995*a*) or seeking to reduce the number of times the simulation model must be called by increasing the efficiency of the search algorithm (Karatzas & Pinder 1993; Karatzas 1997). Traditionally, methods to overcome the computational cost of simulation models within an optimization framework are grouped into two categories: (1) methods that reduce the execution time required for the simulation model through parallel algorithms and computer architectures (Dougherty 1991; Tompson *et al.* 1994); and (2) methods that use an approximation of the simulation model, called a meta-model, to quickly supply predictions during the course of the search (Johnson & Rogers 2000). This latter approach is the focus of the current study.

The idea of using an approximate model to replace an extensive simulation model is quite old in water resource systems modeling. For example, in the unit response matrix method used in conjunctive-use applications a set of linear equations replaces a groundwater model when the model behavior can be reasonably assumed to be linear. Alley (1986), Lefkoff & Gorelick (1990) employed multiple linear regression equations as substitutes for groundwater simulation models.

Artificial neural networks (ANNs), as function approximators and meta-models, have shown different applicability in various engineering problems. ANNs impose fewer constraints on the functional form of the relationships between input and output variables, making them a logical choice for application when the complexity of the mapping is difficult to anticipate. Multilayer perceptrons trained by a backpropagation learning algorithm have been successfully used in modeling complex relations such as rainfall–runoff processes (Smith & Eli 1995), prediction of daily stream flows (Sureerattanan & Phein 1997), forecasting water quality parameters (Maier & Dandy 1996), inferring reservoir operating rules (Raman & Chanramoulia 1996; Ponnambalam *et al.* 2003; Mousavi *et al.* 2007), groundwater systems operation and conjunctive-use modeling (Ranjithan *et al.* 1993; Rogers & Dowla 1994; Johnson & Rogers 2000). Broad *et al.* (2005) used ANNs as metamodels to optimize a water distribution design problem including water quality. Yan & Minsker (2006) proposed a dynamic modeling approach, called adaptive neural networks genetic algorithm, in which ANNs are adaptively trained directly within a genetic algorithm to replace a time-consuming groundwater simulation model.

The model developed in this study integrates MODSIM as the simulation module with a PSO algorithm as an optimization tool for optimum water allocations in the upstream Sirvan river basin in the west of Iran. The resulting model is the PSO–MODSIM model. As the model is highly time-consuming, MODSIM is replaced by a trained ANN model and the resulting PSO–MODSIM–ANN model is then applied to the considered water allocation optimization problem. The outcomes of both models are further analyzed and compared in terms of their quality of solutions and computational loads.

RIVER BASIN SIMULATION MODULE

MODSIM (Labadie 1995) represents a valuable tool to simulate operations of any complex river basin system as a network consisting of nodes and links. MODSIM sequentially solves the following one-period linear optimization problem in each time period over the planning horizon using an efficient minimum cost network flow program:

$$\text{Minimize} \sum_{l \in A} c_l q_l \quad (1)$$

Subject to:

$$\sum_{j \in O_i} q_j - \sum_{k \in I_i} q_k = 0; \text{ for all } \quad i \in N \quad (2)$$

$$l_l \le q_l \le u_l; \quad \text{for all} \quad l \in A \quad (3)$$

In the above, A is the set of all arcs or links in the network, N is the set of all nodes, O_i is the set of all links originating at node i (i.e. outflow links), I_i is the set of all links terminating at node i (i.e. inflow links), q_l is the integer-valued flow rate in link l, c_l are costs, weighting factors or priorities per unit of flow rate in link l, l_l is the lower bound on flow in link l and u_l is the upper bound on flow in link l. Relation (2) represents the mass balance equation that must be satisfied at every node of the model's network. As an example, a fully circulating network is shown in Figure 1. Nodes 1, 2 and 3 are actual, physical system nodes, where node 1 is a reservoir, node 3 is a demand diversion and node 2 is an intermediate node. Details on modeling river basin components in MODSIM and calculation schemes of return flows, stream depletion from pumping and canal seepage can be found in Fredericks *et al.* (1998).

One of the features of MODSIM's latest version (version 8.0) is the ability of preparing customized codes

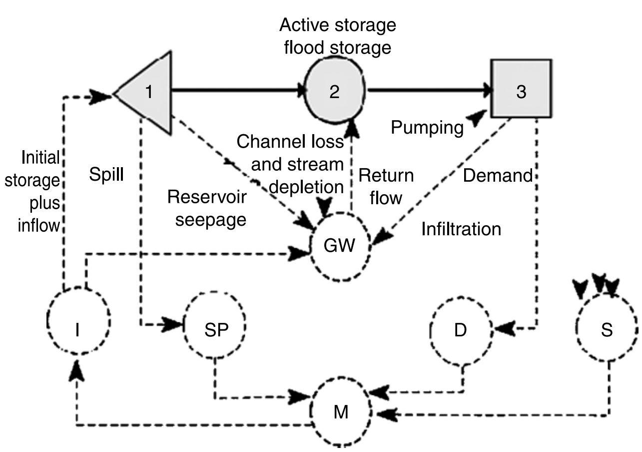

Figure 1 | Network structure for MODSIM with accounting nodes and links.

in the VB.NET or C.NET languages that are compiled with MODSIM through the.NET Framework. Users are provided with access to all key variables and object classes without the need for reprogramming and recompiling the MODSIM source code. Customization granted in MODSIM 8.0 is a prominent feature which a few computerized river basin DSS models developed so far support. Taking advantages of MODSIM's custom coding features, it has been embedded in a PSO algorithm in this study to solve a river basin system optimization problem.

THE PARTICLE SWARM OPTIMIZATION ALGORITHM

The PSO algorithm, originally proposed by Kennedy & Eberhart (1995), Eberhart & Kennedy (1995) is a member of the wide category of swarm intelligence methods for solving global optimization problems. In a PSO algorithm, each particle is a candidate solution equivalent to a point in a D-dimensional space; hence the ith particle's position can be represented as $x_i = (x_{i1}, x_{i2}, \dots, x_{iD})$. Each particle flies through the search space, depending on two important positions, $p_i = (p_{i1}, p_{i2}, \dots, p_{iD})$, the best position the current particle has found so far (*pbest*), and $p_g = (p_{g1}, p_{g2}, \dots, p_{gD})$, the global best position identified in the entire population (*gbest*). The rate of the ith particle's position change is given by its velocity $v_i = (v_{i1}, v_{i2}, \dots, v_{iD})$. Equation (4) updates the velocity for each particle in the next iteration, whereas Equation (5) updates each particle's position in the search space:

$$v_{id}^{n+1} = \chi(\omega v_{id}^n + c_1 r_1^n (p_{id}^n - x_{id}^n) + c_2 r_2^n (p_{gd}^n - x_{id}^n)) \quad (4)$$

$$x_{id}^{n+1} = x_{id}^n + v_{id}^{n+1} \quad (5)$$

where $d = 1,2,\dots,D$; $i = 1,2,\dots,N$ and N is the size of the swarm; χ is a constriction factor used in constrained optimization problems in order to control the magnitude of the velocity. It is usually set to 1.0 in unconstrained optimization problems. ω is called *inertia weight*; c_1, c_2 are two positive constants, called *cognitive* and *social* parameters, respectively; r_1, r_2 are random numbers uniformly distributed in [0,1]; and $n = 1,2,\dots$, denotes the iteration number.

An initial value of 1.2 gradually declining towards 0 can be considered as a good choice for ω (Shi & Eberhart

1998a,b). Therefore, the PSO updates the inertia weight in each iteration using the following equation:

$$w(\text{iter}) = w_{\max} - \frac{w_{\max} - w_{\min}}{\text{iter}_{\max}} \times \text{iter} \quad (6)$$

where w_{iter} is the iteration's inertia weight, $\text{iter}_{\max}$ is the maximum iteration number and $w_{\max}$ and $w_{\min}$ are, respectively, the maximum and minimum inertia weights. It has been reported that it might be better to choose c_1 and c_2 such that $c_1 + c_2 \leq 4$ and c_1 bigger than c_2. The PSO algorithm starts with a set of randomly generated solutions. Then the swarm is updated using Equations (4) and (5) in each iteration. This process is repeated until no further improvement is obtained for the objective function value.

It has been shown that the trajectories of particles oscillate in different sinusoidal waves and converge quickly, sometimes prematurely. During each iteration, particles are attracted towards the *pbest* and *gbest* positions and will eventually lose their exploration capability during future iterations. In order to prevent the premature convergence of the algorithm, in addition to the standard PSO a strategy may be employed to drive the particles and allow them to further explore the decision space. If a particle's velocity decreases to a threshold v_c, a new velocity is assigned using Equation (7). Thus, a turbulent PSO (TPSO) (Liu & Abraham 2001) is used in this study in which the following new velocity update equation is employed:

$$v_{id}^{n+1} = \begin{cases} v_{id}^{n+1} & if|v_{id}^{n+1}| \geq v_c \\ u(-1,1)v_{\max}/\rho & if|v_{id}^{n+1}| \leq v_c \end{cases} \quad (7)$$

where $u(-1,1)$ = a random number uniformly distributed in the interval $[-1,1]$, ρ = a scaling factor which controls the domain of the particle's oscillation according to $v_{\max}$ and v_c = the minimum velocity threshold, a tunable threshold parameter to limit the minimum of the particles' velocity. A large v_c shortens the oscillation period and provides a large probability for the particles to leap over local minima using the same number of iterations. However, a large v_c compels particles in the quick "flying" state, forcing them neither to search the solution nor to refine the search. The search ability can be adjusted by varying v_c dynamically. For the desired exploration–exploitation trade-off, it is better to divide the search procedure into three stages. In the first stage the values for v_c and ρ are set as large and small values, respectively. In the second stage, v_c and ρ are set as medium values and in the last stage, v_c is set as a small value and ρ as a large one. This study employs the PSO algorithm as the main optimization technique to deal with a river basin system management problem.

FEED-FORWARD NEURAL NETWORKS

A feed-forward neural network with backpropagation learning algorithm is trained in this study to replace MODSIM–DSS used as the simulator in the PSO algorithm. The trained network approximates the functional relationship between decision variables of the underlying river basin system optimization problem and the resulting total cost of the system's design and operation. Therefore, this section presents a brief review of the multilayer feed-forward neural networks.

Artificial neural networks are functions that can be trained to map nonlinear complex relations. Sets of input data and their corresponding output vectors are needed to train the network. Once properly trained, the network provides a data-driven model which is capable of giving reasonable answers when presented with input vectors that have not been encountered during the training process. The key to successfully training an ANN is choosing the right network architecture and training algorithm. Feed-forward networks are a subclass of layered networks in which there is no intra-layer connections and whose main feature is that connections are allowed from a node in layer i only to nodes in layer $i + 1$ (Figure 2).

Feed-forward neural networks are among the most common networks in use. The feed-forward process involves presenting an input pattern to neurons that pass the values into the first hidden layer. Determining the architecture of a neural network involves ascertaining the number of layers in the network as well as the number of nodes (neurons) in each layer. It also entails designating the type of transfer (activation) function to be used in each layer. A backpropagation training algorithm can be used to modify the weights so as to minimize the error between the desired and actual outputs of the network. Once trained, the

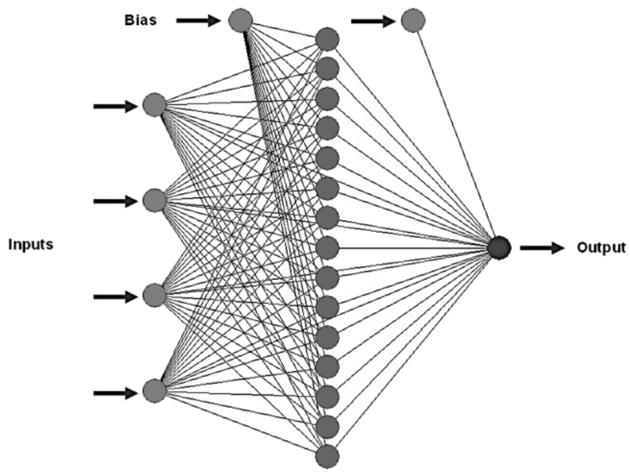

Figure 2 | A feed-forward neural network with one hidden layer.

network weights are frozen and can be used to compute output values for new input samples.

THE UPSTREAM SIRVAN RIVER BASIN SYSTEM

The upstream Sirvan river basin, located in the west of Iran, is considered as a case study. Because available surface water resources in the basin exceed the total water demands, the basin's management plans should consider the construction of infrastructure for water transfers to the neighboring basins. This mainly includes transferring water from the Sirvan to the Karkheh basin, as one of the largest basins in the country with high hydropower and agricultural demands, and transferring water to Ghorveh. Table 1 presents the total annual inflows and demands inside and outside of the Sirvan basin. Figure 3 shows the existing reservoirs and the ones to be constructed (triangles) in the upstream Sirvan basin and the demand nodes (squares) out of the basin. Adding the inside-the-basin demand nodes, Figure 4 shows the detailed topology of the system, represented in MODSIM's Graphical User Interface.

Table 1 | Annual amounts of the inflows and demands of the upstream Sirvan river basin

Amount (MCM)	Item
1719.5	Total inflow to the basin
379.7	Inside the basin demands
431	Upstream Karkheh basin's demand
157.7	Ghorveh system's demand
738.3	Karkheh basin's demand
1706.7	Total demands

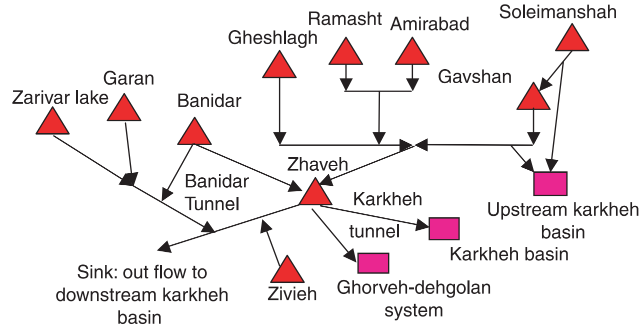

Figure 3 | General plan of the upstream Sirvan basin and water transfer systems.

Among the storage nodes, the capacities of the Banidar and Zhaveh reservoirs, due to the existence of a high topographic potential, need to be defined optimally. Other storage nodes are either under operation or, because of a small potential for their capacities, could have a local demand–supply role, resulting in considering a fixed capacity for them. Table 2 presents the dead storages and the maximum capacities of the reservoirs of the system. Therefore, the design variables of the optimization problem include the Banidar and Zhaveh reservoirs' capacities which affect the capacities of the water transfer systems (i.e. the Banidar and Karkheh tunnels and the Ghorveh pumping system). Topographic conditions in the Banidar and Zhaveh sites limit the maximum allowable capacity to 348.5 and 1,108 million cubic meters (MCM) for the Banidar and Zhaveh reservoirs, respectively. Finding a zero capacity for each reservoir would imply an unbeneficial aspect for construction of the dam.

MODSIM uses a priority-based algorithm to allocate optimally water to demand and storage nodes. These priorities indicate the relative significance between meeting water demands and satisfying reservoirs' target storages in the system. According to the management policies supposed, the priority order of the demands is considered below:

Priority 1:Environmental demands.

Priority 2:Inside the basin and upstream Karkheh basin's municipal demands.

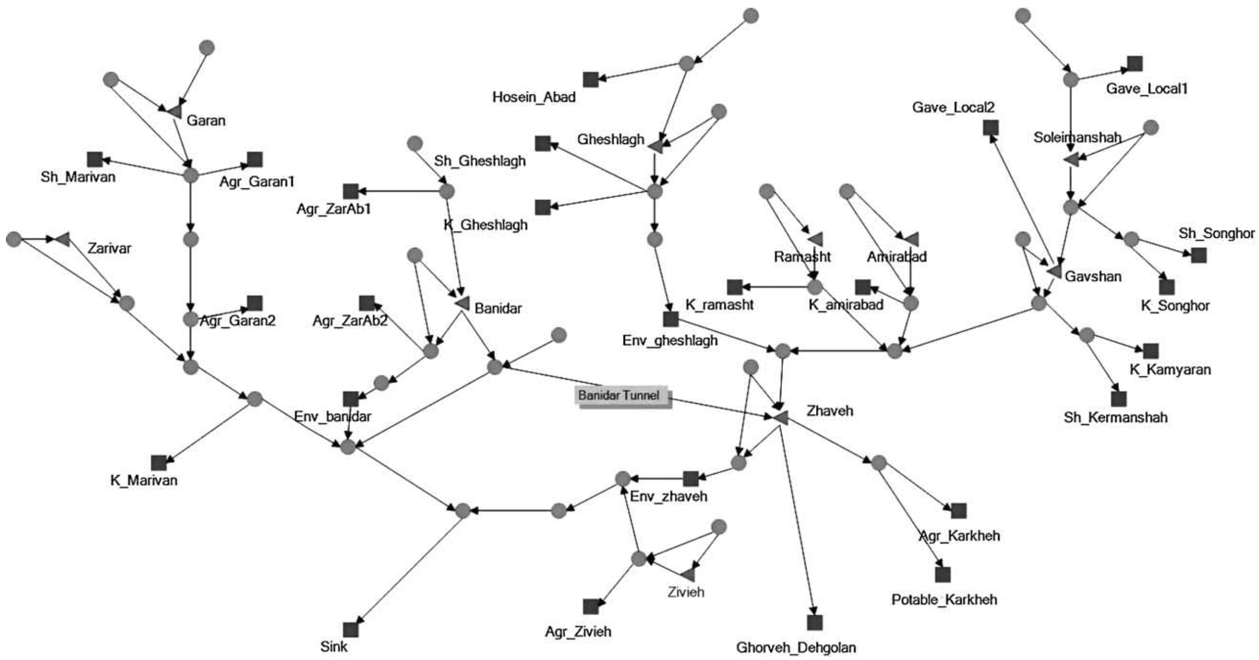

Figure 4 | Representation of the upstream Sirvan river basin and neighboring basins in MODSIM.

Priority 3:Inside the basin and upstream Karkheh basin's agricultural demands.
Priority 5:Karkheh basin's municipal demands.
Priority 6:Karkheh basin's agricultural demands.
Priority 8:Ghorveh – Dehgolan's demand.

Although MODSIM sequentially solves single-period network flow programs (NFPs), multi-period modeling aspects could be somehow indirectly considered through relative priorities of reservoirs' storage targets. Assigning a higher priority to a reservoir target storage relative to the priority of its downstream demands would direct the NFP not to empty the reservoir to meet the downstream demand at the current time step, resulting in storing water in the reservoir as close as possible to the target storage. This provides a partial foresight for single-period NFPs to account for long-term considerations. Therefore, the priorities of the reservoirs' target storages are selected as MODSIM's customized operational variables whose optimum values are to be determined by the PSO algorithm. These priorities are varied between 1 and 8 while the fixed demand priorities are input to MODSIM. Priorities of the Karkheh basin's demands have been set to 5 and 6 to make the opportunity for reservoirs to stay at a lower priority number, say 4, resulting in storing water in case of it being more beneficial in comparison with transferring water to the Karkheh basin. Likewise, a priority of 8 is considered for the Ghorveh and outflow nodes, providing a wider range for selecting the priorities of the system reservoirs. A negative coefficient shows the benefit of water transferring and a positive coefficient represents the cost of water transferring to the end node of the link. Therefore, the model considers the Ghorveh transfer system as a pumping system through assigning a positive coefficient for the associated link.

Table 2 | Capacities of the reservoirs of the upstream Sirvan river basin

Dead storage (MCM)	Max. capacity (MCM)	Reservoir
16	18	Zarivar Lake
5.5	86.5	Garan Dam
25	224	Gheshlagh Dam
0.5	8	Ramasht Dam
0.4	7.3	Amirabad Dam
12	40.4	Soleimanshah Dam
67.8	560	Gavshan Dam
0.5	16	Zivieh Dam
–	Variable	Zhaveh Dam
–	Variable	Banidar Dam

Gravitational water transfer from the Zhaveh reservoir to the Karkheh basin requires Zhaveh to have a minimum operating level of 1,340 m asl (meters above sea level), corresponding to a minimum reservoir capacity of 516.2 MCM. Consequently, if the Zhaveh reservoir capacity has a value less than 516.2, a pumping system is required and a cost coefficient for the link connecting Zhaveh to the Karkheh demand node should be considered; otherwise, the cost coefficient of the Karkheh link would be set to zero. Moreover, socio-economic studies show that increasing the size of the Zhaveh reservoir may cause remarkable damage by inundating agricultural lands and heritages and the need to move local residents. Therefore, for capacities less than 90 MCM, between 90 and 600 MCM, and greater than 600 MCM coefficients of 0, 100 and 200 per unit storage volume of the Zhaveh capacity are, respectively, assigned for the damage cost of the reservoir. These kinds of constraints imposed on the model could be easily considered in simulation-based optimization algorithms like PSO. However, they have to be considered via binary variables in classical optimization methods, resulting in a mixed-integer nonlinear (nonconvex) program which is difficult to solve. Other input data including natural and inter-basin flows, monthly municipal and agricultural water demands, etc., have been introduced to MODSIM over a 20-year period of records.

THE PSO–MODSIM MODEL

In spite of all of MODSIM's remarkable capabilities, it is not a fully dynamic optimization model since it uses a single-period optimization at each time step to simulate a river basin system operation. Consequently, MODSIM, *per se*, would not be able to obtain the optimum design and operation of the system components. To achieve this goal, there is a need to link the MODSIM simulation module with an optimization procedure to find the optimal values for the design and operational variables of the river basin system under study. By coding the PSO algorithm in MODSIM's custom coding environment, the values of decision variables, as input parameters of MODSIM are generated by the PSO algorithm. During the execution of the PSO–MODSIM model, for each candidate solution generated by the PSO algorithm, the objective function is evaluated through execution of MODSIM, thus obtaining the costs and benefits related to the values of the design and operational variables.

In the PSO–MODSIM model, the decision variables are the size of water storage and transfer facilities or pumping systems and the priorities for target storages, through which single-period inner NFPs are linked with a long-term outer optimization model. PSO feeds those decision variables into the inner NFP algorithms. Subsequently, the amounts of water allocated to demand nodes in each time step determined by the NFPs are returned from MODSIM to PSO in order to evaluate the PSO objective function. The procedure is repeated and the PSO decision variables, i.e. particles' values, are evolved towards their optimal values using the PSO updating rules until some stopping criteria are met.

The model's objective function assumed is a simplified function defined according to the scope of this study and not necessarily the one reflecting all of hydrologic, agronomic, socio-economic, environmental and institutional aspects which may exist in the real system. Considering this point, the PSO objective function consists of cost and benefit terms. Costs of facilities to be constructed, viz. new dams and tunnels, include investment fixed costs and variable costs which depend on the size of the facilities. Benefits gained are related to meeting municipal or agricultural water demands. The PSO objective function may be expressed as

$$
\begin{aligned}
Cost = {} & M_1 \times \mathrm{CapitalCost}_{\mathrm{Zhaveh-Dam}} + M_2 \\
& \times \mathrm{CapitalCost}_{\mathrm{Banidar-Dam}} + M_3 \\
& \times \mathrm{CapitalCost}_{\mathrm{Ghorve-Pumpage}} + M_4 \\
& \times \mathrm{CapitalCost}_{\mathrm{Karkheh-Transfer}} + M_5 \\
& \times \mathrm{CapitalCost}_{\mathrm{Banidar-Tunnel}} + C_{\mathrm{cap.}} \times (\mathrm{Cap}_{\mathrm{Zhaveh}} \\
& + \mathrm{Cap}_{\mathrm{Banidar}}) + C_{\mathrm{Damage}} \times \mathrm{Cap}_{\mathrm{Zhaveh}} + C1_{\mathrm{Pumpage}} \times \\
& Q_{\mathrm{Ghorveh}} \times H_{\mathrm{Ghorveh}} + M_6 \times C2_{\mathrm{Pumpage}} \times Q_{\mathrm{Karkheh}} \times \\
& H_{\mathrm{Karkheh}} + M_7 \times C1_{\mathrm{Diameter}} \times Q_{\mathrm{Karkheh}} + C2_{\mathrm{Diameter}} \times \\
& Q_{\mathrm{Banidar}} - \sum_{t=1}^{T} \sum_{j=1}^{\mathrm{NDEM}} C_{\mathrm{Benefit}} \cdot Qdem_{t,j}
\end{aligned}
$$

As described before, the pumping height from the Zhaveh to Karkheh basins is equal to the difference between the minimum operating level of the Zhaveh reservoir and the level of 1,340 m asl (corresponding to the Zhaveh reservoir capacity of 516.2 MCM). If the Zhaveh reservoir's minimum operating level exceeds 1,340 (Zhaveh reservoir capacity exceeds 516.2 MCM), water can be transferred to the Karkheh basin gravitationally. In this condition, $C2_{\text{Pumpage}}$, the cost coefficient of the unit water pumped to Karkheh, equals zero and only the costs due to constructing a transfer tunnel are considered. It is assumed that all the cost and benefit parameter values are the ones calculated after considering the time value of money and variable costs include operation and maintanance costs. The PSO–MODSIM model's formulation can therefore be expressed as follows:

$$\begin{aligned}
\text{Minimize Cost} &= f_1(\text{Cap}_{\text{Zhaveh}}, \text{Cap}_{\text{Banidar}}, \text{Cap}_{G-D\,\text{Transfer}}, \\
&\quad \text{Cap}_{\text{Karkheh-Transfer}}, \text{Cap}_{\text{Banidar-Transfer}}) - f_2(Q_{\text{Demands}}) \\
&= M_1 \times \text{CapitalCost}_{\text{Zhaveh-Dam}} + M_2 \\
&\quad \times \text{CapitalCost}_{\text{Banidar-Dam}} + M_3 \\
&\quad \times \text{CapitalCost}_{\text{Ghorve-Pumpage}} + M_4 \\
&\quad \times \text{CapitalCost}_{\text{Karkheh-Transfer}} + M_5 \\
&\quad \times \text{CapitalCost}_{\text{Banidar-Tunnel}} + C_{\text{cap.}} \times (\text{Cap}_{\text{Zhaveh}} \\
&\quad + \text{Cap}_{\text{Banidar}}) + C_{\text{Damage}} \times \text{Cap}_{\text{Zhaveh}} + C1_{\text{Pumpage}} \times \\
&\quad Q_{\text{Ghorveh}} \times H_{\text{Ghorveh}} + M_6 \times C2_{\text{Pumpage}} \times Q_{\text{Karkheh}} \times \\
&\quad H_{\text{Kharkheh}} + M_7 \times C1_{\text{Diameter}} \times Q_{\text{Karkheh}} + C2_{\text{Diameter}} \times \\
&\quad Q_{\text{Banidar}} - \sum_{t=1}^{T} \sum_{j=1}^{\text{NDEM}} C_{\text{Benefit}} \cdot \text{Qdem}_{t,j}
\end{aligned}$$

Subject to:

$\text{Cap}_{\text{min-Zhaveh}} \leq \text{Cap}_{\text{Zhaveh}} \leq \text{Cap}_{\text{max-Zhaveh}}$

$\text{Cap}_{\text{min-Banidar}} \leq \text{Cap}_{\text{Banidar}} \leq \text{Cap}_{\text{max-Banidar}}$

$\text{Priority}_{\text{min}} \leq \text{Priority}_{\text{Res.}(i)} \leq \text{Priority}_{\text{max}, i=1}, \ldots,$ no. of storage nodes in the network

$\text{Qdem}_{t,j} = f_3(\text{Prioriy}_{\text{Res.}(1)}, \ldots, \text{Prioriy}_{\text{Res.}(i)}), i = 1, \ldots,$ no. of storage nodes in the network

$$f_3 = \text{minimize}\left[\sum_{l \in A} c_l q_l\right]_t; A = \{\text{all links in the network}\},$$

q_l = flow rate in link l, c_l = costs, weighting factors or priorities per unit of flow rate in link l, $t = 1, \ldots, T$ (= no. of time steps)

Subject to:

$\left[\sum_{j \in O_i} q_j - \sum_{k \in I_i} q_k\right]_t = 0$; for all $i \in N = \{\text{all nodes}\}$, $O_i = \{$all links originating at node $i\}$, $I_i = \{$all links terminating at node $i\}$, $t = 1, \ldots, T$ $l_l \leq [q_l]_t \leq u_l$; for all $l \in A$, l_l = the lower bound on flow in link l, u_l = the upper bound on flow in link l, $t = 1, \ldots, T$ $c_l = -(50{,}000 - 10 \times \text{Priority}_{\text{Res.}(i)})$; for all $l \in S_i = \{$accounting links originating from storage node $i\}$, $i = 1, \ldots,$no. of storage nodes in the network $c_l = -(50{,}000 - 10 \times \text{Priority}_{\text{Dem.}(i)})$; for all $l \in D_i = \{$accounting links originating from demand node $i\}$, $\text{Priority}_{\text{Dem.}(i)}$ = priority for demand node i, $i = 1, \ldots,$ no. of demand nodes in the network.

The last two equations are used in MODSIM's NFPs to evaluate the costs for accounting active storage and demand links. There are some points to be mentioned about the optimization algorithm used in the PSO–MODSIM model. The problem's decision variables are the capacities of the new reservoirs ($\text{Cap}_{\text{Zhaveh}}$ and $\text{Cap}_{\text{Banidar}}$), as the design variables and the priorities of reservoirs' target storages in the system ($\text{Priority}_{\text{Res.}(i)}$) as the operational variables. The other variables in the objective function, such as the capacities of the transfer systems, are functions of these decision variables. As seen in the procedure described above, the variables of the reservoir capacities are directly used in the PSO objective function; however, the priorities are used in MOSDIM to obtain the water allocations. In each time period of the simulation horizon, an NFP is solved in MODSIM using the Lagrangian relaxation algorithm. The flow diagram of the PSO–MODSIM model is presented in Figure 5. It is worth mentioning that solving such a complex optimization problem using one of the constrained optimization methods is extremely difficult, if not impossible.

THE PSO–MODSIM–ANN MODEL

In the PSO–MODSIOM model, MODSIM is executed *neval* times which is the number of function evaluations equal to the number of particles in each population (swarm size) multiplied by the number of generations required for PSO to converge. On the other hand, accomplishing MODSIM may be time-consuming as an NFP is solved for

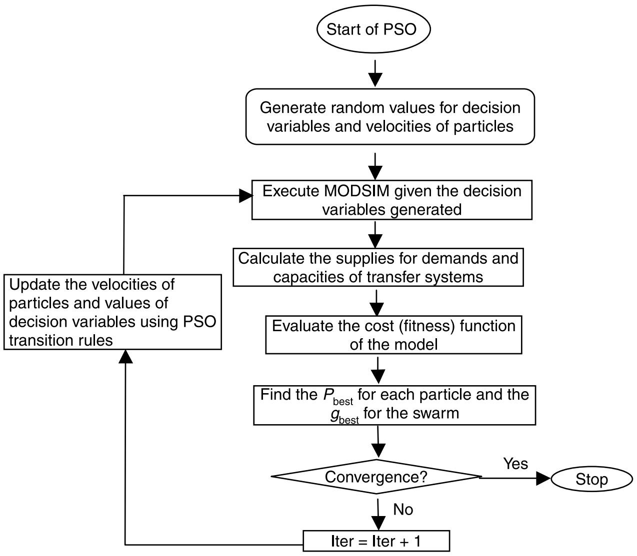

Figure 5 | The flow diagram of the PSO–MODSIM model.

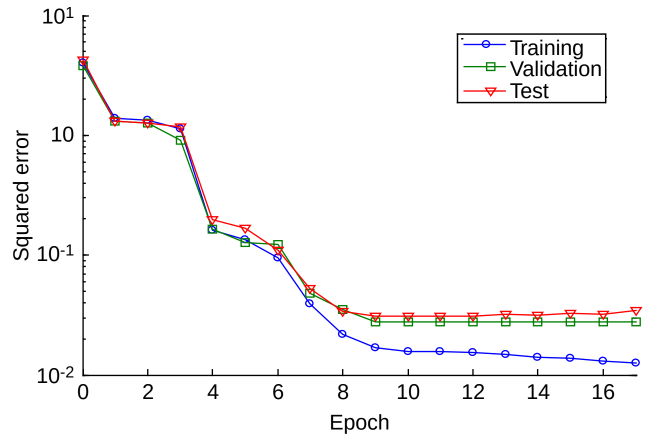

Figure 6 | Variations of the MSE for the training, validation and testing data.

each time period of the planning horizon T. The model's computational difficulty is realized to be more challenging by considering the fact that usually a sensitivity analysis should be carried out for tuning the PSO algorithm parameters, akin to any population-based random search algorithm.

Therefore, a feed-forward neural network is trained off-line and replaces MODSIM in the PSO–MODSIM model. To train this network, a set of input–output data pairs was purposefully prepared. Hence, 1,200 sets of decision variables were randomly generated and for each of them MODSIM, which has been already calibrated for the upstram Sirvan basin, was executed and the objective function was evaluated. Such a data-generation task was performed automatically using MODSIM's custom coding features. The available datasets were then divided into three separate sets as training, validating and testing sets. The ANN model developed has 12 neurons in the input layer equal to the number of input vector elements (the problem's decision variables), one neuron in the output layer equal to the number of output vector elements (objective function) and 10 neurons in the hidden layer selected after some trial and error examinations. The variations of the mean square errors (MSE) of the training, validation and testing datasets over training iterations (epochs) are shown in Figure 6. The average MSE for the training data was 0.002 while it was 0.0078 for the testing data. Figure 7 shows the plot of objective function values predicted by ANN versus its actual values for the testing dataset.

RESULTS

Based on some trial and error work and experiences reported in the literature, the PSO parameters are selected as follows: the swarm size is considered as 20 in the PSO–MODSIM model and 100 in the PSO–MODSIM–ANN model, the maximum and minimum inertia weights are, respectively, considered as $w_{max} = 1.2$, $w_{min} = 0.1$ and acceleration constants as $c_1 = c_2 = 0.5$ in the PSO–MODSIM model and $c_1 = 2.3$ and $c_2 = 1.2$ in the PSO–MODSIM–ANN model. The algorithm stops if there is no

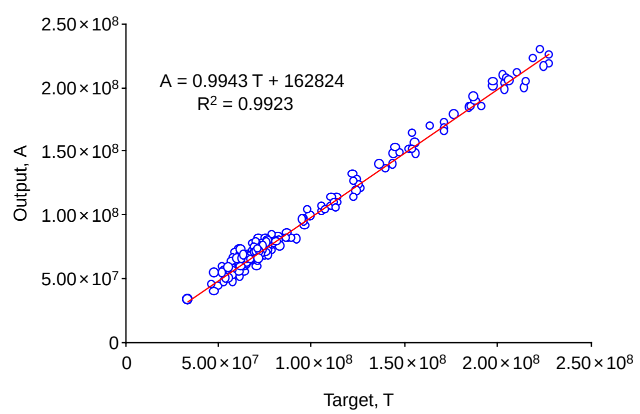

Figure 7 | ANN-based predictions vs. actual values of the objective function for the set of testing data.

improvement in the objective function value over 100 successive iterations. Tables 3 and 4 present the results of the models obtained for design and operational decision variables, respectively.

The minimum total cost obtained by the PSO–MODSIM and PSO–MODSIM–ANN models are −67,095,025 and −70,029,240 units, respectively. In order to see how accurate the approximate model performs, the system performance is simulated by MODSIM under the solution obtained by the PSO–MODSIM–ANN model. The corresponding actual total cost equals −67,211,126 units. Surprisingly, it is seen that the PSO–MODSIM–ANN model is able to arrive at a better solution in comparison with the PSO–MODSIM model in terms of their objective function values. One of the reasons for this may be due to the possibility of testing the approximate model with a larger swarm size (100 particles) because of its computational efficiency whereas the maximum swarm size considered in the PSO–MODSIM was set to 20 particles. Another important point is the task of fine-tuning and performing sensitivity analysis on the PSO parameters which can be done efficiently in the PSO–MODSIM–ANN due to the short time of execution of the model. For example, values of $c_1 = c_2 = 0.5$ used in the PSO–MODSIM model are recommended in the PSO literature while $c_1 = 2.3$ and $c_2 = 1.2$ in the PSO–MODSIM–ANN provided a better solution. This feature might be an important factor in the PSO–MODSIM–ANN model which helps us find a better and more robust solution than that of the PSO–MODSIM model.

It is seen in Table 3 that, due to the significant cost consequences to inundated lands and properties by enlarging the Zhaveh reservoir, a capacity of 83 and 89.6 (lower than 90) MCM were, respectively, obtained by the models.

Table 3 | Optimum capacities of the structural components obtained by the PSO–MODSIM and PSO–MODSIM–ANN models

Capacity		
PSO–MODSIM–ANN	**PSO–MODSIM**	**Structure**
89.6 MCM	82.8 MCM	Zhaveh Dam
348.85 MCM	336.5 MCM	Banidar Dam
9 m^3/s	8 m^3/s	Banidar Tunnel
20 m^3/s	18 m^3/s	Karkheh Tunnel
–	–	Ghorveh Pumping System

Table 4 | Optimum priorities of reservoir storages obtained by the PSO–MODSIM and PSO–MODSIM–ANN models

Priority		
PSO–MODSIM–ANN	**PSO–MODSIM**	**Reservoir**
1	1	Zarivar
7	6	Garan
8	8	Gheshlagh
8	8	Ramasht
1	5	Amirabad
8	8	Soleimanshah
8	8	Gavshan
8	5	Zivieh
8	8	Zhaveh
8	8	Banidar

In order to increase the water supply to the Karkheh basin, both models suggested the construction of the Banidar dam and the tunnel transferring water to the Karkheh basin. Because of the high cost of pumping water to Ghorveh, the amount of water transferred to Ghorveh equals zero in both models. It is also seen in Tables 3 and 4 that the models' results are close to each other in general except for the optimum priority of Amirabad's target storage, which is a small reservoir in the system. Figures 8 and 9 show the variations of the models' objective function over the PSO generations associated with different particles. It is clear in Figure 9 that large oscillations in the PSO–MODSIM solutions are a result of introducing turbulence to the PSO (TPSO).

Table 5 presents the amounts of optimum annual supply and demand values for different sub-basins in the system obtained by the models. A flow of 705–708 MCM is

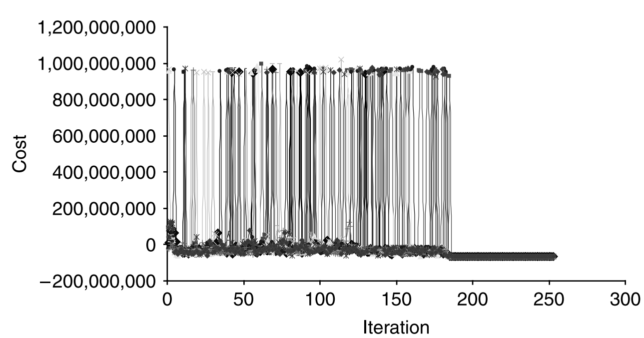

Figure 8 | Variations of the PSO particles' objective function over the iterations in the PSO–MODSIM model.

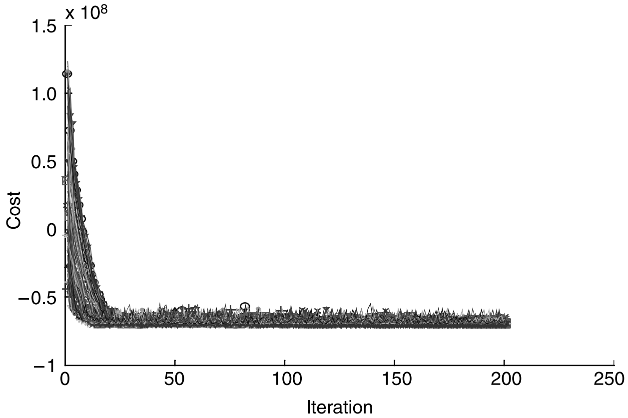

Figure 9 | Variations of the PSO particles' objective function over the iterations in the PSO–MODSIM–ANN model.

drained to the sink node annually. This is because of the high cost of pumping water to Ghorveh as the models find it more beneficial to let the flow pass instead of pumping it to Ghorveh.

An important question which may raise is how significant the role of optimizing the reservoirs' priorities, as operational variables, could be. To answer this, a further investigation was carried out in which the optimization problem was solved with the decision variables of the Zhaveh and Banidar capacities while considering the priorities set to the lower possible value, i.e. 8. This is equivalent to supplying the demands downstream of the reservoirs in each time period as much as possible without performing a kind of hedging strategy dealing with the deficits in future time periods. The PSO–MODSIM model was run to find the optimum values of two decision variables in this situation. The resulted optimum capacities are equal to 90 and 299.3 MCM for the Zhaveh and Banidar dams, respectively. In addition, the minimum total cost of the system becomes −59 646 755 units indicating an 11% excess cost compared to the previous case where operational and design variables are optimized. This result implies that optimizing the reservoirs' priorities yields a significant saving in the system's design and operational cost.

Table 5 | Annual supply and demand values obtained by PSO–MODSIM and PSO–MODSIM–ANN models

Supply (MCM) PSO–MODSIM–ANN	PSO–MODSIM	Demand (MCM)	Item
313	315	379.7	Inside the basin
245.7	245.6	431	Upstream Karkheh basin
–	–	157.7	Ghorveh
595.4	590.6	738.3	Karkheh basin
1154.1	1151.2	1076.7	Sum
705.7	708.2	–	Outflow

SUMMARY AND CONCLUSIONS

In this paper, the general river basin network flow model, MODSIM, as the simulation engine, was embedded in a PSO algorithm as a population-based evolutionary optimization algorithm. The resulting PSO–MODSIM model was used for optimum design and operation of the upstream Sirvan basin in Iran. Noting that this model is computationally intensive, MODSIM was approximated by a trained multilayer feed-forward neural network as a meta-modeling tool. The models were successfully applied to the upstream Sirvan basin system in Iran considering the issues of water transfer to the neighboring basins. It has been shown that the resulted PSO–MODSIM–ANN model is much faster than the original PSO–MODSIM model. The short time of the objective function evaluations in the PSO–MODSIM–ANN model makes it possible to run it with a larger swarm size which increases the exploration capability of the search algorithm. Also, the sensitivity analysis, and thus fine tuning, of the model parameters can be done more easily. As a result, the PSO–MODSIM–ANN model has been able to find a better solution compared to the one obtained by the PSO–MODSIM model.

To be fair about the two modeling tools developed, it should be noted that, although the PSO–MODSIM–ANN is much faster than the PSO–MODSIM model, it takes time to generate the input–output dataset needed for training the network. This needs us to execute MODSIM several times each providing a pair of input–output data and a sufficient number of exemplars is required to be used in the network training. However, the methodology is still useful and significant as the off-line training process is performed only one time. The purpose and scope of this study is to present the possibility of solving a complex and real-world river basin optimization problem using meta-modeling. However, as pointed out by one of the reviewers and presented in the literature, the problem can be solved more efficiently by nesting the training of ANNs within the evolutionary loop and perform incremental training implemented within the backpropagation algorithm. It is also possible to use smaller training sets and retrain the network as the solutions converge toward the optimal set. Making use of adaptive and dynamic meta-modeling techniques is the subject of our ongoing research work in a complementary study. It is believed that the models represent useful tools for strategic decision-making regarding water transfer and future water resource developments at basin scales.

ACKNOWLEDGEMENTS

The authors would like to thank Dr Enrique Triana and Professor John W. Labadie for letting us use MODSIM 8.0 and their help in this regard. We are also indebted to two reviewers of the paper for their valuable suggestions and comments.

REFERENCES

Alley, W. M. 1986 Regression approximations for transport model constraint sets in combined aquifer simulation-optimization studies. *Water Resour. Res.* **22** (4), 581–586.

Broad, D., Dandy, G. C. & Maier, H. R. 2005 Water distribution systems optimization using metamodels. *Water Res. Plann. Manage. ASCE* **131** (3), 172–180.

Cai, X., McKinney, D. C. & Lasdon, L. S. 2002 A framework for sustainability analysis in water resources management and application to the Syr Darya Basin. *Water Resour. Res.* **38**(6), 1085.

Cai, X. M., McKinney, D. C. & Lasdon, L. S. 2003 Integrated hydrologic-agronomic-economic model for river basin management. *Water Res. Plann. Manage. ASCE* **129** (1), 4–17.

Dougherty, D. E. 1991 Hydrologic applications of the connection machine CM-2. *Water Resour. Res.* **27** (12), 3137–3147.

Eberhart, R. C. & Kennedy, J. 1995 A new optimizer using particle swarm theory. In *Proceedings of Sixth Symposium on Micro Machine and Human Science*, pp. 39–43. IEEE Service Center. Piscataway, NJ.

Fredericks, J., Labadie, J. W. & Altenhofen, J. 1998 Decision support system for conjunctive stream-aquifer management. *Water Res. Plann. Manage. ASCE* **124** (2), 69–78.

Hsu, N. -S. & Cheng, K. -W. 2002 Network flow optimization model for basin-scale water supply planning. *Water Res. Plann. Manag. ASCE* **128** (2), 102–112.

Jenkins, M. W., Lund, J. R., Howitt, R. E., Draper, A. J., Msangi, S. M., Tanaka, S. K., Ritzema, R. S. & Marques, G. F. 2004 Optimization of California's water system: results and insights. *Water Res. Plann. Manage. ASCE* **130** (4), 271–280.

Johnson, V. M. & Rogers, L. L. 2000 Accuracy of neural network approximators in simulation-optimization. *Water Res. Plan. Manag. ASCE* **126** (2), 48–56.

Karatzas, G. K. 1997 An effective procedure to incorporate uncertainty into subsurface remediation models using a multi-scenario approach. In *Proceedings of 4th SIAM Conf. on Mathematical and Computational Issues in the Geosciences*, Society for Industrial and Applied Mathematics. Albuquerque, NM. p. 58.

Karatzas, G. K. & Pinder, G. F. 1993 Groundwater management using numerical simulation and the outer approximation method for global optimization. *Water Resour. Res.* **29** (10), 3371–3378.

Kennedy, J. & Eberhart, R. C. 1995 Proceedings IEEE International Conference on Neural Networks, IV. In *Particle swarm optimization*, IEEE Service Center, Piscataway, NJ, pp. 1942–1948.

Kuczera, G. 1993 Network linear programming codes for water supply headworks modeling. *Water Res. Plann. Manag. ASCE* **119** (3), 412–417.

Kuczera, G. & Diment, G. 1988 General water supply simulation model: WASP. *Water Res. Plann. Manage. ASCE* **114** (3), 365–382.

Labadie, J. W. 1995 . *MODSIM: River Basin Network Flow Model for Conjunctive Stream-aquifer Management. Program User Manual and Documentation*. Colorado State University, Boulder, CO.

Lefkoff, L. J. & Gorelick, S. M. 1990 Simulating physical processes and economic behavior in saline, irrigated agriculture: model development. *Water Resour. Res.* **26** (7), 1359–1369.

Liu, H. & Abraham, A. 2001 Fuzzy adaptive turbulent particle swarm optimization. In *Proceedings of IEEE International Conference on Evolutionary Computation*, IEEE Service Center, Piscataway, NJ. pp. 101–106.

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

McBride, R. D. 1985 Solving embedded generalized network problem. *Eur. J. Oper. Res.* **21**, 82–92.

McKinney, D. C., Cai, C., Rosegrant, M. W., Ringler, C. & Scott, C. A. 1999 *Integrated Basin-scale Water Resources Management Modeling: Review and Future Directions*. IWMI SWIM Paper No. 6. International Water Management Institute, Colombo, Sri Lanka.

McKinney, D. C. & Lin, M. D. 1994 Genetic algorithm solution of groundwater management models. *Water Resour. Res.* **30** (6), 1897–1906.

Mousavi, S. J., Ponnambalam, K. & Karray, F. 2007 Inferring operating rules for reservoir operation using fuzzy regression and ANFIS. *Fuzzy Sets Syst.* **158**, 1064–1082.

Ponnambalam, K., Karray, F. & Mousavi, S. J. 2003 Minimizing variance of reservoir systems operations benefits using soft computing tools. *Fuzzy Sets Syst.* **139** (2), 451–461.

Raman, H. & Chandramouli, V. 1996 Deriving a general operating policy for reservoirs using neural networks. *Water Res. Plann. Manage. ASCE* **122** (5), 342–347.

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

Rogers, L. L. & Dowla, F. U. 1994 Optimization of groundwater remediation using artificial neural networks with parallel solute transport modeling. *Water Resour. Res.* **30** (2), 457–481.

Shi, Y. & Eberhart, R. C. 1998*a* A modified particle swarm optimizer. In *Proceedings of the 1998 IEEE Conference on Evolutionary Computation*, IEEE Service Center, Anchorage, AK. pp. 69–73.

Shi, Y. & Eberhart, R. C. 1998*b* Parameter selection in particle swarm optimization. In *Evolutionary Programming VII* (ed. V. W. Porto, N. Saravanan, D. Waagen & A. E. Eiben), Springer, Berlin. pp. 611–616.

Smith, J. & Eli, R. N. 1995 Neural network models for rainfall-runoff process. *Water Res. Plann. Manage. ASCE* **121** (6), 499–508.

Sun, Y-H., Yeh, W. W-G., Hsu, N-S. & Louie, P. W-F. 1995 Generalized network algorithm for water supply optimization. *Water Res. Plann. Manage. ASCE* **121** (5), 392–398.

Sureerattanan, S. & Phein, H. N. 1997 Backpropagation networks for daily streamflow forecasting. *Water Resour. Res.* **C/195**, 1–7.

Tompson, A. F. B., Ashby, S. F., Falgout, R. D., Smith, S. G., Fogwell, T. W. & Loosmore, G. A. 1994 Use of high performance computing to examine the effectiveness of aquifer remediation. In *Computational Methods in Water Resources X* (ed. A. Peters), Vol. 2, Kluwer, Dordrecht. pp. 875–882.

Wagner, B. J. 1995*a* Recent advances in simulation-optimization groundwater management modeling. US National Rep. Prepared for International Union of Geodesy and Geophysics, 1991–1994. *Rev. Geophys.* **33** (Suppl.), 1021–1028.

Wagner, B. J. 1995*b* Sampling design methods for groundwater modeling under uncertainty. *Water Resour. Res.* **31** (10), 2581–2591.

Yan, S. & Minsker, B. 2006 Optimal groundwater remediation design using an adaptive neural network genetic algorithm. *Water Resour. Res.* **42**, 1–14. W05407, doi:10.1029.

First received 8 November 2007; accepted in revised form 22 April 2008