# PUMPING OPTIMIZATION OF COASTAL AQUIFERS USING 3-D DENSITY MODELS AND APPROXIMATIONS WITH NEURAL NETWORKS

G. KOURAKOS AND A. MANTOGLOU

*Department of Rural and Surveying Engineering, Technical University of Athens, 9, Iroon Polytechniou Str., Athens 15 780 Greece*

## ABSTRACT

A method for pumping optimization of coastal aquifers based on a three-dimensional density model is developed. The objective is to maximize the total pumping rates subject to constraints that protect the aquifer from saltwater intrusion. The numerical code SEAWAT was utilized to evaluate the distribution of salts in the brackish zone for various pumping scenarios. Since pumping optimization using SEAWAT is computationally time intensive, a model approximation based on neural networks was developed. Then the optimization utilized the simpler models based on neural network approximations rather than complex models based on numerical solution of the governing partial differential equations of flow and transport. The formulation of the constraints is based on the neural network model approximations. The results are compared in a test problem to the "exact" results based on the governing partial differential equations and a good correlation was found.

## 1. INTRODUCTION

One of the most difficult problems in groundwater hydrology is management of coastal aquifers where saltwater intrusion and degradation of freshwater can be caused by extensive groundwater extractions. This problem has been investigated using three-dimensional, variable-density, groundwater flow models considering advection and dispersion of salts. A number of numerical codes such as SEAWAT, FEFLOW, SUTRA, etc [*Smith,* 2004; *Bakker et al.,* 2004 etc.] exist which were tested on several benchmark problems, however, they have not been integrated in a practical way in optimization routines since they require exceptionally large CPU times. Various modelling approximations such as the sharp interface approximation [*Park and Aral*, 2004; *Mantoglou,* 2003; *Mantoglou et al.*, 2004] have been proposed to reduce model complexity and CPU time requirements.

Here, approximations of complex variable-density models based on neural networks are developed. Artificial Neural Networks (ANNs) are parallel distributed processors made up of simple processing units, which have a natural characteristic for storing experiential knowledge and making it available for use [*Haykin*, 1999]. ANNs have gained wide acceptance in many scientific fields owning to their simple structure and ability to approximate even highly non linear and discontinuous systems with good generalization capabilities.

ANNs have been extensively applied in several studies in the field of groundwater hydrology [*Singh et al.*, 2004; *Coppola et al.*, 2003]. *Morshed and Kaluarachchi*, [1998] utilized ANNs to simulate the input-output response of an unsaturated zone flow and contaminant transport system. *Arndt et al*. [2003] employed ANNs to substitute the FEFLOW code, where data input were the pumping rates of 5 wells and output data were the groundwater levels in four observation points. *Rao et al. ,*2004 and *Rao et al.,* 2005 proposed to use ANNs as approximations of complex seawater intrusion models.

In this paper, ANNs are utilized in an optimization model for controlling seawater intrusion in coastal aquifers with non-uniform hydraulic conductivities. After the ANN has been trained, tested and validated it is integrated into an optimization procedure based on Sequential Quadratic Programming (SQP). The optimum solutions obtained by SQP algorithm are simulated with the numerical model and the comparisons indicate a very good match.

## 2. GOVERNING EQUATIONS OF FLOW AND TRANSPORT

The following governing equations describe the intrusion of seawater in coastal aquifers:

i) The equation of mass balance for water:

$$\frac{\partial(n\rho)}{\partial t} = -\nabla \cdot (\rho \mathbf{q}) + \bar{\rho} q_s \qquad [1]$$

where $\rho$ is the variable fluid density, $\mathbf{q}$ is the specific discharge vector, $\bar{\rho}$ is the density of the water entering from a source or leaving through a sink, $q_s$ is the volumetric flow rate per unit volume of aquifer representing sources and sinks, $n$ is them aquifer porosity, and $t$ is time.

ii) Equation of the solute transport:

$$\frac{\partial C}{\partial t} = \nabla \cdot (\mathbf{D} \cdot \nabla C) - \nabla(\mathbf{v} C) - \frac{q_s}{n} C_s \qquad [2]$$

where $\mathbf{D}$ is the hydrodynamic dispersion tensor, $\mathbf{v}$ is the fluid velocity vector and $C_s$ is the solute concentration of water from sources or sinks.

iii) The Darcy law for variable-density groundwater flow:

$$q_x = -\frac{k_x}{\mu}\frac{\partial p}{\partial x}, \quad q_x = -\frac{k_y}{\mu}\frac{\partial p}{\partial y}, \quad q_z = -\frac{k_z}{\mu}\left(\frac{\partial p}{\partial z} + \rho g\right) \qquad [3]$$

where $p$ is the fluid pressure, $q_x, q_y, q_z$ are the individual components of specific discharge, $\mu$ is the dynamic viscosity, $k_x, k_y, k_z$ are the intrinsic permeabilities along the three principal components $x, y, z$ and $g$ is the gravity constant.

iv) Equation relating the fluid density to concentration:

$$\rho = \rho_o (1 + \beta_c c) \qquad [4]$$

where $\rho_o$ is the density of freshwater, and $\beta_c$ is a constant given by the expression $\beta_c = 1/\rho_o \cdot \partial\rho/\partial c$, $c$ is the normalized salt concentration $c \in [0,1]$.

These equations can be solved with various numerical methods such as finite difference or finite element methods. In the present paper the SEAWAT code, based on a finite different model, was employed to simulate the movement of groundwater and solute transport. SEAWAT code combines modified versions of MODFLOW and MT3D packages into a single program that solves the flow and solute-transport equations through a synchronous time-stepping approach which cycles between MODFLOW solutions of the flow equation and MT3D solutions of the transport equation [*Guo & Langevin,* 2002].

## 3. ARTIFICIAL NEURAL NETWORKS

Generally, ANNs are mathematical constructs which have similarities to the operation of the human brain (nonlinear and parallel system, which has the ability to build up its own rules through what is commonly known as experience, *Haykin,* 1999). The fundamental element of a human brain and is the neuron. When a neuron receives information –input- perceives it, and according to the experience (training) makes an appropriate decision (output). ANN use a mathematical equivalent of this element which is outlined in  where a neuron receives several inputs and returns an output through an activation function.

ANNs are organized in the form of layers. The first layer receives input from the external environment and is known as the input layer. Depending on the topology of the network the input layer can be followed either by one or more hidden layers, or by an output layer that returns the results back to the external environment. The most common types of neural networks are the ones where information flows in one direction from the input layer to the hidden layers to the output layer and are called feed-forward networks while the corresponding connections are called feed-forward connections [*Mantoglou & Kourakos*, 2002].

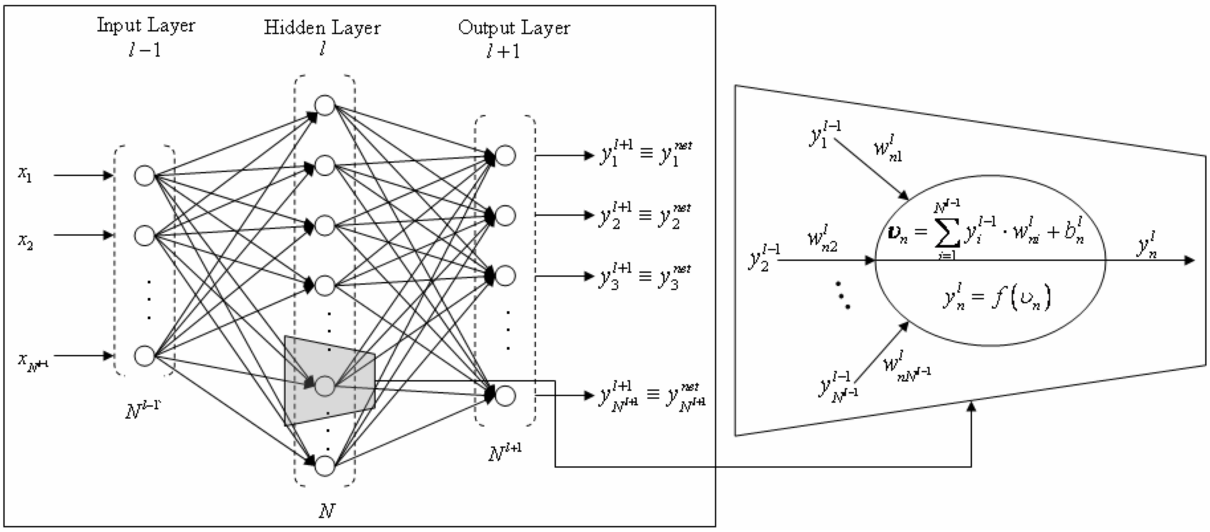


Fig. 1. Artificial Neural Network with one hidden layer

Figure 1 illustrates a feed-forward network with one hidden layer. Lets consider the neuron $n$ of the $l^{\text{th}}$ layer receiving $N^{l-1}$ inputs form the previous layer $l-1$. The total input to the neuron $n$ is expressed by the following sum

$$\upsilon_n = \sum_{i=1}^{N^{l-1}} y_i^{l-1} \cdot w_{ni}^l + b_n^l \qquad [5]$$

Where $y_i^{l-1}$ is the output of the $i$ neuron of layer $l-1$, $i \in \left[1, N^{l-1}\right]$, $N^{l-1}$ is the number of neurons of the previous layer, $w_{ni}^l$ is the synaptic weight between the neuron $n$ of the $l$ layer and the neuron $i$ of the $l-1$ layer, and $b_n^l$ is a scalar bias that corresponds to neuron $n$ of layer $l$. Next sum $\upsilon_n$ is transformed through an activation-transfer function into neural output:

$$y_n^l = f\left(\upsilon_n\right) \qquad [6]$$

where $y_n^l$ is the output of neuron $n$ of layer $l$ and $f$ is the activation function.

There is a variety of activation functions; three of them being the most popular: i) linear, ii) sigmoid, iii) hyperbolic tangent functions, as follows:

$$i)\, y(x) = x, \qquad ii)\, y(x) = \frac{1}{1+e^{-ax}}, \qquad iii)\, y(x) = \tanh(x) \qquad [7]$$

The feed forward ANNs are able to approximate quite complex systems when they are properly trained in a supervised manner. According to supervised training, a sufficient sample of the input - output response is used for training. The most known training algorithm is the back-propagation, which is separated into two steps. In the first step the inputs are used and the ANN returns an output. During this step the synaptic weights are fixed. In the second step the weights are adjusted according to the error signal, which is the diversion between the responses of the network from the actual (desired) responses. The objective of the back propagation algorithm is to adjust the synaptic weights and biases so as to minimize the an error function such as the mean squared error

$$E = \sum_{i=1}^{S} \sum_{j=1}^{N^{NET}} \left( y_{j,i}^{net} - d_{j,i} \right)^2 \qquad [8]$$

Where $y_{j,i}^{net}$ is the $j$ network output that corresponds to input pattern $i$, $d_{j,i}$ is the desired output $j$ that corresponds to input pattern $i$, $j \in \left[1, N^{NET}\right]$, $N^{NET}$ is the number of outputs, and $S$ is the number of sample patterns.

Once an ANN is trained it is very quick to run and requires very short CPU times. The basic idea of this paper is to use an ANN to approximate the complex behaviour of the real aquifer. Since there are no comprehensive data on real aquifers, the ANN was trained using the model output generated by the complex and CPU time consuming SEAWAT on various stressing conditions, After the ANN has been trained, tested and validated it is integrated into an optimization procedure based on Sequential Quadratic Programming (SQP).

## 4. OPTIMIZATION MODEL

The optimization model of the coastal aquifer involves maximization of pumping rates of several wells, subject to a set of constraints that limit the concentration at various observation points below a specified concentration value. The above statement can be expressed in mathematical form as:

$$\max\left(\sum_{i=1}^{N} Q_i\right) \qquad [9]$$

Subject to the following set of constrains:

$$\begin{aligned} c_j &< c_j^{\max}; \qquad j = 1, \cdots M \\ Q_{\min} \le Q_i &\le Q_{\max}; \quad i = 1, \cdots N \end{aligned} \qquad [10]$$

where $Q_i$ is the pumping rate of the $i^{\text{th}}$ well, $i \in [1, N]$, $N$ is the number of pumping wells, $c_j$ is the concentration at specified observation point $j \in [1, M]$, and $Q_{\min}, Q_{\max}$ are the minimum and maximum allowable pumping rates. The $j$ observation points can be the same with the well location points or may be any other set of points of the aquifer, depending on the nature and the objectives of the problem.

The concentration $c_j$ in [10] depends on the decision variables $Q_i$ and is calculated using the fast ANN model approximation rather than CPU time consuming SEAWAT code. The optimization procedure is based on Sequential Quadratic Programming (SQP) and was coded in MATLAB, similarly to *Mantoglou et al.*, 2004.

# 5. APPLICATION

The proposed methodology was tested in a hypothetical orthogonal aquifer with impermeable north and south boundaries and a sea boundary at the east side, while there was a constant flux $q_s = 18.4\, m^3 / day$ at the west boundary. Hydraulic conductivity was assumed non-uniform as illustrated by Fig. 2. Precipitation was assumed uniform and equal to $N = 0.0004\, m / day$ while the longitudinal and transverse dispersivities are $\alpha_L = 20\, m$ and $\alpha_T = 5\, m$ respectively.

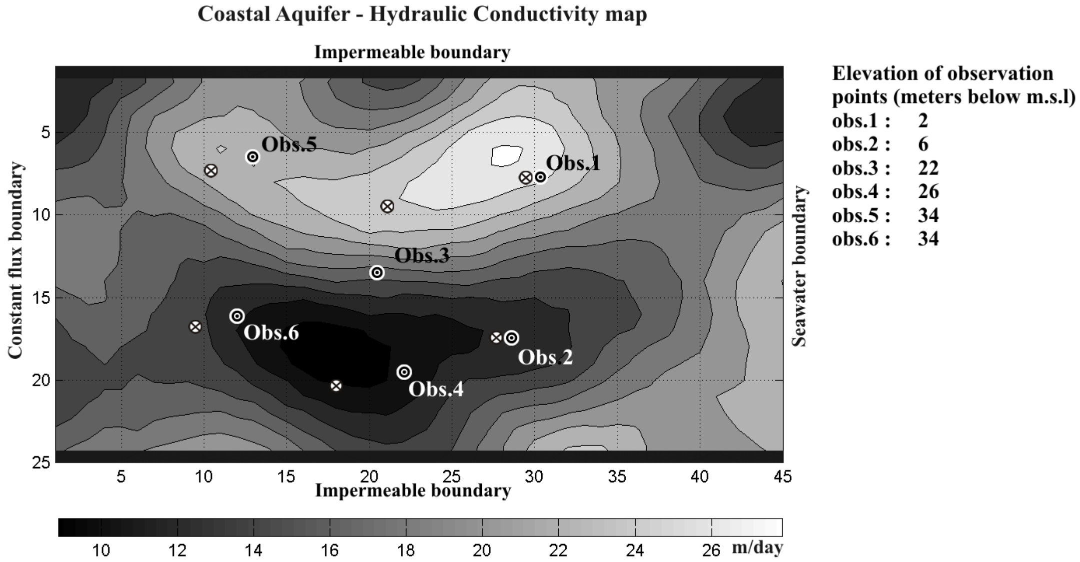


Fig. 2. Coastal Aquifer

An important issue in ANN modelling is the selection of an appropriate structure which involves the choice of the number of hidden neurons as well as the choice of the activation function for each neuron. It has been established by experience [*Haykin*, 1999] that it is a good practice each layer to have a specific type of transfer function. Neural networks with one hidden layer were examined in this work, and it was found that the best generalization capabilities were obtained with hyperbolic tangent hidden layers and linear output layers.

The appropriate number of hidden neurons was investigated next. A training sample was created using the SEAWAT code (45x20=900 cells and 10 layers), where the inputs are the pumping rates of six wells (see Fig 2), and the outputs are the concentration values of six pre-selected observation points after a simulation time of 3000 days. The observation points are also shown in Fig. 2 and are located at various depths. In order to perform a comprehensive analysis, the concentration at all discretization points of the aquifer was saved for each input pattern. The total number of input patterns is $S = 600$, while half were used for training and half for validation. Numerous trials were carried out in an attempt to identify the impact of the number of hidden neurons. Since the initial weights affect the final performance of the ANN the trials for each combination of neurons were repeated 20 times using random initial weight and biases values. Fig. 3 plots the number of hidden neuron vs. correlation coefficient for the 6 outputs for the training and validation set respectively. It can be seen that a number of hidden neurons over 7 gives very good performance while the performance is not practically improved when the ANN has more than 10 neurons. Hence an ANN with 6 inputs, 10 hidden and 6 output neurons was considered adequate for this application, (in ANN terminology this is a 6-10-6 network).

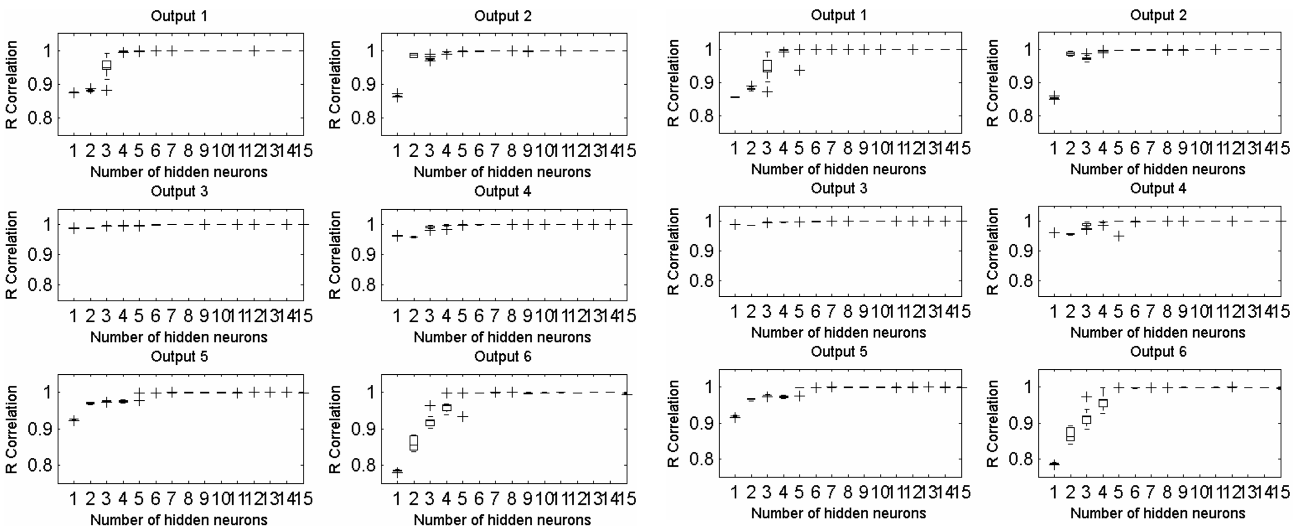


Fig. 3. Performance of ANN vs. number of hidden neuron for training and testing set

Using the training set of 300 input patterns a 6-10-6 ANN was trained using the back propagation algorithm. Figure 4 illustrates the correlation between the concentrations calculated with SEAWAT and those calculated with the ANN for the train and test set respectively. It can be seen that the predictions obtained by ANN are very accurate, thus this network was used to substitute SEAWAT in the optimization procedure below.

The generalized optimization model described in previous section was applied to the hypothetical aquifer of Fig. 2, where the objective is to maximize the pumping rates of the six production wells specified in the figure, while the concentration at the six observation points has to be maintained below a critical concentration value $c_j \le c^{\max} = 2kg/m^3,\ \ j \in [1 \ldots 6]$.

After optimization, the optimum pumping rates obtained using the ANN model in the optimization procedure, were inserted into SEAWAT in order to check and if the constrain set is satisfied as well as the generalization ability of ANN specific to the optimization problem. T compares the predictions of SEAWAT to the predictions of ANN. The dotted line is a 45$^{o}$ degree line indicating a very good match between the two sets. Hence the neural network has the ability to predict new data besides the ones used for training. The figure also shows that the constraints $c_j \le c^{\max} = 2kg/m^3$ are also satisfied when running the “exact” SEAWAT model using the optimal pumping rates obtained by the simplified ANN model.

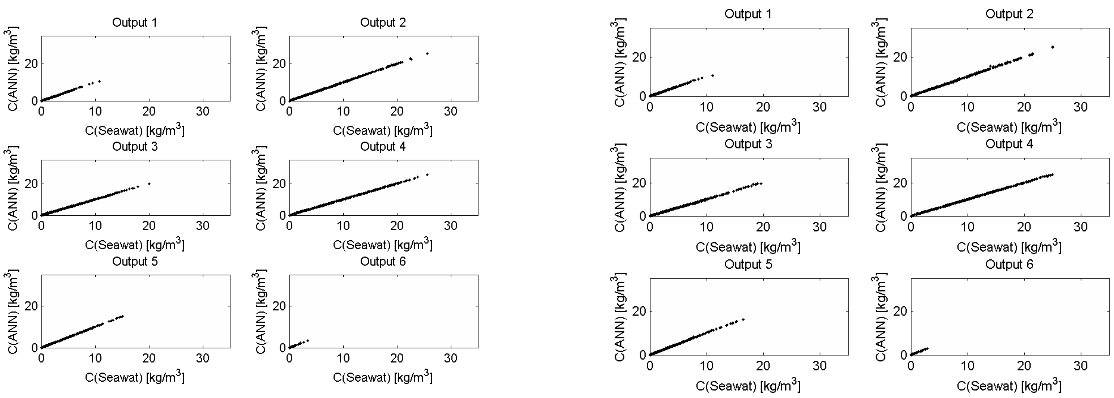


Fig. 4. SEAWAT vs. ANN simulated concentrations: training (left) and testing set (right)

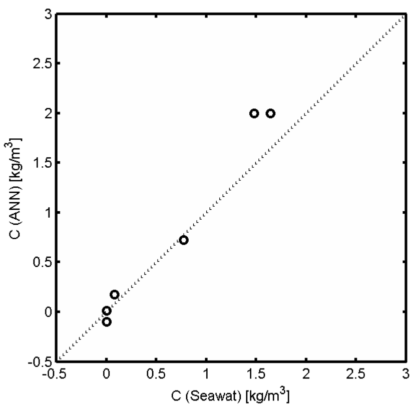


Fig. 5. SEAWAT and ANN simulated concentrations for the six reference points using ANN optimized pumping rates

# 6. CONCLUSIONS

A methodology was presented for optimal management of coastal aquifers with nonuniform hydraulic conductivity. The objective is the maximization of pumping rates of several production wells, while protecting the aquifer from seawater intrusion and degradation of freshwater. The basic idea is to use an ANN to approximate the complex behaviour of the real aquifer. Since there are no comprehensive data on real aquifers, the ANN was trained using the model output generated by the complex and CPU time consuming SEAWAT on various stressing conditions, After the ANN has been trained, tested and validated it is integrated into an optimization procedure based on Sequential Quadratic Programming (SQP) and was coded in MATLAB.

The generalization characteristics of ANNs were investigated in the application and revealed that for a network with six inputs and six outputs the number of hidden neurons must to be of the order of 10. Since the number of hidden neurons is crucial for good performance of ANNs, it is strongly suggested that a similar analysis must always be performed.

An important consideration is the size of the training sample. Although it may look meaningless to create a sample of over 500 input-output patterns, since an average SQP optimization with 6 decision variables requires less than 400 function evaluations, it is recommended to save the concentrations for all input patterns and all discretization cells in a database. Hence using the same training set, it is possible to deal with different constraints. Since SQP optimization is influenced by the initial values of the decision variables, it may be necessary to perform the optimization many times with different initial values in order to obtain a solution close to the global optimum.

The optimum solution obtained by SQP optimization was used in the numerical model and the results were in agreement with the results from the ANN simulator. The results show that it is possible to train an ANN with good generalization capabilities and use it in place of the numerical model.

ACKNOWLEDGMENT This work was supported by a grant by the General Secretariat for Research & Technology of Greece.

# REFERENCES

Arndt, O., T. Barth, B. Freisleben, and M. Grauer (2005), Approximating a finite element model by neural network prediction for facility optimization in groundwater engineering, *European Journal of Operational Research, 166(3)*, 769-781

Bakker, M., G. H. P. O. Essink, and C. D. Langevin (2004), The rotating movement of three immiscible fluids-a benchmark problem, *Journal of Hydrology 287,* 270-278

Coppola, E. Jr., Szidarovszky, F., M. Poulton, and E. Charles, (2003), Artificial Neural Network Approach for Predicting Transient Water Levels in a Multilayered Groundwater System under Variable State, Pumping, and Climate Conditions, *Journal of Hydrologic Engineering 8(6),* 348-360.

Guo, W., and C. D. Langevin (2002), User guide to SEAWAT: A Computer Program for Simulation of Three-Dimensional Variable-Density Ground-Water Flow,*U.S.G.S.,* Open-File Report 01-434

Haykin, S., (1999), *Neural Network a comprehensive foundation,* 2nd ed., 842 pp., Prentice-Hall,Inc., New Jersey.

Mantoglou A., (2003), Pumping management of coastal aquifers using analytical models of saltwater intrusion, *Water Resources Research 39(12),* 1379-1397

Mantoglou, A., and G. Kourakos (2002), Modeling the Rainfall-Runoff Process using Neural Networks with Emphasis on Optimization of Network Configuration using Genetic Algorithms, 5th International Conference "Water Resources Management in the Era of Transition", European Water Resources Association, Athens, 4-8 September

Mantoglou, A., M. Papantoniou, and P. Giannoulopoulos (2004), Management of coastal aquifers based on nonlinear optimization and evolutionary algorithms, *Journal of Hydrology 297,* 209-228.

Morshed, J., and J. J. Kaluarachchi (1998), Application of artificial neural network and genetic algorithm in flow and transport simulations, *Advances in Water Res., 22(2),* 145-158.

Park, C.-H., and M. M. Aral (2004), Multi-objective optimization of pumping rates and well placement in coastal aquifers, *Journal of Hydrology 290,* 80-99

Rao, S. V. N., V. Sreenivasulu, S. M. Bhallarmudi, B. S. Thandaveswara, and K. P. Sudheer (2004), Planning groundwater development in coastal aquifers, *Hydrological Sciences – Journal – des Sciences Hydrologiques, 49(1),* 155-170

Rao, S. V. N., S. M. Bhallamudi, B. S. Thandaveswara, and V. Sreenivasulu (2005), Planning Groundwater Development in Coastal Deltas with Paleo Channels, *Water Resources Management 19(5),* 625-639

Singh, R. M., B. Datta, and A. Jain (2004), Identification of Unknown Groundwater Pollution Sources Using Artificial Neural Networks, *Journal of Water Resources Planning and Management, 130(6),* 506-514

Smith, A. J., (2004), Mixed convection and density-dependent seawater circulation in coastal aquifers, *Water Resources Research 40(8).*