WATER RESOURCES RESEARCH, VOL. 34, NO. 5, PAGES 1101–1113, MAY 1998

Parameter estimation using artificial neural network and genetic algorithm for free-product migration and recovery

Jahangir Morshed and Jagath J. Kaluarachchi[1]

Department of Civil and Environmental Engineering and Utah Water Research Laboratory, Utah State University, Logan

Abstract. Artificial neural network (ANN) is considered to be a universal function approximator, and genetic algorithm (GA) is considered to be a robust optimization technique. As such, ANN regression analysis and ANN-GA optimization techniques can be used to perform inverse groundwater modeling for parameter estimation. In this manuscript the applicability of these two techniques in solving an inverse problem related to a light-hydrocarbon-contaminated site is assessed. The critical parameters to be evaluated are grain-size distribution index α and saturated hydraulic conductivity of water K_{sw}, since these parameters control free-product volume predictions and flow. A set of published data corresponding to a light-hydrocarbon-contaminated unconfined aquifer was used as the base case to determine the applicability of these methods under a variety of scenarios. Using limited monitoring- and recovery-well data under homogeneous and heterogeneous conditions, the critical parameters were evaluated. The results were used to determine the relative effectiveness of each method and corresponding limitations. The results of the work suggested that ANN regression analysis has limited utility, especially with heterogeneous soils, whereas the ANN-GA optimization can provide superior results with better computational efficiency. Finally, a general guideline for solving inverse problems using the two techniques is outlined.

1. Introduction

Hydrocarbon-contaminated sites are common occurrences in the United States. When a light-hydrocarbon (lighter than water) spill or a leakage occurs, the free-product, commonly known as "oil," migrates toward the groundwater table, accumulates on the table, spreads over the water table, and flows along the natural hydraulic gradient until saturation and conductivity become small. Usually, the bulk of the free-product volume may remain in the capillary zone waiting for detection and remediation. Typically, free-product recovery involves simultaneous pumping of oil and groundwater from a single recovery well using dual pumps [*Blake and Lewis*, 1982; *Kaluarachchi et al.*, 1990]. Often, the design is developed using appropriate multiphase flow models, and the reliability of the design is dependent on the model parameters. In general, a groundwater model is considered to be a distributed parameter model, since the primary variables, such as fluid heads and contaminant concentrations, are governed by partial differential equations with spatially distributed parameters such as grain-size distribution data and saturated hydraulic conductivity. Often, a reliable groundwater model application is dependent on the availability of important site-specific data. In most cases it is difficult to estimate the complete set of input data through field investigations. Usually, site investigations are limited to a selected number of monitoring wells and field tests, owing to excessive costs. Therefore it is common practice to obtain part of the required input data from historical observations using parameter estimation tools such as inverse modeling. In solving an inverse problem, unknown parameters are estimated from observed data such as water table elevation, concentration, or recovery data recorded in a space-time domain.

Willis and Yeh [1987] provided a general discussion on inverse groundwater modeling. Theoretically, a distributed parameter estimation over the domain using point observations in the space-time domain is desired. However, in practice, this theoretical desire is infeasible. Often, the inverse problem is ill-posed, and the solutions are plagued with instability and nonuniqueness. Instability is caused by noise where the parameters exhibit high sensitivity to the observed data, and the inverse problem can produce inaccurate solutions for small errors in the observed data [*Emsellem and de Marsily*, 1971]. Nonuniqueness is caused by point observations where distributed parameter estimates are attempted from point observations, and the inverse problem can produce multiple solutions satisfying the point observations [*Chavant*, 1974]. Since the inverse problem has these two inherent limitations, the acceptable solutions may only be determined by minimizing the errors, and therefore the true solution may never be guaranteed.

A variety of techniques were developed to solve the inverse groundwater problem over the past few decades. *Neuman* [1973] classified these techniques into two groups: "direct" and "indirect" methods. *Yeh* [1986] described these methods in detail and summarized the applications of various techniques within each method. The direct method treats the parameters as dependent variables in a formal inverse problem. If distributed observations and their derivatives are known over the entire space-time domain, and if the distributed observation

[1]Currently at Royal Institute of Technology, Stockholm, Sweden.

Paper number 98WR00006.
0043-1397/98/98WR-00006$09.00

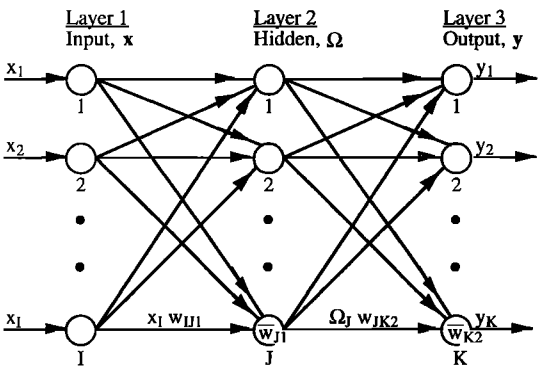

Figure 1. Two-layer feed-forward artificial neural network.

and space-time discretization errors are negligible, this method may be applied; otherwise, its application may lead to instability. In contrast, the indirect method treats the parameters as decision variables in an optimization model; the objective is to minimize an error criterion, a "norm" of the difference between estimated and observed data at limited observation points, subjected to constraints involving governing equations and possible parameter bounds. If point observations are available, this method may be applied, and differentiation of the observed data is not required. However, the optimization model is nonlinear, and it is nonconvex in the presence of noise. As such, traditional gradient-based algorithms can only guarantee local-optimal solutions and can never guarantee the global-optimal solution. Recently, research in groundwater optimization suggested that artificial neural network (ANN) and genetic algorithm (GA) were a robust function approximator and a robust optimization technique, respectively [*Rogers and Dowla*, 1994]. ANN-GA optimization is anticipated to yield faster near-optimal (near-global-optimal) solutions in real time without becoming trapped to local-optimal solutions. Therefore ANN-GA may be used to solve inverse problems formulated as an optimization model using the indirect method. Recently, *Sawyer et al.* [1995] solved an inverse groundwater problem using ANN. An ANN was trained using example groundwater simulations to estimate the unknown saturated hydraulic conductivity from known water head observations. In effect, ANN was used as a regression analysis to approximate an inverse observation-parameter function through which the unknown parameters may be estimated.

As the previous discussion indicated, ANN regression and ANN-GA optimization techniques may be used in inverse groundwater modeling. Although ANN regression was assessed in a single problem by *Sawyer et al.* [1995], the assessment was preliminary and more research is needed to assess this technique rigorously. In contrast, to the best of our knowledge, ANN-GA is yet to be assessed, especially in relation to hydrocarbon remediation designs. In this manuscript the focus is to assess the applicability of these two techniques in solving an inverse problem related to a free-product-contaminated unconfined aquifer. Previous work [*Lenhard and Parker*, 1990; *Kaluarachchi et al.*, 1990; *Kaluarachchi and Elliott*, 1995] related to free-product remediation designs has shown that soil grain-size distribution index of the van Genuchten parametric model α [*van Genuchten*, 1980] is critical in estimating the correct free-product volume [*Parker and Lenhard*, 1989; *Lenhard and Parker*, 1990]. In addition, saturated hydraulic conductivity of water K_{sw}, which is used to determine oil-phase conductivity, is important in predicting the free-product movement to a recovery well. Therefore these two parameters were used in this work using ANN regression and ANN-GA optimization. The base case data were obtained from the work of *Cooper et al.* [1995], which included fluid-table elevation data from a selected number of observation wells and free-product recovery information. The base case data were subsequently modified to implement heterogeneous conditions. Several example problems were solved using the two techniques, and their applicabilities were compared. In addition, the applicability of three different monitoring data sets generated from the same base problem in solving the inverse problems was assessed. Finally, a general set of guidelines for solving inverse problems using the two techniques is outlined.

2. Artificial Neural Network and Genetic Algorithm

To assess the applicability of ANN regression and ANN-GA optimization techniques in solving inverse groundwater problems, ANN and GA mathematics related to function approximation and near-optimal optimization, respectively, need to be comprehended. In the next section, a summary description on ANN and GA techniques is provided.

2.1. Artificial Neural Network

Several authors have discussed the theory and applications related to ANN [*Beale and Jackson*, 1991; *NeuralWare, Inc.*, 1991]. ANN is a flexible mathematical structure patterned after the biological functioning of the nervous system [*McCulloch and Pitts*, 1943], and a two-layer, feed-forward ANN is a universal function approximator [*Hecht-Nielsen*, 1987]. Figure 1 shows such an ANN for approximating an input-output function. The ANN consists of two weight layers corresponding to two neuron layers and connecting three nodal layers; the input layer processes no signal, and this layer is not considered to be a neuron layer. Each node is connected to all the nodes in the adjacent layers. Figure 1 shows an *I-J-K* ANN, and the notations used are

$$\mathbf{x} = [x_1, x_2, \cdots, x_I]^T \tag{1a}$$

$$\boldsymbol{\Omega} = [\Omega_1, \Omega_2, \cdots, \Omega_J]^T \tag{1b}$$

$$\mathbf{y} = [y_1, y_2, \cdots, y_K]^T \tag{1c}$$

$$\bar{\mathbf{w}} = [\bar{w}_1, \bar{w}_2, \cdots, \bar{w}_{jk}, \cdots, \bar{w}_{J+K}]^T \tag{1d}$$

$$\mathbf{w} = [w_1, w_2, \cdots, w_{ijk}, \cdots, w_{J(I+K)}]^T \tag{1e}$$

where $\mathbf{x}$ is an input vector of I components; $\boldsymbol{\Omega}$ is a hidden vector of J components; $\mathbf{y}$ is an output vector of K components; $\bar{\mathbf{w}}$ is a threshold-weight vector of $(J + K)$ components, where $\bar{w}_{jk}$ is the threshold of the jth neuron in the $(k + 1)$th neuron layer; and $\mathbf{w}$ is a weight vector of $J(I + K)$ components, where w_{ijk} is the weight connecting the ith neuron in the kth neuron layer to jth neuron in the $(k + 1)$th layer. As such, I and K fix the number of neurons in the input and output layers, and the *Hecht-Nielsen* [1987] suggestion may be used to define an upper limit of $J = (2I + 1)$ neurons in the hidden layer. Each hidden or output neuron is associated with a transfer function $f(\cdot)$, and a commonly used function is

$$f(\cdot) = \tanh(\cdot) = \frac{\exp(\cdot) - \exp(-\cdot)}{\exp(\cdot) + \exp(-\cdot)} \tag{2}$$

where tanh $(\cdot)$ is a hyperbolic-tangent function. In approximating the **x-y** function, **x** is fed at the input layer and it moves layer-by-layer toward the output layer. Each neuron receives a weighted sum from all the nodes in the previous layer, adds this sum to the threshold weight, and yields a nodal input; then the neuron passes this input through $f(\cdot)$ and yields a nodal output. At the output layer, **y** is approximated as

$$y_j = \Gamma_j(\bar{\mathbf{w}}, \mathbf{w}, \mathbf{x}) \tag{3a}$$

$$\Gamma(\cdot) = [\Gamma_1(\cdot), \Gamma_2(\cdot), \cdots, \Gamma_K(\cdot)]^T \tag{3b}$$

where $\Gamma(\cdot)$ is the underlying **x-y** vector function approximated by ANN and $\Gamma_j(\cdot)$ is the jth component of $\Gamma(\cdot)$. However, the accuracy of the approximation depends on $\bar{\mathbf{w}}$ and **w** estimates, and the estimation approach is designed during ANN development.

ANN is developed in two phases: training (accuracy or calibration) phase and testing (generalization or validation) phase. In general, a subset of patterns (**x-y** vectors) is first sampled from the domain into a training-and-testing subset S, and S is then exhausted by allocating the patterns to a training subset S_1 and a testing subset S_2. In the ANN training phase the objective is to determine the $\bar{\mathbf{w}}$ and **w** that minimize a specified error for the patterns contained in S_1. Often, this criterion is defined by the mean-squared-error function, which is defined as

$$E = \frac{1}{2}\sum_{i=1}^{K}(y_i - d_i)^2 \tag{4}$$

where E is the mean-squared-error of a training pattern in S_1 and d_i is the desired value corresponding to y_i. In general, a training pattern is selected at random from S_1, and E for this pattern is minimized using the standard or a modified backpropagation algorithm (BPA) [*Rumelhart et al.*, 1986; *NeuralWare, Inc.*, 1991]. BPA is a gradient-based algorithm, and the weight update equation may be written as

$$w^m = w^{m-1} - \mu\frac{\partial E^m}{\partial w^{m-1}} + \xi(w^{m-1} - w^{m-2}) \tag{5}$$

$$\mu \in (0, 1), \qquad \xi \in [0, 1]$$

where $w = w_{ij}$ or w_{ijk}, m is the updating index, μ is the training rate, and ξ is the momentum factor. It should be noted that $\xi = 0$ for $m = 1$. BPA continues updating $\bar{\mathbf{w}}$ and **w** until $m = \bar{M}$, where $\bar{M}$ is a user-defined number. In the ANN testing phase the objective is to determine the acceptability of the computed $\bar{\mathbf{w}}$ and **w** in minimizing the same error criterion for the patterns contained in S_2. As these patterns are not used in determining $\bar{\mathbf{w}}$ and **w**, ANN testing assesses domain generalization achieved by the trained ANN, thereby building the confidence level of the trained ANN for future estimations. At this stage, ANN is noted to depend on the number of training patterns in S_1. Usually, too many patterns may have to be sampled for approximating complex functions, and ANN implementation may become time consuming. Also, if a unique **x-y** function is nonexistent, ANN is not applicable, and any subsequent ANN approximations will be poor.

2.2. Genetic Algorithm

Several authors have discussed GA in detail [*Holland*, 1975; *Goldberg*, 1989]. GA is a heuristic, probabilistic, search-based optimization technique patterned after the biological process of evolution [*Holland*, 1975]. GA finds near-optimal solutions with high probability without becoming trapped to a local-optimal solution. GA searches near-optimal solutions using fitness values. Fitness measures the performance of a trial solution; high fitnesses are provided to feasible solutions closer to the global-optimal solution. Fitness is defined by combining the objective function with the constraints. Often, a penalty method is used to define the fitness. At the beginning, GA encodes an initial population of N trial solutions using binary strings of L bits (N is population size, and L is string length). Then GA associates a fitness to each string and propagates this initial population through G generations using these fitnesses. In propagating $(g - 1)$ to g (g is generation index), GA performs four operations: scaling, selection, crossover, and mutation. The procedure can be summarized as follows: (1) GA linearly scales the fitnesses in the $(g - 1)$th population within an appropriate range using a scaling coefficient C (C is the number of best strings expected in the scaled population) [*Goldberg*, 1989]; (2) GA updates this population by selecting strings with higher fitnesses with higher probabilities; (3) GA perturbs the resulting population by performing crossover with a probability of p_c; and (4) GA further perturbs the resulting population by performing mutation with a probability of p_m. As such, GA evaluates $T = N(G + 1)$ trial solutions, and the near-optimal solution is searched from these trial solutions. At this stage, GA is noted to perform too many fitness evaluations; these evaluations may be time consuming, and GA may generate slower solutions in real time. Also, if the global-optimal solution is tightly surrounded by poor solutions, then such a problem is considered to be GA deceptive. In such a condition, GA may not be applicable, and the corresponding solutions will be poor [*Goldberg*, 1989].

2.3. Artificial Neural Network and Genetic Algorithm

Several authors have discussed the combined approach of ANN-GA [*Rogers and Dowla*, 1994; *Roger et al.*, 1995]. Since ANN is a universal function-approximator, it may be used to approximate an explicit **x-y** function and hence bypass the time-consuming intermediate steps required in the rigorous mapping from **x** space to **y** space. As such, by approximating an explicit **x-y** (**x** is a decision variable vector and y is fitness), ANN may be used to speed up the fitness evaluations of GA, and ANN-GA may be used to generate faster solutions in real time.

3. Free-Product Migration and Recovery From Unconfined Aquifers

In this manuscript an areal two-phase flow analysis of simultaneous water and free-product pumping from a recovery well in an unconfined aquifer was used in the inverse analysis. In the next section the flow model is briefly described, and additional information is available from *Kaluarachchi et al.* [1990] and *Environmental Systems and Technologies, Inc.* [1994].

3.1. Governing Equations

The model formulation was based on the following assumptions: the subsurface is incompressible; the wettability (affinity to the solid matrix) increases from air to oil to water, the air phase remains at atmospheric conditions without producing resistance to water and oil flow, and oil and water are incompressible and mobile and are under isothermal and vertical equilibrium conditions.

After vertical integration of the three-dimensional two-phase (water and oil) flow equations subjected to the vertical equilibrium condition, the governing areal flow equations can be written as [*Parker and Lenhard*, 1989]

$$\frac{\partial V_w}{\partial t} = \frac{\partial}{\partial x}\left(T_{wx}\frac{\partial z_{aw}}{\partial x}\right) + \frac{\partial}{\partial y}\left(T_{wy}\frac{\partial z_{aw}}{\partial y}\right) + \sum_{k=1}^{N} R_{wk}\delta(x - x_k)(y - y_k) \tag{6a}$$

$$\frac{\partial V_o}{\partial t} = \frac{\partial}{\partial x}\left(T_{ox}\frac{\partial z_{ao}}{\partial x}\right) + \frac{\partial}{\partial y}\left(T_{oy}\frac{\partial z_{ao}}{\partial y}\right) + \sum_{k=1}^{N} R_{ok}\delta(x - x_k)(y - y_k) \tag{6b}$$

where V_p is p-phase specific fluid volume per unit area (p = o (oil) or w (water)); z_{pq} is fluid table elevation between p and q phases, where the capillary pressure head $h_{pq} = 0$ at $z = z_{pq}$; T_{pi} is the p-phase transmissivity along the i direction (i = x, y); R_{pk} is the p-phase pumping rate from recovery well k located at (x_k, y_k); δ is the dirac delta function; N is the number of wells; x is the horizontal-longitudinal coordinate; and y is the horizontal-transverse coordinate. The transmissivities for this formulation are given as

$$T_{wi} = K_{swi}[z_{ow} - z_l] \tag{7a}$$

$$T_{oi} = \frac{\rho_{ro}K_{swi}}{\mu_{ro}}\int_{z_l}^{z_u} k_{ro}\,dz \tag{7b}$$

where K_{swi} is the saturated hydraulic conductivity of water along the i direction, k_{ro} is the relative conductivity of oil, ρ_{ro} is the specific gravity of oil, μ_{ro} is the ratio of oil to water dynamic viscosities, z is the vertical coordinate, z_u is the maximum elevation to the presence of free-product, z_l is the lower effective elevation, and t is time. Equations (6a) and (6b) show the vertically integrated three-dimensional flow equations where the vertical heterogeneity is lumped together to form the areal flow equations.

The relationships governing the fluid-table elevations can be obtained using the vertical equilibrium condition and are given as [*Parker and Lenhard*, 1989]

$$z_{ao} = z_{ow} + H_o \tag{8a}$$

$$z_{aw} = z_{ow} + \rho_{ro}H_o \tag{8b}$$

$$H_o = z_{ao} - z_{ow} \tag{8c}$$

$$z_u = z_{ow} + \frac{\rho_{ro}\beta_{ao}H_o}{\rho_{ro}\beta_{ao} - (1 - \rho_{ro})\beta_{ow}} \tag{8d}$$

where H_o is free-product thickness; z_{ao} is the air-oil fluid-table elevation, where $h_{ao} = 0$; z_{ow} is the oil-water fluid-table elevation, where $h_{ow} = 0$; z_{aw} is the air-water fluid-table elevation, where $h_{aw} = 0$; h_{ao} is the air-oil capillary pressure head; h_{ow} is the oil-water capillary pressure head; β_{ao} is the air-oil fluid scaling parameter defined as the ratio between the water surface tension and the air-oil interfacial tension; and β_{ow} is the oil-water fluid scaling parameter defined as the ratio between the water surface tension and the oil-water interfacial tension.

3.2. Constitutive Relationships

The constitutive relationships for this model formulation are based on the van Genuchten soil parametric model with suitable fluid scaling relationships [*Parker et al.*, 1987]. A summary of the capillary pressure–saturation relationships can be given as follows:

$$S_{of} = S_t - S_w \tag{9a}$$

$$S_w = (1 - S_m)[1 + (\alpha\beta_{ow}h_{ow})^n]^{-m} + S_m \tag{9b}$$

$$S_t = (1 - S_m)[1 + (\alpha\beta_{ao}h_{ao})^n]^{-m} + S_m \quad m = 1 - n^{-1} \tag{9c}$$

$$h_{ow} = (1 - \rho_{ro})(z - z_{ow}) \tag{9d}$$

$$h_{ao} = \rho_{ro}(z - z_{ao}) \tag{9e}$$

where S_{of} is free-oil saturation, S_w is water saturation, S_t is total liquid (oil plus water) saturation, S_m is residual saturation, α is soil grain-size distribution index of the van Genuchten model, and n is a curve fitting parameter. The corresponding relative hydraulic conductivities can be given as

$$k_{rw} = (\bar{S}_w)^{0.5}[1 - (1 - \bar{S}_w^{1/m})^m]^2 \tag{10a}$$

$$k_{ro} = (\bar{S}_t - \bar{S}_w)^{0.5}[(1 - \bar{S}_t^{1/m})^m - (1 - \bar{S}_w^{1/m})^m]^2 \tag{10b}$$

$$\bar{S}_w = \frac{S_w - S_m}{1 - S_m} \tag{10c}$$

$$\bar{S}_t = \frac{S_t - S_m}{1 - S_m} \tag{10d}$$

3.3. Specific Oil Volumes

In the subsurface, oil can exist under three distinct conditions: (1) free or mobile, (2) trapped or immobile in the water phase, and (3) residual and gravity-drained oil held by capillary forces in the liquid unsaturated zone. As such, oil saturation can be expressed as

$$S_o = S_{of} + S_{ot} + S_{og} \tag{11}$$

where S_o is total oil saturation, S_{ot} is trapped oil saturation, and S_{og} is gravity-drained oil saturation. The corresponding specific oil volumes are related as follows:

$$V_o = V_{of} + V_{ot} + V_{og} \tag{12}$$

where V_o is total specific oil volume, V_{of} is free specific oil volume, V_{ot} is trapped specific oil volume, and V_{og} is gravity-drained specific oil volume. These specific volumes can be defined as

$$V_{of} = \int_{z_l}^{z_u} \phi S_{of}\,dz \tag{13a}$$

$$V_{ot} = \int_{z_l}^{z_u} \phi S_{ot}\,dz = \phi S_{or}(z_{ow} - z_{ow}^{\min}) \tag{13b}$$

$$\Delta V_{og} = \phi S_{og}\,\Delta z_{ao} \tag{13c}$$

where ϕ is porosity, S_{or} is maximum residual oil saturation in the liquid-saturated zone, $z_{ow}^{\min}$ is historical minimum z_{ow}, ΔV_{og} is change in V_{og} over a given time increment, S_{og} is maximum residual oil saturation in liquid-unsaturated zone, and Δz_{ao} is change in z_{ao} over a given time increment.

4. Research Approach

In this manuscript the focus is to assess the applicability of ANN regression and ANN-GA optimization techniques in solving inverse problems for parameter estimation. The problem used in the present work is related to a free-product migration and recovery from an unconfined aquifer. Once the base case scenarios and related properties were defined, simulations were performed to generate observed data at the monitoring and recovery wells for different scenarios, and these simulations were performed using the computer code ARMOS [*Environmental Systems and Technologies, Inc.*, 1994]. In the subsequent work, base scenarios were changed to develop various physical conditions to evaluate the useful ANN concepts related to architecture, sampling, and training, and GA concepts related to objective-function formulation, population size, and the number of generations. Finally, the applicability of ANN regression and ANN-GA optimization were compared, and a general set of guidelines for solving inverse problems was outlined.

Table 1. Soil and Fluid Properties and Recovery Well Data Used in the Homogeneous and Heterogeneous Base Problems

Common Properties		
Fluid		Soil
ρ_{ro} = 0.8		n = 2.5
μ_{ow} = 0.6		ϕ = 0.41
β_{ao} = 3.2		S_m = 0.2
β_{ow} = 1.45		S_{or} = 0.21
		S_{og} = 0.08
Zone	α, cm^{-1}	K_{sw}, m d^{-1}
	Heterogeneous Case	
1	4	12
2	2	4
3	5	23
4	2	1
5	1	1
	Homogeneous Case	
	7.5	10

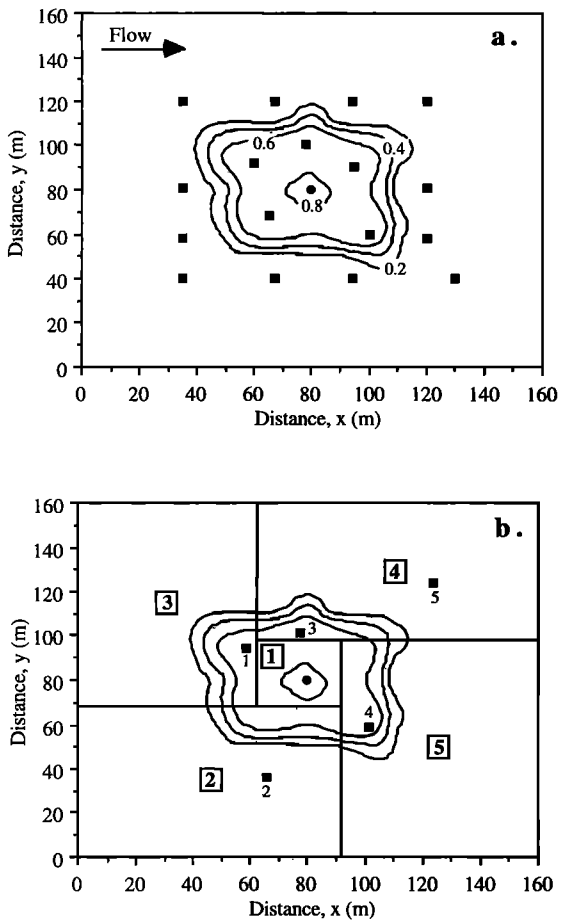

Figure 2. Physical domain used in the base case simulations: (a) homogeneous case with free-product thickness contours in meters and (b) the five different soil zones used in the heterogeneous case with same free-product thickness as in Figure 2a. Solid squares show the monitoring wells. Numbered wells indicate locations where observations were taken in the parameter estimation. Solid circle shows the recovery well.

4.1. Base Problem

The original problem was obtained from the work of *Cooper et al.* [1995] that corresponds to a homogeneous unconfined aquifer with an existing free-product plume. In free-product volume estimation, it is well understood that the parameters of the constitutive model play an important role in the overall predictions. In general, the soil grain-size distribution index α of the van Genuchten model is extremely sensitive to the volume predictions [*Lenhard and Parker*, 1990]. In recovery design a good understanding of the saturated hydraulic conductivity of water is also essential. Although there are a number of other fluid and soil parameters involved in the analysis, it is essential to know these two critical parameters for a good system design. Therefore in this work the focus will be to determine α and K_{sw} of the soil(s) under homogeneous and heterogeneous conditions using recovery/monitoring well data. Since these parameters are known from the original problem, these known values can be directly used to compare the accuracy of the estimated values. In the next section the homogeneous problem of *Cooper et al.* [1995] will be described, and results related to estimation of these two parameters will be discussed. Thereafter, the problem will be modified to accommodate heterogeneous conditions with five different soil zones.

4.1.1. Homogeneous condition. Figure 2a shows a 160 m × 160 m homogeneous unconfined aquifer with an existing gasoline plume. The subsurface soil consists of medium sand. The site has one recovery well at the center and 17 monitoring wells in and around the oil plume. The radius of the recovery well is 0.15 m, and free-product thicknesses H_o in these monitoring wells varies from 0 to 0.82 m. The elevation of the groundwater table z_{aw} at the left boundary is 20.5 m with a gradient of 0.3125% from the left boundary to the right boundary. Table 1 shows the soil and fluid properties of the site. Figure 2a also shows the observed initial free-product thickness contours at the site using the monitoring data obtained from the 17 wells. Using these values, the estimated initial free-product volume V_{oI} was 215 m^3.

For numerical simulations the site was discretized into 324 rectangles using 361 nodes, with the smallest spacing of 3 m

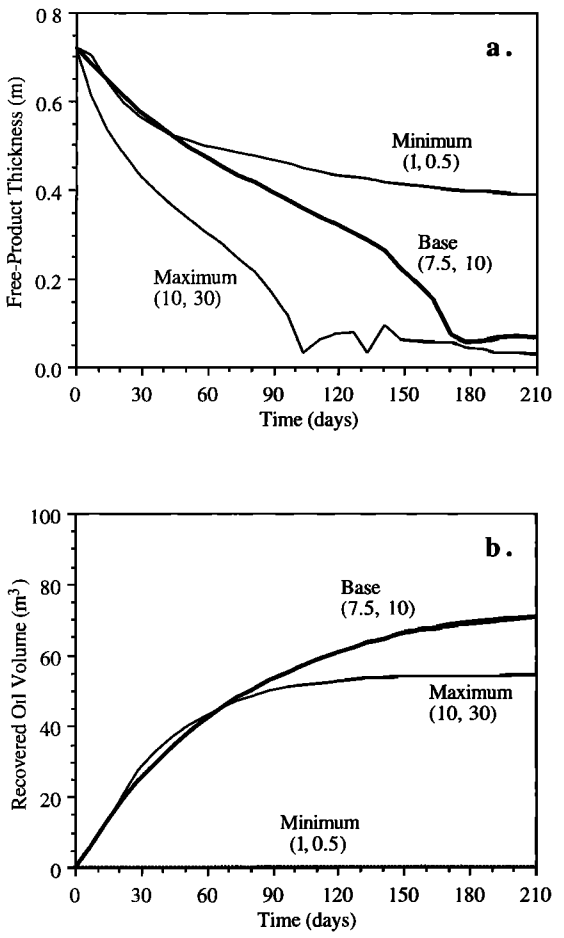

Figure 3. Sensitivity of the base case homogeneous problem for different combinations of (α, K_{sw}) values selected from the stipulated ranges: (a) free-product thickness versus time at monitoring well 1 and (b) recovered free-product volume versus time.

near the recovery well. The four boundary conditions applied were zero oil flux at the perimeter, assuming the free-product plume will not reach the boundary over the simulation time, and z_{aw} fixed at initial values along the perimeter. At the recovery well the boundary conditions were R_o = 1 m^3/d and R_w = 50 m^3/d (R_o is the free-product pumping rate, and R_w is the water pumping rate) [*Cooper et al.*, 1995]. The numerical model has the capability of switching the recovery well boundary condition between type 1 and type 2 automatically, depending on the actual rate of oil movement to the well. If the actual free-product flow rate is less than the stipulated value, then the model will assign the H_o = 0 condition.

In order to estimate the parameters from recovery/monitoring well data, the possible bounds of the parameters need to be established. Based on the soil type present in the site, which is medium sand in this case, the ranges of α and K_{sw} were assumed to be (1, 10) and (0.5, 30), respectively. The base values of (α, K_{sw}) were (7.5, 10), as given in Table 1. The problem was solved over 210 days for (α, K_{sw}) combinations of {(1, 0.5), (7.5, 10), (10, 30)} to understand the free-product migration pattern and to evaluate the potential difficulty of the parameter estimation problem. Figure 3a shows the three H_o curves at monitoring well 1 for the three (α, K_{sw}) pairs. As (α, K_{sw}) is increased, H_o decreases monotonically through time, and an inverse function (one-to-one mapping from H_o space to (α, K_{sw}) space) appears to exist. Figure 3b shows the three free-product recovery volume V_{oR} curves for the three pairs. As (α, K_{sw}) is increased, V_{oR} varies nonmonotonically through time, and an inverse function (one-to-one mapping from V_{oR} space to (α, K_{sw}) space) does not appear to exist. Thus, depending on the observation location and type, the inverse function may or may not be ill-posed.

4.1.2. Heterogeneous condition. The physical domain is identical to the homogeneous base problem except, in this case, the subsurface was divided into five different soil zones (Figure 2b) with properties given in Table 1. As mentioned earlier, this problem is complex as the parameter estimation needs to evaluate five values of (α, K_{sw}) pairs for the five soil zones using recovery/monitoring well data. Once computed, the values can be compared with the known exact values given in Table 1. Using the free-product contours given in Figure 2a, the estimated initial free-product volume V_{oI} = 51 m^3. Note that in this case we used the same observed free-product heights as in the homogeneous case with different soil types and therefore the initial free-product volume changed. For this example the inverse problem is 10-dimensional and the problem is likely to be ill-conditioned. As such, the problem provides a good test for the applicability of these ANN and GA methods.

4.1.3. Recover/monitoring well data. The monitoring or observed data required in the parameter estimation were produced by simulating the numerical model with exactly known parameters, and a subset of these results was selected for use in the parameter estimation. The subset of results was obtained using the initial 140 days at 14-day intervals. In a typical situation, this scenario corresponds to a short-term recovery operation where the information is used in parameter estimation to update the existing recovery design and operation. Therefore with the initial conditions, the observed number of data sets corresponds to 11 observations per monitoring well. The observations correspond to the recovered free-product volume versus time and free-product thickness versus time from monitoring and recovery wells. These observations correspond to 50 observations of H_o and 10 observations of V_{oR}. In developing the observed data, the subset of results was added with a random noise of ±5% to reflect the observation errors. Further details on the added noise are given below.

4.2. Function Approximation Using ANN

ANN development requires precise identification of the input and output vectors. In ANN regression, ANN is used to approximate an underlying inverse function (one-to-one mapping from observation space to parameter space), and the inverse function may be expressed as

$$\begin{bmatrix} \boldsymbol{\alpha} \\ \mathbf{K_{sw}} \end{bmatrix} = \boldsymbol{\Gamma}^{-1} \begin{bmatrix} \mathbf{H_o} \\ \mathbf{V_{oR}} \end{bmatrix} \tag{14a}$$

where $\boldsymbol{\alpha}$ is a vector of all α parameters, $\mathbf{K_{sw}}$ is a vector of all K_{sw} parameters, $\mathbf{H_o}$ is a vector of all H_o observations; $\mathbf{V_{oR}}$ is a vector of all V_{oR} observations, and $\boldsymbol{\Gamma}^{-1}$ is a vector of all Γ^{-1} (inverse) functions. In ANN-GA optimization, ANN is used to approximate a forward underlying function from the parameter space to the observation space, and the forward function may be expressed as

$$\begin{bmatrix} \mathbf{H}_o \\ \mathbf{V}_{oR} \end{bmatrix} = \mathbf{\Gamma} \begin{bmatrix} \boldsymbol{\alpha} \\ \mathbf{K}_{sw} \end{bmatrix} \tag{14b}$$

where $\mathbf{\Gamma}$ is a vector of all Γ (forward) functions.

4.3. Training-and-Testing Subsets

As mentioned previously, the α and K_{sw} parameter bounds were selected as $\alpha \in [1, 10]$ and $K_{sw} \in [0.5, 30]$, and the training-and-testing subset S was sampled from this domain. For the homogeneous condition, a subset of 100 realizations of (α, K_{sw}) was sampled at random; $(\mathbf{H}_o, \mathbf{V}_{oR})$ for each realization was determined using the numerical simulations, and the 100 patterns were placed in S. For the heterogeneous condition, a subset of 300 realizations of $(\boldsymbol{\alpha}, \mathbf{K}_{sw})$ was sampled at random; $(\mathbf{H}_o, \mathbf{V}_{oR})$ for each realization was determined using numerical simulations, and the 300 patterns were placed in S.

4.4. Allocation Method With Noise

In allocating S to a training subset S_1 and a testing subset S_2, a simple allocation method with noise is applied pattern-by-pattern on S. This method is based on a user-defined expected fraction $\bar{f}$, where $\bar{f} \in (0, 1)$. In allocating a pattern to S_1 or S_2, a random number $r \in (0, 1)$ is generated. If $r \leq \bar{f}$, the pattern is allocated to S_1 without noise; otherwise, it is allocated to S_2 with noise. For both inverse and forward ANNs given by (14a) and (14b), respectively, noise is added to the $(\mathbf{H}_o, \mathbf{V}_{oR})$ components of the testing patterns. The sequence of r is simulated using a random number generator initiated by a seed ρ, and the allocation method becomes a function of $\bar{f}$ and ρ. For a given S the allocation method is expected to allocate $\bar{f}$ and $(1 - \bar{f})$ fractions of S to S_1 and S_2, respectively, and this approach is expected to generate S_1 and S_2 of different or similar sizes for different values of $\bar{f}$ and ρ. In adding noise to an observed H_o of a testing pattern, a simple procedure is followed; the H_o is replaced with a random value taken from the interval $H_o \in [H_o(1 - \sigma), H_o(1 + \sigma)]$, where σ is a user-defined noise in decimal format. In the present case, $\sigma = 0.05$, and it corresponds to $\pm 5\%$ noise. A similar procedure was followed with V_o.

4.5. ANN Development

In this manuscript, ANNs were developed using the Neural Works Professional II/Plus, a commercially available software package [*NeuralWare, Inc.*, 1991]. The internal parameters of the ANN included initial weight distribution, transfer function, input-output scaling, training rate μ, momentum factor ξ, training rule, and the number of weight updates $\bar{M}$. ANN training was performed using the default values in the package, except $\bar{M}$ and seed ρ. A common ρ was used for the allocation method and training phase for consistency. As such, the initial weights were randomly distributed over $[-0.1, +0.1]$. The transfer function used was tanh(·). The input-output scaling was performed using the minimum and maximum values of each input and output component contained in S_1; the input components are linearly scaled over $[-1, +1]$, and the output components are linearly scaled over $[-0.8, +0.8]$ for the tanh (·) function. The generalized delta training rule for BPA was used. As such, the performance of BPA may be expressed as

$$\eta = \eta(\rho, \bar{f}, J) \tag{15}$$

where η is performance of BPA.

4.6. Performance Criteria

ANN is trained to approximate a vector function $\Gamma(\cdot)$, and the performances of ANN in approximating each ith component $\Gamma_i(\cdot)$ during training and testing phases need to be assessed. In the ANN training phase the objective is to match the desired response d_i with the ANN response y_i at each ith output neuron for all the patterns in S_1. As such, the performance of training in approximating $\Gamma_i(\cdot)$ is assessed using the correlation coefficient R_i. If the ANN contains too many output nodes, presentation of all values of R_i becomes difficult, and the average of the R_i values is provided. In addition, the performance of training is assessed using scatterplots of y_i versus d_i, and the scatter of (d_i, y_i) from the 45° line is assessed using two error bounds to help assess the percent deviation of y_i from d_i. Also, these plots can provide an improved interpretation of the R_i values. In the ANN testing phase the objective is to match d_i with y_i at each ith output neuron for all the patterns in S_2. As such, the performance of testing in predicting $\Gamma(\cdot)$ is assessed by repeating the above procedure for S_2.

5. Results and Discussion

5.1. Homogeneous Case

5.1.1. ANN regression. In this case there are 60 inputs $(I = 60)$, corresponding to the 60 observations, and two outputs $(K = 2)$, corresponding to (α, K_{sw}). Therefore a hidden layer of 121 neurons $(J = 121)$ was used [*Hecht-Nielsen*, 1987], and a 60-121-2 ANN was used to approximate the unknown inverse function. The allocation method was used for allocating S to S_1 and S_2. As no guideline for allocation exists, $\bar{f} = 0.5$ was used to minimize bias on the allocation, and $\rho = 1$ was arbitrarily taken. In Figures 4a and 4b, the empty squares represent the testing results for α and K_{sw}, respectively. For both parameters the computed $R = 0.990$ and the errors of prediction were within $\pm 10\%$ for the base problem. The ANN estimated $(\alpha, K_{sw}) = (6.9, 10.9)$, which are close to the true values, (7.5, 10).

Table 2 summarizes the sensitivity results of the ANN regression analysis to $\bar{f} = \{0.25, 0.50, 0.75\}$, $\rho = \{1, 5, 257\}$, and $J = \{61, 121, 242\}$. For $\bar{f} = \{0.25, 0.50, 0.75\}$, ANN performance increases with increasing $\bar{f}$. As $\bar{f}$ increases, ANN receives more patterns for training, acquires a clearer picture of the domain, and improves the generalization. For $\rho = \{1, 5, 257\}$, ANN performance is worst for $\rho = 1$, best for $\rho = 5$, and intermediate for $\rho = 257$. As ρ varies, ANN acquires different weight configurations due to the entrapment of BPA at different local optimal solutions. For $J = \{61, 121, 242\}$, ANN performance slightly decreases. This suggests that ANN is not sensitive to J, and a much smaller J than that suggested by *Hecht-Nielsen* [1987] may be used. Finally, it should be noted that testing patterns were not included in ANN training and each pattern may be considered as an unique inverse problem. In ANN testing the ANN basically solved 43 independent inverse problems with parameters selected at random. As the solutions were acceptable, ANN presents itself to be a robust regression technique for determining parameter estimates of the homogeneous soil as indicated by Figures 4a and 4b.

5.1.2. ANN-GA optimization. In this case, the goal of ANN training is to bypass numerical simulations. In solving the optimization model, GA requires observed data for each trial α and K_{sw} for fitness evaluations. Finding these observed data using numerical simulations would be time consuming, and

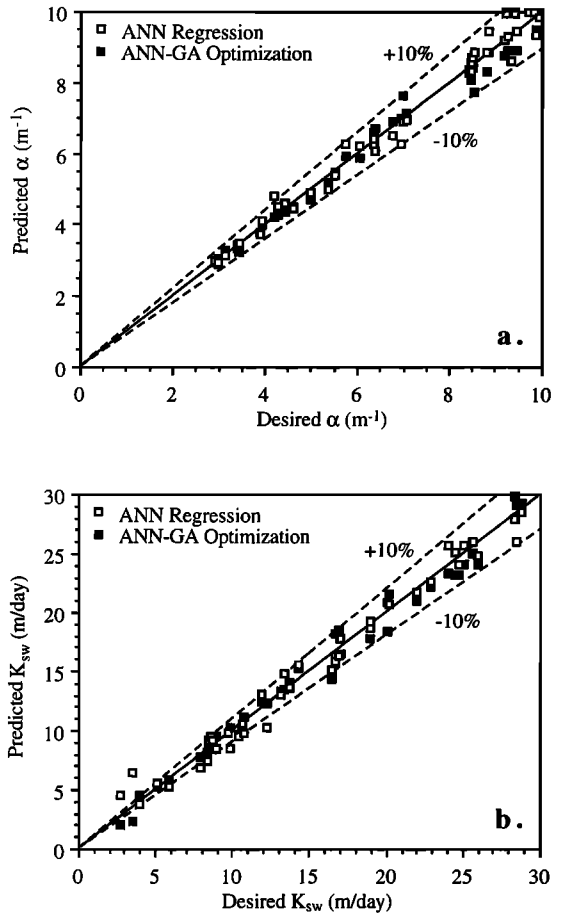

Figure 4. Performance of ANN regression for the homogeneous case, where solid symbols refer to ANN-GA and open symbols refer to ANN regression. Solid line refers to the ideal case, and dashed lines show the maximum error bounds: (a) α and (b) K_{sw}.

ANN was trained to simulate this task. In the following, ANN application in simulations will first be discussed followed by a discussion on GA application in optimization.

Contrary to the previously discussed ANN, the present ANN had two inputs ($I = 2$), corresponding to (α, K_{sw}), and 60 outputs ($K = 2$), corresponding to 60 observed data. A hidden layer of five neurons ($J = 5$) was used [*Hecht-Nielsen*, 1987],

Table 2. Sensitivity Analysis for ANN Regression and ANN-GA Optimization in Solving the Homogeneous Base Problem

		ANN Regression			ANN-GA	
ρ	$\bar{f}$	J	α	K_{sw}	J	Average R
1	0.25	121	0.976	0.887	5	0.846
1	0.50	121	0.992	0.990	5	0.960
1	0.75	121	0.995	0.992	5	0.938
1	0.50	121	0.992	0.990	5	0.960
5	0.50	121	0.989	0.988	5	0.908
257	0.50	121	0.987	0.976	5	0.950
1	0.50	61	0.996	0.991	3	0.844
1	0.50	121	0.992	0.990	5	0.960
1	0.50	242	0.987	0.989	10	0.973

and a 2-5-60 ANN was used to approximate the required function. The allocation method was used for allocating S to S_1 and S_2, with $\bar{f} = 0.5$ and $\rho = 1$. Figure 5 shows the scatterplot of predicted and desired recovered free-product volumes at $t = 14$ days. The regression analysis for this plot produced $R = 0.990$, and the error of predicting was within ±15%.

Table 2 summarizes the sensitivity results of ANN-GA analysis to $\bar{f} = \{0.25, 0.50, 0.75\}$, $\rho = \{1, 5, 257\}$, and $J = \{61, 121, 242\}$. For $\bar{f} = \{0.25, 0.50, 0.75\}$, average ANN performance varied from 0.846 to 0.960 to 0.938. The reduction of correlation with $\bar{f}$ is probably due to the use of constant number of weight updates. At a large $\bar{f}$ the number of training patterns increases, and the weights need to be updated for a longer period to attain a higher performance. Also, the average R is not a good measure of the ANN performance. For $\rho = \{1, 5, 257\}$, average performance is best for $\rho = 1$, worst for $\rho = 5$, and intermediate for $\rho = 257$. As ρ was varied, BPA became entrapped at different local optimal solutions. For $J = \{61, 121, 242\}$, average ANN performance increased with increasing J. This suggested that a much larger J than that suggested by *Hecht-Nielsen* [1987] may be used when few inputs are present. However, the J suggested by Hecht-Nielsen still appears acceptable.

For the GA used in the optimization analysis for error minimization, the objective function used was

$$\min E = \frac{1}{2} \sum_{i=1}^{K} (y_i - d_i)^2 R_i$$

subject to the constraints given by the spatial domain definition in the training-and-testing subset section. ANN has interpolation errors, and it predicts the output with a varying degree of correlation. As such, R_i of the testing set is used to provide a higher confidence on the outputs with a higher correlation. Also, the scaled y_i and d_i are used to reduce the bias of E to extreme values. Scaling similar to ANN training was performed, which is considered to be a simpler and a more consistent approach. The minimum and maximum values of the d_i in the training set were used to scale d_i and y_i. Thereafter, ANN-GA was implemented to solve the base problem.

In finding the optimal estimates, GA was implemented with $L = 10$, $C = 2$, $N = 30$, $G = 50$, $p_c = 0.6$, $p_m = 0.033$,

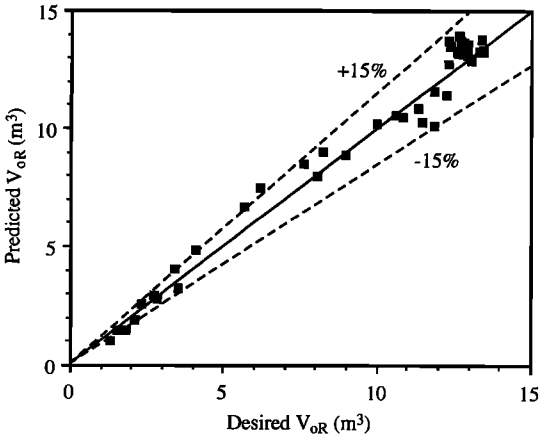

Figure 5. Performance of ANN regression in predicting the recovery volume at 14 days for the homogeneous base case. Solid line refers to the ideal case, and dashed lines show the maximum error bounds.

and $\rho = 1$ [*DeJong*, 1975; *Goldberg*, 1989]. Thus an initial population of 30 strings was sampled at random from the decision space defined by the spatial domain, and this population was taken through 50 generations. Figure 6 shows the minimum, average, and maximum objective values of the population at any generation. After the 10th generation, the average became close to the minimum, and the population converged approximately to a near-optimal solution. Table 3 shows that for the base problem, ANN-GA found $(\alpha, K_{sw}) = (7.8, 10.5)$, which are close to the true values (7.5, 10).

Table 3 summarizes the sensitivity result of ANN-GA to (N, G) combinations of $\{(30, 50), (14, 100), (60, 25)\}$, corresponding to $T = \{1530, 1414, 1560\}$, where $T = N(G + 1)$ is the total number of fitness evaluations performed by ANN-GA in propagating N strings through G generations. As fitness evaluation constitutes the major computational component of ANN-GA, this evaluation serves as an index to determine the computational effort of ANN-GA. Thus Table 3 compares three distinct combinations of (N, G) of comparable computational efficiency in determining the optimal estimates. For all these combinations, ANN-GA has found the true estimates satisfactorily. Although the final results suggest the estimates may have been obtained using combinations of (14, 44), (60, 10), or (30, 37) in order of increased computational effort, such a notion is difficult to anticipate, and $T = 1500$ may be used to determine the combination of N and G. In searching the optimal solutions, GA required 1500 simulations, and ANN solved the problem in 15 s. In the absence of ANN, the search would have been extremely time consuming. The numerical model required 5 min to solve the base problem, and GA would require approximately 5 days to complete the 1500 simulations in the same computer.

Finally, each problem represented by a given testing pattern was solved using ANN. The results are presented in Figures 4a and 4b for α and K_{sw}, respectively. For the 43 independent inverse problems with parameters selected at random, ANN-GA found the true estimates within ±10%. As such, ANN-GA can also be considered as a robust technique for parameter estimation in a homogeneous soil.

5.1.3. Effect of different types of observed data. ANN regression and ANN-GA optimization were observed to provide excellent parameter estimates for homogeneous soils. As ANN regression is simpler and straightforward to use and implement, it is used to understand the physical behavior of the problem. The task here is to identify which observation data combination from H_o only, V_o only, or H_o and V_o is the best for accurate prediction of the α and K_{sw} values.

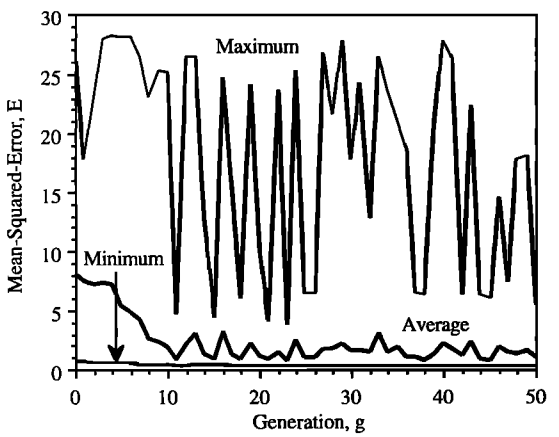

Figure 6. Performance of GA demonstrated using the different objective values during different generations for the homogeneous base case.

Table 3. Solutions to the Homogeneous Base Problem Using ANN-GA Optimization With Different (N, G) Combinations

Parameter	Base	ANN Regression	ANN-GA Optimization		
			Combination 1	Combination 2	Combination 3
$\tilde{f}$	...	...	0.5	0.5	0.5
N	...	...	30	15	60
G	...	...	50	10	25
T	...	...	1530	1414	1560
α	7.5	6.90	7.77	7.53	7.58
K_{sw}	10	10.93	10.5	10.6	10.6
N^*	...	...	18	3	26
G^*	...	...	37	44	10
E^*	...		0.293	0.267	0.270

*Computed best estimates.

First, the use of H_o only is studied. As there are 50 observed data of H_o and two unknown parameters, a 50-101-2 ANN was used [*Hecht-Nielsen*, 1987]. In Figure 7a the solid squares present the results for α in testing. The computed $R = 0.963$ and the errors of prediction are within ±10% except for one outlier. In Figure 7b the solid squares represent the results for K_{sw} in testing. Again the computed $R = 0.993$ and the errors of prediction are within ±10% except for one outlier. Here ANN estimated $(\alpha, K_{sw}) = (7.1, 10.7)$, which were close to the true values of (7.5, 10) as given by Table 4. Second, the use of V_o only is studied. As there are 10 observed values of V_o and two parameters to be estimated, a 10-21-2 ANN was used [*Hecht-Nielsen*, 1987]. In Figure 7a the crosses present the results for α in testing. The computed $R = 0.994$ and the errors of prediction are within ±15% with no outliers. In Figure 7b the crosses present the results for K_{sw} in testing. As in the previous case, the computed $R = 0.957$ and the errors of prediction are within ±15% except for one outlier. When the observed data of the base problem were used, ANN estimated $(\alpha, K_{sw}) = (7.4, 10.4)$, which were close to the true values of (7.5, 10.0) as given by Table 4. Third, the use of both H_o and V_o observations is considered, and this corresponds to the results presented above in the ANN regression. As such, the empty squares of Figures 4a and 4b were replotted in Figures 7a and 7b. As mentioned earlier, the computed R values for both parameters were 0.990, and the errors of predicting them were within ±10%. When the observed data of the base problem were used, ANN estimated $(\alpha, K_{sw}) = (6.7, 10.9)$ as given in Table 4.

Overall results indicated little difference in accuracy between the different data types. In general, Table 4 results suggest that the V_o-only option provided the best estimate, followed by H_o only, and V_o and H_o sets in predicting the base case parameters. However, the overall testing accuracy decreased in the same order (Figures 7a and 7b).

5.2. Heterogeneous Case

5.2.1. ANN regression. Here the domain was divided into five different soil zones as shown by Figure 2b, with each zone

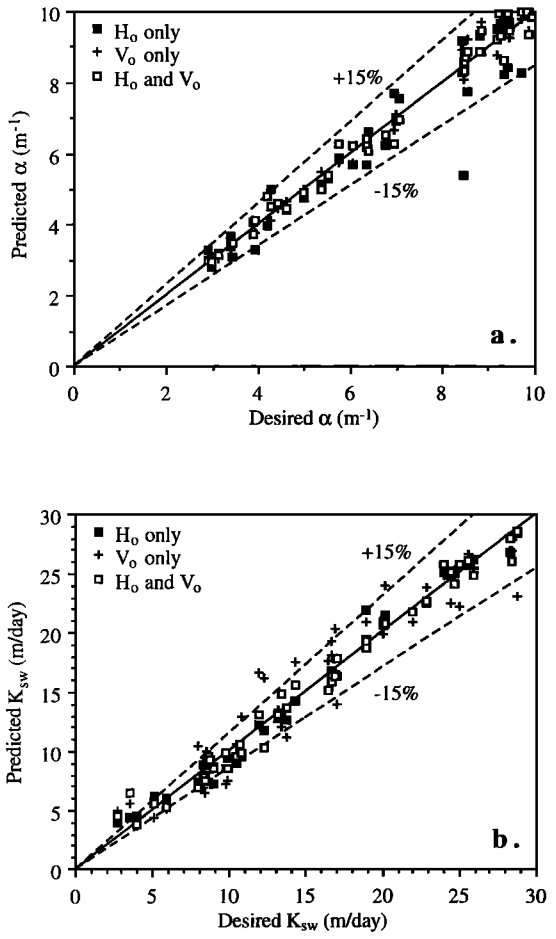

Figure 7. Accuracy of the predicted parameters using different combinations of observation data types for the homogeneous base case. Solid line refers to the ideal case, and dashed lines show the maximum error bounds: (a) α and (b) K_{sw}.

representing one set of α and K_{sw} values and the total number of parameters to be estimated is 10. As there are 60 inputs (I = 60), corresponding to the 60 observations, and 10 outputs (K = 10), a hidden layer of 121 neurons (J = 121) was used [*Hecht-Nielsen*, 1987], and a 60-121-10 ANN was used to approximate the unknown inverse function. Also, 300 patterns were considered (S = 300), and 100,000 weight updates were considered in training. The allocation method was used for allocating S to S_1 and S_2. As no guideline for allocation exists, $\bar{f}$ = 0.5 was used to minimize bias on the allocation, and ρ = 1 was arbitrarily taken. ANN testing showed the average R = 0.463 (Table 5). Although training was satisfactory, the ANN performance in testing was extremely poor. This suggested that either ANN may approximate the inverse function, provided sufficient training patterns are considered to generalize the parameter domain (one-to-one mapping present), or ANN is overfitting the data points without approximating the inverse function, which is difficult to establish or nonexistent (one-to-one mapping is absent). As the ANN performance was poor, the estimated parameters were far from their true values and therefore are not shown here. The objective of ANN regression was to approximate the true values, and this was not achieved with 300 patterns.

Table 5 summarizes the sensitivity result of ANN regression in testing to $\bar{f}$ = {0.25, 0.50, 0.75}, ρ = {1, 5, 257}, and J = {61, 121, 242}. For $\bar{f}$ = {0.25, 0.50, 0.75}, ANN performance increased with increasing $\bar{f}$. As $\bar{f}$ increases, ANN receives more patterns for training, acquires a clearer picture of the domain, and improves the generalization. However, the increase occurred at a slow pace, suggesting the inverse function to be difficult to model, and the patterns needed for domain generalization may become computationally unattractive. For ρ = {1, 5, 257}, ANN performance slightly varied; it is worst for ρ = 1, best for ρ = 5, and intermediate for ρ = 257. As ρ varies, ANN acquires different weight configurations due to the entrapment of BPA at different local optimal solutions. For J = {61, 121, 242}, ANN performance varied slightly. This suggests that ANN is not very sensitive to J, and a much smaller J than that suggested by *Hecht-Nielsen* [1987] may be used.

The results showed that ANN regression was not encouraging for the heterogeneous case. Possible reasons for this difficulty are that one-to-one mapping is difficult to establish and the inverse function is probably nonexistent. If either of these conditions exists, then ANN becomes confused and loses accuracy in trying to balance multiple outputs for a particular input. Since the ANN regression showed poor results, the next section will focus on ANN-GA optimization.

5.2.2. ANN-GA optimization. For the present case, ANN has 10 inputs (I = 10), corresponding to (α, K_{sw}) of the five zones, and 60 outputs (K = 60), corresponding to 60 observed data. A hidden layer of 21 neurons (J = 10) was used [*Hecht-Nielsen*, 1987], and a 10-21-60 ANN was used to approximate the required function. The allocation method with $\bar{f}$ = 0.5 and ρ = 1 was used for allocating S to S_1 and S_2.

Table 5 summarizes the sensitivity result of the ANN part of the overall analysis to $\bar{f}$ = {0.25, 0.50, 0.75}, ρ = {1, 5,

Table 4. Inverse Modeling of the Homogeneous Base Problem Using Three Different Observation Data Types

	Correlation Coefficient R			Estimated Value	
Data Type	ANN	α	K_{sw}	α	K_{sw}
Base	...	...	...	7.5	10
H_o only	50-101-2	0.963	0.993	7.2	10.7
V_o only	10-21-2	0.994	0.957	7.4	10.4
H_o and V_o	60-121-2	0.992	0.990	6.9	10.9

Table 5. Sensitivity Analysis for ANN Regression and ANN-GA Optimization in Solving the Heterogeneous Base Problem

		ANN Regression		ANN-GA	
ρ	$\bar{f}$	J	Average R	J	Average R
1	0.25	121	0.411	21	0.670
1	0.50	121	0.463	21	0.828
1	0.75	121	0.533	21	0.891
1	0.50	121	0.463	21	0.828
5	0.50	121	0.476	21	0.839
257	0.50	121	0.482	21	0.839
1	0.50	61	0.534	11	0.828
1	0.50	121	0.463	21	0.828
1	0.50	242	0.523	42	0.843

Table 6. Sensitivity of ANN-GA Optimization to N, G, and $\bar{f}$ in Solving the Heterogeneous Base Problem

			(N, G) Combinations			Effects of $\bar{f}$		
Zone	Parameter	Base	GA	GA	GA	GA	GA	GA
	$\bar{f}$	···	0.5	0.5	0.5	0.25	0.50	0.75
	N	···	30	14	60	14	30	60
	G	···	50	100	25	100	50	25
	T	···	1530	1414	1560	1414	1530	1560
1	α	4	4.38	4.29	4.23	3.54	4.38	4.01
2	α	2	2.51	7.28	2.41	9.93	2.51	5.70
3	α	5	3.17	3.33	2.70	4.21	3.17	2.00
4	α	2	5.62	1.19	6.40	1.55	5.62	4.37
5	α	1	1.09	5.06	1.37	1.63	1.09	1.15
1	K_{sw}	12	15.5	16.3	7.2	8.2	15.5	14.3
2	K_{sw}	4	25.9	1.4	0.7	2.7	25.9	26.6
3	K_{sw}	23	28.3	11.1	4.5	8.7	28.3	14.5
4	K_{sw}	1	4.6	20.4	5.0	1.9	4.6	17.7
5	K_{sw}	1	0.9	0.9	1.4	7.7	0.9	0.8
	N^*	···	13	7	56	8	13	16
	G^*	···	42	47	25	45	42	25
	E^*	···	0.522	0.725	0.650	0.397	0.522	0.621
V_{oR}	51.3	63.9	103.8	64.2	90.2	63.8	68.9	

257}, and J = {61, 121, 242}. For $\bar{f}$ = {0.25, 0.50, 0.75}, average ANN performance varied from 0.670, 0.828, and 0.891. The qualitative behavior of the results is similar to that of the homogeneous case with ANN-GA. The objective function for the ANN-GA was subject to the constraints given by the spatial domain for each α and K_{sw}. In finding the optimal estimates, GA was implemented with L = 10 (substring length), C = 2, N = 30, G = 50, p_c = 0.6, p_m = 0.033, and ρ = 1 [*DeJong*, 1975; *Goldberg*, 1989]. Thus an initial population of 30 strings was sampled at random from the decision space, and this population was taken through 50 generations. The behavior of the minimum, average, and maximum objective values of the population at any generation was similar to the homogeneous case given in Figure 6. After the 10th generation, the average became close to the minimum, and the population converged approximately to near-optimal solutions. Table 6 summarizes the results found by ANN-GA for the base problem. Although the estimated objective value E^* = 0.522, the estimated parameter values are not close to the true values. However, considering the instability and nonuniqueness of the inverse problem, the estimated values can be considered to be acceptable. In zone 1, where the center of the free-product plume is located, the estimated parameters of (4.4, 15.5) are close to the true values of (4.0, 12).

Also, Table 6 summarizes the sensitivity results of ANN-GA to (N, G) combinations of {(30, 50), (14, 100), (60, 25)}, corresponding to T = {1530, 1414, 1560}. The objective values were 0.522, 0.725, and 0.650 respectively. The results indicate that the parameter estimates appear to be nonunique. However, in a true parameter estimation problem, the true values of the parameters are unknown. In such a situation the accuracy of the estimate is evaluated by comparing to the secondary observations [*Yeh*, 1986]. In this case the accuracy of predicting the product volume and the free-product thickness at observation wells is a good measure of accuracy of the estimates. Table 6 summarizes the predicted initial product volume V_{oI} for different estimates. As such, the different (N, G) combinations of (30, 50), (14, 100), and (60, 25) predicted V_{oI} = 63.9, 103.8, and 64.2 m^3, respectively, when the desired value is 51.3 m^3. Figures 8a and 8b compare the H_o and V_o distributions with time up to 210 days at monitoring well 1 and the recovery well, generated by the three parameter sets. The bold line is the ideal condition depicted by the true values, and the solid squares show the point observations with noise used in solving the inverse problem. In Figure 8a the three predictions for H_o may be considered comparable, and the robustness of the three parameter sets is difficult to rank. In Figure 8b the predictions for V_o are markedly different, and the parameter sets may be ranked. According to these results, the best combination is (30, 50), worst is (14, 100), and intermediate is (60, 25). The (30, 50) and (60, 25) predicted the desired V_o and hence the initial V_{oI} also (Table 6). The results suggest that several GA solutions should be used in the analysis, and an acceptable solution should be determined by considering their prediction of observed V_o or H_o. To determine the different solutions, different combinations of (N, G) with T = 1500 may be used.

5.2.3. Sensitivity of ANN-GA to size of training set. Sensitivity of ANN-GA to training set size was evaluated using $\bar{f}$ = {0.25, 0.50, 0.75}. The problem is solved using the three combinations of (N, G); the parameter estimates from each solution are checked with the observed V_o, and the prediction that closely matched the observed was judged by visual inspection.

Table 6 summarizes the sensitivity results for the heterogeneous case. For $\bar{f}$ = {0.25, 0.50, 0.75}, the initial V_{oI} estimates were 90.2, 63.8, and 68.9 m^3, when the desired value is V_o = 51.3 m^3. As $\bar{f}$ changes from 0.25 to 0.50, the accuracy of the V_o estimation improves significantly. As $\bar{f}$ increased from 0.50 to 0.75, the solution slightly degraded, while it is expected to improve. This contradiction cannot be answered through a simple argument. An inverse problem is a difficult problem to solve for a variety of reasons, as explained above. In addition, ANN has interpolation errors and the objective function of the ANN training and the inverse problems are plagued with local optimal solutions. As such, $\bar{f}$ = 0.5 appears reasonable for training the ANN for the present problem. Therefore

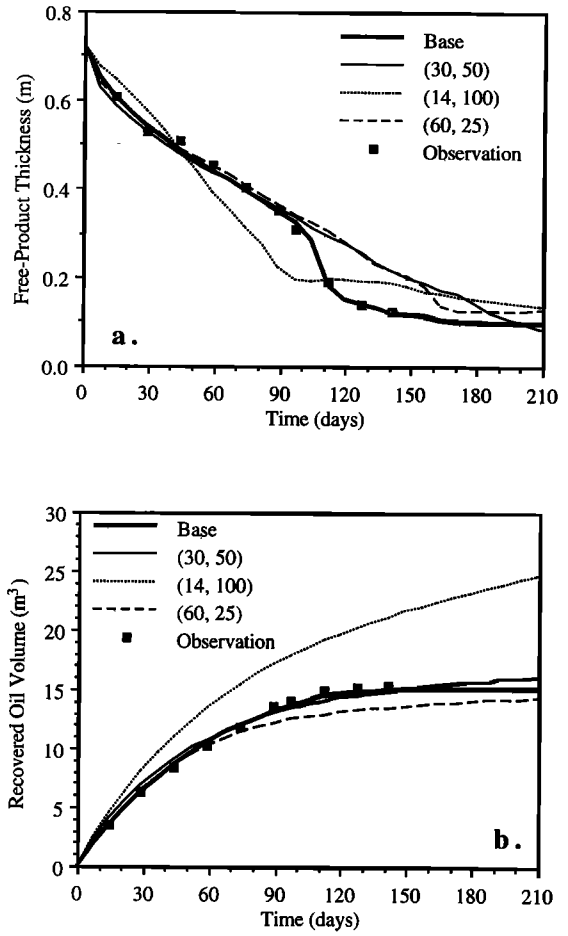

Figure 8. Accuracy of the ANN-GA method for the heterogeneous base case demonstrated by comparing to different observations. Solid line shows the base case, and symbols refer to the observation values used in the estimation. Other lines refer to the different combinations of (N, G): (a) free-product thickness at monitoring well 1 and (b) recovered volume.

150 patterns selected at random appear reasonable for solving the present problem.

The final analysis was performed to investigate the overall assessment of the ANN-GA optimization from the entire 138 testing patterns and to provide additional confidence on the base problem results. In this case, each testing pattern represents a given problem, and each of these problems was solved using ANN-GA with (N, G) = (30, 50). As the testing set has 138 patterns, it is beyond the scope of this paper to present all these solutions. As such, four statistically meaningful solutions are presented. After solving the 138 patterns, the objective function of the solutions was ranked from the best (minimum) to the worst (maximum); the best solution represents a problem with maximum adaptability to the present ANN-GA technique, while the worst value represents a problem with least adaptability. The objective values of the first 137 ranked patterns ranged from 0.006 (rank 1) to 0.882 (rank 137) with a median of 0.213 (rank 70). The worst pattern had a objective value of 2.217 (rank 138). Owing to this jump from 0.882 to 2.217, this pattern was considered and can be defined as an outlier. All four problems were solved again using the three combinations of (N, G), and the best combination that matched the observed V_o from visual inspection was selected as the optimal solution. Table 7 summarizes these solutions together with the desired or base solutions. The predicted V_{oI} matched well with the observed V_{oI} for all cases except the outlier case. The results show that ANN-GA may have the difficulty in solving such an outlier. However, the probability of such an outlier occurring is small; this probability is 0.72% for the present case. The overall results of this analysis showed the applicability of the ANN-GA optimization in estimating parameters by satisfying the minimum error criterion.

6. Conclusions

ANN regression and ANN-GA optimization appear to be simple and robust techniques for solving groundwater inverse problems related to free-product migration and recovery. A small subset of the responses to hydrogeologic conditions needs to be simulated using the numerical multiphase flow model, and these simulations may be inversely used to estimate the unknown hydrogeologic parameters from the observed responses. However, these techniques have limitations, and the following conclusions may help to understand and to overcome these limitations.

1. ANN regression and ANN-GA optimization may be grouped as indirect methods for solving inverse groundwater problems; they are applicable for point observations, and they do not require differentiation of the observations. Also, the algorithm for these techniques is simple compared to traditional techniques.

2. If an inverse function exists, ANN regression may be applied to approximate the function. In solving homogeneous problems, the technique estimated true solutions with ±15% error. In solving heterogeneous problems, the technique appeared poor. However, the error in training was insignificant, while the error in testing was significant. ANN regression may be used to solve heterogeneous problems but at an extensive computational effort.

3. If an objective function with error minimization is considered, ANN-GA optimization may be applied. In solving homogeneous problems, the technique was comparable to ANN regression and ANN-GA estimated true solutions with ±15% error. In solving heterogeneous problems, the technique failed to predict the true solutions; however, it searched acceptable solutions with respect to the error minimization. These solutions were searched using three combinations of N and G satisfying T = 1500 approximately. In a real problem the true solutions are unknown, and therefore the error minimization criterion appears acceptable.

4. As expected, the accuracy of parameter estimation increased with increased number of observations. In solving the homogeneous problems using three different types of observation data, the accuracy slightly increased from V_o only (10 observations) to H_o only (50 observations) to H_o and V_o combined (60 observations), and their errors were within ±15%.

5. In ANN development for ANN regression, a set of 100 patterns and *Hecht-Nielsen* [1987] suggestions appeared reasonable for solving homogeneous problems, while a set of 300 patterns appeared unreasonable for solving heterogeneous problems. In ANN development for ANN-GA optimization, the sets of 100 and 300 patterns together with *Hecht-Nielsen* [1987] suggestions appeared reasonable for solving both homogeneous and heterogeneous problems.

Table 7. Ranked Solutions of Different Training Patterns Using ANN-GA Optimization for the Heterogeneous Case

		Minimum		Median		Maximum		Outlier		
Zone	Parameter	Base	GA	Base	GA	Base	GA	Base	GA	GA
	Pattern	...	78	...	87	...	32	...	101	101
	E^*	...	0.066	...	0.213	...	0.882	...	2.217	2.217
	Rank	...	1	...	70	...	137	...	138	138
	$\bar{f}$	...	0.5	...	0.5	...	0.5	...	0.5	0.5
	N	...	14	...	60	...	30	...	60	30
	G	...	100	...	25	...	50	...	25	200
	T	...	1414	...	1560	...	1530	...	1560	6030
1	α	4.14	4.20	8.57	8.70	9.94	9.03	3.77	2.80	3.86
2	α	2.42	4.34	7.06	5.53	2.28	2.87	9.85	9.46	8.57
3	α	8.70	8.03	2.94	3.38	2.12	5.62	9.25	5.03	9.70
4	α	7.45	6.24	1.69	3.80	1.79	3.50	3.56	5.47	5.72
5	α	7.43	9.46	5.81	3.49	8.59	6.69	2.04	3.68	4.84
1	K_{sw}	26.9	23.0	25.7	14.9	19.9	29.4	8.7	2.1	3.9
2	K_{sw}	11.1	10.2	15.5	5.4	2.0	0.8	27.7	27.8	29.6
3	K_{sw}	18.7	13.8	27.8	29.3	9.7	23.5	11.7	22.6	3.3
4	K_{sw}	9.5	15.8	14.7	16.7	6.2	18.0	23.7	8.0	14.9
5	K_{sw}	27.4	12.2	11.6	22.8	28.6	29.6	2.2	4.4	3.6
	N^*	...	7	...	9	...	8	...	43	29
	G^*	...	48	...	24	...	22	...	24	96
	E^*	...	0.11	...	0.21	...	0.91	...	2.28	1.69
	V_{oR}	149.7	167.5	154.7	135.1	155.0	160.4	130.9	114.2	159.7

References

Beale, R., and T. Jackson, *Neural Computing: An Introduction*, Adam Hilger, Techno House, Bristol, England, 1991.

Blake, S. B., and R. W. Lewis, Underground oil recovery, in *Proceedings of Second National Symposium on Aquifer Restoration and Groundwater Monitoring*, Natl. Water Well Assoc., Dublin, Ohio, 1982.

Chavant, G., Identification of functional parameters in partial differential equations, in *Identification of Parameters in Distributed Systems*, edited by R. E. Goodson and M. Polis Edition, pp. 31–48, United Eng. Cent., Am. Soc. of Mech. Eng., New York, 1974.

Cooper, G. S., Jr., R. C. Peralta, and J. J. Kaluarachchi, Stepwise pumping approach to improve free phase light hydrocarbon recovery from unconfined aquifers, *J. Contam. Hydrol.*, *18*, 141–159, 1995.

DeJong, K. A., An analysis of the behavior of a class of genetic adaptive systems, Ph.D. dissertation, Univ. of Mich., Ann Arbor, 1975.

Emsellem, Y., and G. de Marsily, An automatic solution for the inverse problem, *Water Resour. Res.*, *7*(5), 1264–1283, 1971.

Environmental Systems and Technologies, Inc., ARMOS version 5.1: Areal multiphase organic simulator for free-phase hydrocarbon migration and recovery, Blacksburg, Va., 1994.

Goldberg, D. E., *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Reading, Mass., 1989.

Hecht-Nielsen, R., Kolmogorov's mapping neural network existence theorem, in *First IEEE International Joint Conference on Neural Networks, San Diego, CA*, Inst. of Electr. and Electron. Eng., New York, 1987.

Holland, J. H., *Adaptation in Natural and Artificial Systems*, Univ. of Mich. Press, Ann Arbor, 1975.

Kaluarachchi, J., and R. Elliott, Design factors for improving the efficiency of free-product recovery systems in unconfined aquifers, *Ground Water*, *33*(6), 909–916, 1995.

Kaluarachchi, J. J., J. C. Parker, and R. J. Lenhard, A numerical model for areal migration of water and light hydrocarbon in unconfined aquifers, *Adv. Water Resour.*, *13*, 29–40, 1990.

Lenhard, R. J., and J. C. Parker, Estimation of free hydrocarbon volume from fluid levels in observation wells, *Ground Water*, *28*, 57–67, 1990.

McCulloch, W. S., and W. Pitts, A logical calculus of the ideas imminent in nervous activity, *Bull. Math. Biophys.*, *5*, 115–133, 1943.

Neuman, S. P., Calibration and distributed parameter groundwater flow models viewed as a multiple-objective decision process under uncertainty, *Water Resour. Res.*, *9*(4), 1006–1021, 1973.

NeuralWare, Inc., *Neural Computing, NeuralWorks Professional II/Plus and NeuralWorks Explorer*, 360 pp., Pittsburgh, Pa., 1991.

Parker, J. C., R. J. Lenhard, and T. Kuppusamy, A parametric model of constitutive properties governing multiphase flow in porous media, *Water Resour. Res.*, *23*, 618–624, 1987.

Parker, J. C., and R. J. Lenhard, Vertical integration of three phase flow equations for analysis of light hydrocarbon plume movement, *Transp. Porous Media*, *5*, 187–206, 1989.

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

Rogers, L. L., F. U. Dowla, and V. M. Johnson, Optimal field-scale groundwater remediation using neural networks and the genetic algorithm, *Environ. Sci. Technol.*, *29*(5), 1145–1155, 1995.

Rumelhart, D. E., J. L. McClelland, and the PDP Research Group, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, vol. 1, MIT Press, Cambridge, Mass., 1986.

Sawyer, C. S., L. E. K. Achenie, and K. K. Lieullen, Estimation of the aquifer hydraulic conductivities: A neural network approach, Proceedings of the IAHS Conference on Models for Assessing and Monitoring Groundwater Quality, Boulder, CO, *IAHS Publ. 227*, 1995.

van Genuchten, M. T., A closed-form equation for predicting the hydraulic conductivity of unsaturated soils, *Soil Sci. Soc. Am. J.*, *44*, 892–898, 1980.

Willis, R. L., and W. W.-G. Yeh, *Groundwater Systems Planning and Management*, Prentice-Hall, Englewood Cliffs, N. J., 1987.

Yeh, W. W.-G., Review of parameter identification procedures in groundwater hydrology: The inverse problem, *Water Resour. Res.*, *22*(2), 95–108, 1986.

J. J. Kaluarachchi and J. Morshed, Department of Civil and Environmental Engineering, Utah State University, Logan, UT 84322-8200. (e-mail: jkalu@cc.usu.edu)

(Received March 17, 1997; revised December 23, 1997; accepted December 29, 1997.)