WATER RESOURCES RESEARCH, VOL. 35, NO. 8, PAGES 2415–2425, AUGUST 1999

Comparison of a genetic algorithm and mathematical programming to the design of groundwater cleanup systems

Alaa H. Aly
Utah State University Research Foundation, Logan

Richard C. Peralta
Department of Biological and Irrigation Engineering, Utah State University, Logan

Abstract. We present and apply a new simulation/optimization approach for single- and multiple-planning period problems in groundwater remediation. Instead of the traditional control locations for contaminant concentrations, we use an L_∞ norm as a global measure of aquifer contamination (CMAX). We use response-surface constraints to represent CMAX within the optimization model. We compare the performance of formal mixed integer nonlinear programming and a genetic algorithm for several optimization scenarios.

1. Introduction

A common means of containing and/or remediating contaminated groundwater aquifers is to extract contaminated water and treat it at the surface. This is known as pump and treat (P&T). Although several alternative remediation technologies have been utilized recently, no technology has proven superior to P&T for large plume problems [*Mercer et al.*, 1990; *Hoffman*, 1993; *Marquis and Dineen*, 1994].

P&T systems are usually employed to control contaminated groundwater migration and/or to achieve aquifer cleanup. In both situations basic design variables are well locations and pumping schedules.

P&T system design is an important topic because well locations and pumping rates can affect system performance significantly. Many studies have reported coupling optimization techniques with groundwater flow and transport simulation for designing P&T systems [e.g., *Gorelick et al.*, 1984; *Marryott et al.*, 1993; *Rogers and Dowla*, 1994; *McKinney and Lin*, 1995; *Peralta et al.*, 1995; *Xiang et al.*, 1995].

Early studies used first-order approximation of the groundwater flow equation to formulate a linear optimization problem [*Atwood and Gorelick*, 1985; *Peralta and Ward*, 1991]. However, contaminated groundwater management required the use of nonlinear optimization. *Gorelick et al.* [1984] used a contaminant transport simulation model within a robust nonlinear optimization algorithm. They evaluated the derivatives numerically using forward differences in earlier iterations and central differences in later iterations.

Numerical evaluation of the derivatives requires many computations. For large-scale problems the computational burden can be prohibitive. In order to make the optimization problem computationally tractable for large problems, *Ahlfeld et al.* [1988] applied sensitivity theory to the solute transport equation to evaluate the derivatives more efficiently.

Nonlinear programming techniques cannot guarantee global optimality when applied to large nonconvex groundwater management problems. For real problems where the time required to simulate the groundwater system is significant, nonlinear programming methods may need prohibitive amounts of CPU time.

Paper number 1998WR900128.
0043-1397/99/1998WR900128$09.00

The limitations of mathematical programming have motivated some researchers to use simplified expressions inside the optimization model. Several simple functions have been used in groundwater simulation/optimization (S/O) models [*Alley*, 1986; *Lefkoff and Gorelick*, 1990; *Ejaz and Peralta*, 1995; *Cooper et al.*, 1998].

More recent studies investigated the use of alternative optimization techniques such as simulated annealing [*Rizzo and Dougherty*, 1996] and genetic algorithms [*McKinney and Lin*, 1994; *Ritzel et al.*, 1994; *Rogers and Dowla*, 1994]. In this study we combine the response surface method with either mathematical programming or a genetic algorithm.

An attractive feature of the genetic algorithm (GA) is that it does not require the continuity or differentiability of the objective function. Below, we exploit this feature and contrast the GA solution to a formal mathematical programming solution. The intent is to compare the performance of the GA and mathematical programming for groundwater remediation problems.

The proposed methodology employs flow and transport simulation models externally to the optimization. As a result, the presented techniques are independent of the specific flow and transport simulators used. This allows using special-purpose codes or newly developed simulation codes for design purposes. Moreover, the presented formulation permits time-varying management priorities and restrictions.

Manuscript organization is as follows. In section 2 we formulate the management problem and describe the selected functional form used to describe the response surfaces. We also describe the robust regression technique used to evaluate the coefficients of that function. In section 3 we describe the study area and outline tested scenarios. In section 4 we show simple cases of the optimization problem and develop the response surfaces used in the optimization model. In sections 5 and 6 we describe the genetic algorithm and the mathematical programming techniques used to solve the optimization problem. Then we contrast results from the two approaches and summarize findings.

2. Optimization Problem Formulation

Consider an aquifer contaminated with a dissolved contaminant. A P&T system is to be designed for a treatment facility of specified flow capacity. The goal is to determine the best

Table 1. Cost Function Coefficients, Second Formulation

Coefficient	Value
WT_1	1.00
WT_2, WT_3 (3 years, 5%)	2.7232
WT_2, WT_3 (2 years, 5%)	1.8594
C^{IP} (installation cost)	11,900 \$/well
C^{pq} (pumping cost)	7.7324E-4 \$ per foot^4/d year
C^p (treatment cost)	3.55 \$ per foot^3/d year

One foot^3/d = 0.0283 m^3/d; 1 foot = 0.3047 m.

pumping schedules for M^p possible wells at prescribed locations.

We approximate the multiple-period planning problem using a series of single-period problems. Each single period can be simulated using either steady state or transient conditions. The results of implementing the optimal strategy of one planning period are used as initial conditions for the next planning period. This myopic stepwise optimization greatly simplifies the analysis and has been demonstrated in other groundwater management studies [*Ahlfeld*, 1990; *Rizzo and Dougherty*, 1996]. However, this approach might produce a less optimal solution than a fully dynamic approach, as indicated by *Ahlfeld* [1990].

The following optimization problem formulations describe a single planning period but they can address multiple-period problems. Notice that the objective function and the constraints can change from one planning period to another.

Formulation 1: The goal is to minimize the largest concentration remaining in the aquifer at the end of a single planning period (CMAX) while satisfying system and/or budget constraints. One constraint is used to prevent total pumping from exceeding the treatment facility's flow capacity (P^{MAX}). Another constraint forces total extraction to equal total injection. Because extraction and injection have different signs, forcing total extraction to equal total injection is equivalent to reinjecting all extracted water (after treatment). This constraint is only used when injection rates are computed by the optimization model.

Minimize CMAX subject to

$$p^L(\hat{e}) \le p(\hat{e}) \le p^U(\hat{e}) \qquad \hat{e} = 1, 2, \cdots, M^p \tag{1}$$

$$\sum_{\hat{e}=1}^{M^{ex}} |p(\hat{e})| \le P^{MAX} \tag{2}$$

$$\sum_{\hat{e}=1}^{M^p} p(\hat{e}) = 0 \tag{3}$$

$$\text{CMAX} = f_c(p(1), p(2), \cdots, p(M^p)) \tag{4}$$

where CMAX is the maximum concentration in the aquifer at the end of the planning period [M L^{-3}]; M^{ex} is the number of extraction wells; $p(\hat{e})$ is the steady pumping rate at location $\hat{e}$ [L^3 T^{-1}]; $p^L(\hat{e})$ and $p^U(\hat{e})$ are lower and upper bounds on pumping rate at location $\hat{e}$ [L^3 T^{-1}]; and P^{MAX} is the maximum allowed pumping from all extraction wells, usually equal to the treatment facility's flow capacity [L^3 T^{-1}].

Formulation 2: The goal is to find the pumping strategy that has the lowest cost while achieving aquifer cleanup by the end of the planning period. Aquifer cleanup is achieved by specifying a target CMAX value at the end of each planning period. The total cost objective function is mixed-integer nonlinear (equation (5)).

The first objective function component is the well installation cost. This cost is incurred once at most and only if a well is used for pumping. This is a discrete operation and requires the use of binary variables in the optimization model. The installation cost is zero for any well that has pumped in any previous period.

The second component in the cost function is the pumping cost. This cost is a function of the pumping rate and hydraulic lift. This term is quadratic because the head at the well is represented as a linear function of pumping rates (as explained in the next section). The third component is the treatment cost. For a specific treatment facility, treatment cost is considered linearly proportional to pumping volume. This term is linear in the pumping rates.

Minimize

$$\begin{aligned} \text{PW} = \text{WT}_1 \sum_{\hat{e}=1}^{M^p} C^{IP}(\hat{e})\text{IP}(\hat{e}) \\ + \text{WT}_2 \sum_{\hat{e}=1}^{M^{ex}} C^{pq}(\hat{e})p(\hat{e})(\text{TELEV} - h(\hat{e})) \\ + \text{WT}_3 \sum_{\hat{e}=1}^{M^{ex}} C^p(\hat{e})p(\hat{e}) \end{aligned} \tag{5}$$

subject to

$$h(\hat{e}) = f_h(p(1), p(2), \cdots, p(M^p)) \tag{6}$$

$$\begin{aligned} \text{IP}(\hat{e}) = 1 \qquad |p(\hat{e})| > 0 \\ \text{IP}(\hat{e}) = 0 \qquad |p(\hat{e})| = 0 \end{aligned} \tag{7}$$

$$\text{CMAX} \le C^T \tag{8}$$

$$\text{CMAX} = f_C(p(1), p(2), \cdots, p(M^p)) \tag{9}$$

$$p^L(\hat{e}) \le p(\hat{e}) \le p^U(\hat{e}) \qquad \hat{e} = 1, 2, \cdots, M^p \tag{10}$$

$$\sum_{\hat{e}=1}^{M^{ex}} |p(\hat{e})| \le P^{MAX} \tag{11}$$

$$\sum_{\hat{e}=1}^{M^p} p(\hat{e}) = 0 \tag{12}$$

where PW is total present worth of the P&T operation including well installation, pumping, and treatment costs [\$]; WT_1, WT_2, and WT_3 are factors used to convert the well installation, operational pumping, and treatment costs, respectively, into present values [\$ per \$] (usually $WT_1 = 1$); IP($\hat{e}$) is an indicator variable for pumping at location $\hat{e}$; $C^{IP}(\hat{e})$ is cost of installing a well at location $\hat{e}$ [\$ per well]; $C^{pq}(\hat{e})$ is cost of pumping water from the aquifer using the extraction well at location $\hat{e}$ [\$ per L^4 T^{-1}]; $C^p(\hat{e})$ is cost of treating contaminated groundwater from well at location $\hat{e}$ [\$ per L^3 T^{-1}]; $h(\hat{e})$ is groundwater head at location $\hat{e}$ [L]; TELEV is inlet elevation of the treatment facility [L]; C^T is target contaminant concentration at end of planning period (usually MCL). Cost coefficient values are listed in Table 1.

Gorelick [1983] describes two techniques for defining the functions f_h (equation (6)) and f_C (equations (4) and (9)) within an optimization model. According to *Gorelick* [1983], in the "embedding method," finite difference or finite element approximations of the governing groundwater flow equations are treated as part of the constraint set of a linear programming model [*Gorelick et al.*, 1984; *Peralta et al.*, 1995; *Gharbi and Peralta*, 1994; *Takahashi and Peralta*, 1995]. This definition can be extended to include optimization models that use full simulation models to evaluate the state variables [e.g., *McKinney and Lin*, 1995].

The other technique described by *Gorelick* [1983] is the "response matrix" approach. In this approach an external groundwater simulation model is used to develop unit responses. This definition can also be extended to include using simulations to fit approximation functions. These approximation functions can be derived using either Taylor series or curve fitting methods. When a first-order Taylor series is used, this approach is known as the response matrix method. More generally, this approach can be considered a response surface (RS) method.

The embedding method can sometimes be more accurate and provides more potential for controlling the physical system [*Peralta et al.*, 1991]. However, an optimization problem formulated using this method is nonlinear, nonconvex, and very large. For such problems the computational effort required to find an optimal solution can be prohibitive. A promising remedy for this problem is to use algorithms that can take advantage of parallel processors [*McKinney and Lin*, 1994]. *Rogers and Dowla* [1994] suggested another remedy. They used an artificial neural network in conjunction with a GA to reduce the computational effort for a groundwater remediation problem.

The RS method generally yields a fairly simple optimization problem. Usually, little effort is required to incorporate the constraints within optimization algorithms. Another RS advantage is that the flow and transport simulations can be recycled. For example, if more accuracy is desired in a given solution space neighborhood, more simulations can be performed in that neighborhood and the results can be used along with earlier simulations. A third RS advantage is the ease of running needed simulations in parallel or even on separate CPUs. Together, these advantages can result in significant CPU and real time savings. In this study, using the RS approach made it easy to find the best set of control parameters for the GA (crossover and mutation probabilities and population size). The response surface must be found for each planning period. In other words, the RS for the second planning period is constructed using the optimal results from the first planning period as initial conditions.

Few forms have been suggested in the literature for representing contaminant concentrations as a function of pumping rates. *Alley* [1986] found that simple linear regression provided sufficient accuracy for predicting solute concentrations for the tested problem. However, in our study, simple linear regression was inadequate for representing CMAX as a function of pumping rates.

Lefkoff and Gorelick [1990] used regression to approximate salt mass transport and found that this has greatly simplified the analysis. However, they did not show the functional form used. *Cooper et al.* [1998] represented light nonaqueous phase liquid mass via regression in their groundwater S/O model. In this study we found that a polynomial function with second-order interaction terms accurately approximated CMAX.

In the following sections we construct the function f_C using a robust estimation technique (summarized later) and f_h using a first-order Taylor series. We generate data for the regression from numerous groundwater flow and transport simulations.

2.1. The Approximation Function

Desirable properties for the approximation function are the following: (1) It must be adequately accurate in the decision space neighborhood of interest, (2) it should be easy to use, and (3) it should have continuous derivatives. The last property is desirable for gradient-based mathematical programming algorithms.

We used polynomial functions with two-way interaction terms to represent the response variable (CMAX). The general form of the polynomial function is

$$\mathrm{CMAX} = \beta_0 + \sum_{i=1}^{MP} \sum_{j=1}^{MP} \beta_{i,j} |p(i)|^{\alpha_{i,j}} |p(j)|^{\gamma_{i,j}} \qquad (13)$$

Higher-order interaction terms were not needed for all tested scenarios.

The exponents in the above polynomial are usually different from unity. This means that this polynomial is not simply a quadratic approximation. Rather, the approximation function can represent nonlinear gradients accurately.

To determine the coefficients and exponents for the polynomial function, we used a two-step regression approach. First, we solve a nonlinear regression problem using iteratively reweighted least squares (IRWLS; described in the following section) to determine both the coefficients and exponents. In the second step we fix the exponents and solve a linear regression problem using IRWLS to find the coefficients. In essence, the first step finds the best polynomial transformation of the explanatory variables (pumping rates) and two-way interaction terms. The second step uses that transformation and solves a linear regression problem.

2.2. Robust Regression

Regression analysis is often used to find coefficients of approximating functions. Unfortunately, outliers that appear to conflict with the model can arise and control the computed regression coefficients [*Draper*, 1981]. A robust regression technique will change the computational scheme adaptively to prevent outliers from controlling the computed regression equation. We used IRWLS, which can be summarized as follows:

1. Fit an initial regression equation using a robust regression algorithm such as minimizing the maximum absolute deviation.

2. Compute the residuals (defined as observed minus predicted values of the response variable). Use the residuals to compute weights for the data set. Generally, weights are inversely proportional to the magnitude of the residuals.

3. Fit a weighted least squares regression equation with the weights computed in step 2.

4. If the difference between the estimates of the regression coefficients is larger than desired, go to step 2. Otherwise, stop.

Staudte and Sheather [1990] show that the computed regression coefficients depend on the initial estimator (used in step 1). Therefore it is desirable to use a robust technique for that step. In this study we used a minimum maximum absolute residual criterion (instead of the ordinary least squares criterion).

For optimization problem formulation 1 (equations (1)–(4)) we are minimizing the CMAX resulting after a specified time period. Therefore the solution is generally to pump a total of

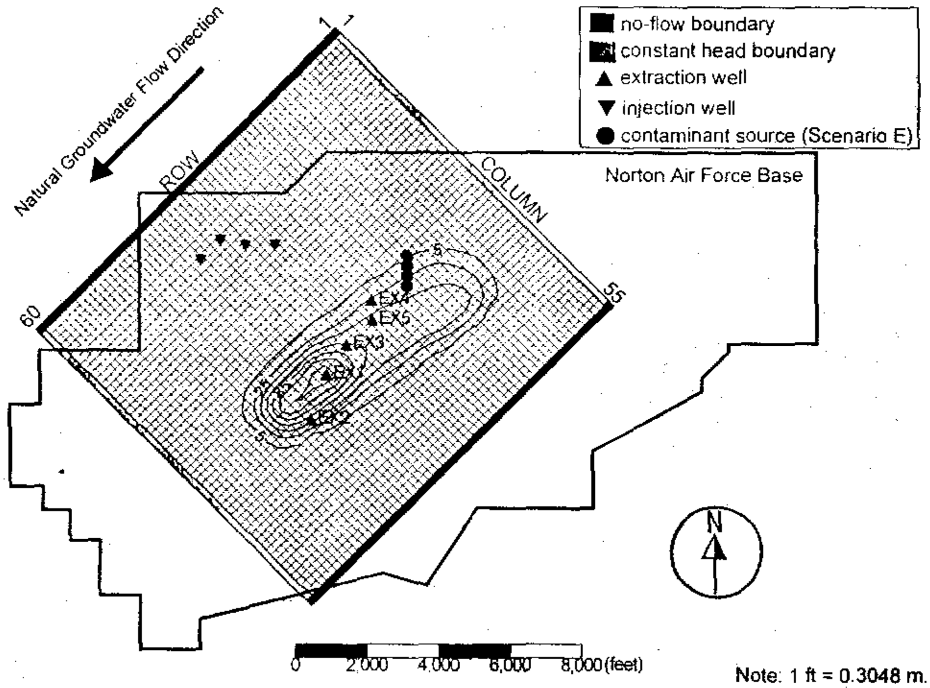

Figure 1. Base boundary, finite difference grid, boundary conditions, well locations, and initial TCE concentrations in parts per billion.

P^{MAX} from all wells. In other words, the solution space is limited to sets of pumping values whose sum is P^{MAX}. Therefore when the data are generated for the regression, we can limit pumping value sets to those whose sum is P^{MAX}. This restriction improved the regression fit for all tested scenarios.

3. Site and Scenarios Description

Norton Air Force Base (NAFB) is located in the San Bernardino Valley, part of the California Peninsular Range geomorphic province. Near NAFB, several groundwater-bearing zones exist. The top layer contains dissolved trichloroethylene (TCE), which is moving with the groundwater. To speed TCE plume cleanup, NAFB has installed a P&T system. This 200-gallons/min (gpm; 760 L/min) P&T system is to be augmented to extract more contaminated groundwater. In the following sections we consider capacities up to 2000 gpm (7600 L/min) in order to achieve aquifer cleanup to maximum contamination limit (MCL). The MCL for TCE is 5 ppb.

The MODFLOW groundwater flow simulation model [*McDonald and Harbaugh*, 1988] has been calibrated to the study area [*EA Engineering, Science and Technology*, 1994]. MT3D [*Zheng*, 1990] is used to simulate plume migration for alternative preliminary well locations and pumping strategies. The finite difference grid has 60 rows and 55 columns. The groundwater aquifer is modeled as a confined aquifer with transmissivities ranging between 0.009 and 0.014 m^2/s, a longitudinal dispersivity of 30.50 m, and a transverse dispersivity of 3.05 m.

Injection well locations have been specified along pipelines. In the following sections we consider five potential extraction wells. One of the extraction wells is already operating (EX1 in Figure 1). Therefore, while the optimal pumping rate for this well is computed, the cost coefficient (C^{IP}) for installing this well is zero.

We develop optimal pumping strategies for five scenario families (A–E). In each family, the first scenario (A1, B1, etc.) uses optimization formulation 1 and the second scenario (A2, B2, etc.) uses optimization formulation 2. Each optimization problem is solved using mathematical programming and a GA.

Table 2. Scenario Families Considered for Mathematical Programming and the Genetic Algorithm Comparison

	Scenario Family				
	A	B	C	D	E
Treatment facility size (P^{MAX} in gpm)	800	2000	2000	2000	2000
Number of considered wells (M^{P})	2	3	5	9	5
Extraction wells used (Figure 1)	EX1, EX2	EX1–EX3	EX1–EX5	EX1–EX5	EX1–EX5
Compute optimal injection rates	no	no	no	yes	no
Number of planning periods	1	1	1	1	2
Number of simulations	153	150	279	350	249, 249

One 1 gpm = 5.4504 m^3/d. The number of simulations is that required to estimate the coefficients of the response surface for each scenario. The target concentration is 5 ppb (MCL for TCE).

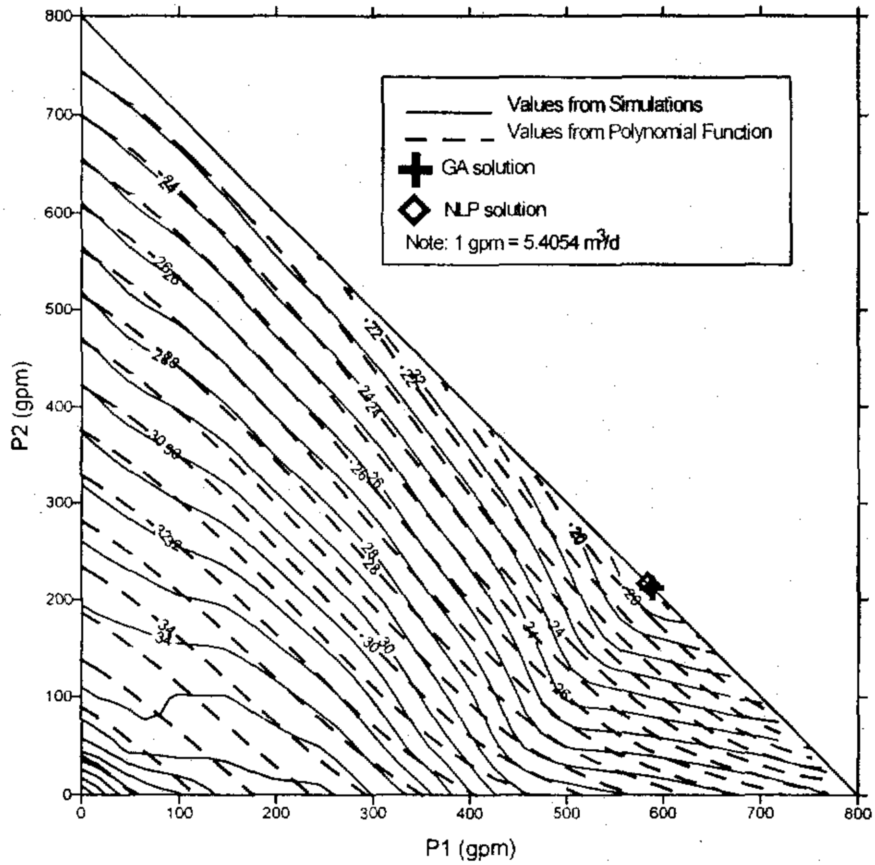

Figure 2. Contours of CMAX for scenario A1.

Table 2 summarizes scenario assumptions and the number of simulations required to construct the RS. In the A, B, C, and E scenario families, extraction rates are computed and injection rates are fixed.

In the E scenarios we consider two planning periods. In the first 2-year period, continuous leaching of contaminant from the vadose zone to the aquifer is present. Leachate concentrations and amounts are based on field data [*EA Engineering, Science and Technology*, 1994]. In the second 1-year period, no contaminant leaching is present. Although we use the same management goals for both periods, the presented methodology also permits changing management requirements for different planning periods.

4. The Response Surface

Rather than including detailed simulation expressions within mixed-integer nonlinear (MINLP) or GA problems, we represent system response to pumping using simple approximation (response surface) functions. In this section we investigate the shape of the RS. We also show how close the approximating function is to the actual surface in the neighborhood of interest.

We first investigate the case of two extraction wells and four injection wells. Injection rates are fixed. We study only the effect of changing the extraction rates on CMAX (scenario A1). For each combination of the two extraction rates we use the flow and transport simulation models to compute CMAX at the end of the planning period. Figure 2 shows the results and the contours of the best polynomial approximating function (found using robust regression). The solid lines in Figure 2 are based on 153 simulations. The pumping rates for these simulations are selected at random in the solution space.

In Figure 2 the minimum CMAX occurs when total extraction from the two wells equals P^{MAX} (along the diagonal line in figure 2). If extraction wells are near areas of high concentration, we would expect concentrations to drop as total extraction increases. This intuitive result is important because it implies that for subsequent cases (with more potential wells) we only need to consider combinations of pumping rates that total P^{MAX}. This will greatly reduce the number of simulations required to construct the RSs. It will also make the approximating functions more accurate since we will consider a much smaller subspace of the decision space. If this assumption is not used, we expect the number of simulations required to fit the polynomials to grow by a factor of at least 2.

Another feature is more easily observed by examining Figure 3, which shows the results for the combinations of extraction rates for which total extraction equals P^{MAX}. There is only one global minimum (at P1 = 600) and one local minimum (at P1 = 0). Also, the approximating function is at its minimum at almost exactly the same location as the RS.

For the case of 3 extraction wells and 4 injection wells (with fixed injection rates) we study the effect on CMAX of changing the extraction rates (scenario B1). To be able to visualize the results, we consider only pumping sets that total P^{MAX}. Figure 4 shows the CMAX resulting from simulations and contours of the best polynomial approximating function.

In Figure 4 the approximating function does not fit the data

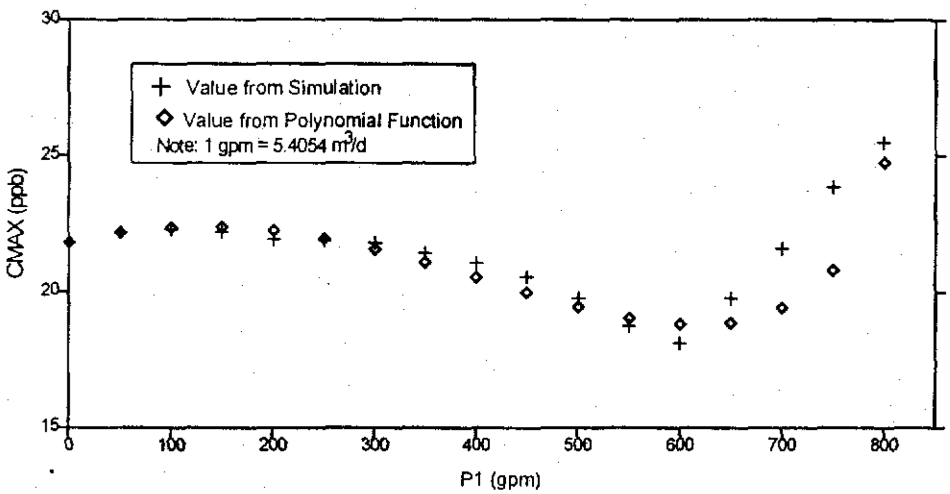

Figure 3. Observed and predicted CMAX versus P1 (P1 + P2 = 800 gpm, or 3000 L/min).

as well as Figure 2. However, the fit is still acceptable. Notice the obvious global minimum and the flat area on the response surface around the minimum point. This shows that there is a large region of nearly optimal solutions. Any solution in that region will result in a CMAX value that is very close to the smallest achievable CMAX. Table 3 shows the polynomial coefficients and exponents for scenarios A and B.

5. The Genetic Algorithm

GAs are heuristic rules for searching a solution space to identify the best solution. A solution determined using a GA is not necessarily optimal. It is merely the best solution identified. The use of GAs was first suggested by *Holland* [1975], who based his search on a survival-of-the-fittest rule. Since then,

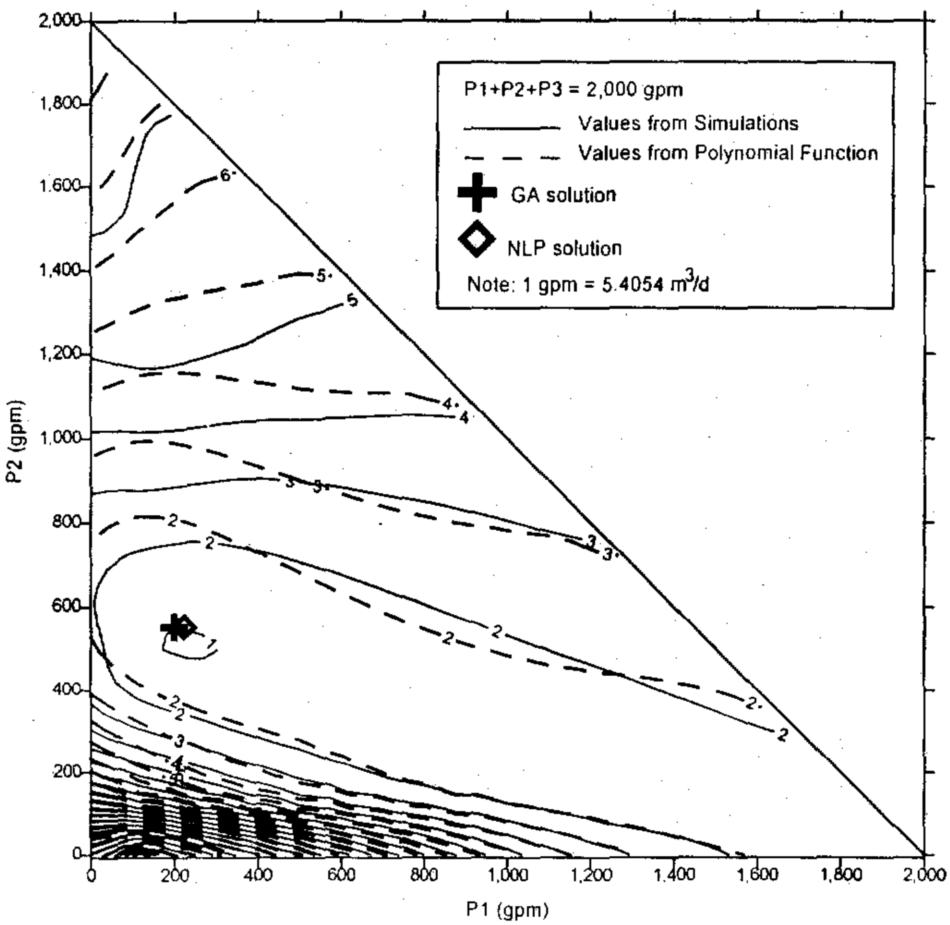

Figure 4. Contours of CMAX for scenario B1.

GAs have been used in many disciplines [*Davis*, 1991; *Goldberg*, 1989].

In groundwater management, GAs have been used by *McKinney and Lin* [1994], *Ritzel et al.* [1994], *Rogers and Dowla* [1994], *Cieniawski et al.* [1995], and others. In this paper we focus on how the GA is implemented to address the problem at hand.

The major advantage of GAs is that they are independent of the particular problem being analyzed. The only requirement is an objective (fitness) function indicating system performance. This function can be nonlinear, nondifferentiable, or discontinuous. A GA requires only that system performance can be evaluated for any set of the decision variables. In formulation 1 the fitness value is the reciprocal of CMAX. Therefore the GA tries to find the pumping rates that will result in the smallest CMAX. In formulation 2 the fitness is the reciprocal of total cost.

We used a GA with the basic reproduction, crossover, and mutation operators. The GA used is very similar to the simple genetic algorithm (SGA) of *Goldberg* [1989]. The only difference is that we use tournament selection [*Goldberg*, 1990] instead of the roulette-wheel selection of the SGA.

One problem with GAs is that they do not provide an explicit method to handle constraints. Instead of explicitly considering constraints, penalty terms are added to the objective (fitness) function. In formulation 1 a single constraint limits total pumping. A simple method to handle such a constraint in a GA is to assign a very low fitness value for any set of pumping rates whose sum exceeds the upper bound on total pumping. In all tested problems, after few iterations the GA hardly tries to evaluate the fitness value for any set of pumping rates whose sum exceeds P^{MAX}.

In formulation 2 we added an adaptive penalty term to the total cost to handle the more complex constraint on CMAX. This adaptive penalty term adds a large cost to any set of pumping rates that result in CMAX greater than the prescribed cleanup value (Figure 5). Each unit of CMAX greater than cleanup value has a cost that is 2 orders of magnitude larger than total economic cost. This makes a pumping strategy with less total cost more favorable than another with a larger total cost even if neither achieves the required cleanup by the same amount. This method was very effective and gave better answers than the nonstationary penalty function of *Joines and Houck* [1994], which increases the penalty function as the generation number increases. The number of pumping rate sets that do not achieve acceptable CMAX values was very small after 10–25 generations.

The methodology proposed herein differs from that of *McKinney and Lin* [1994] in that we use an RS approach inside the optimization model while McKinney and Lin used an embedding approach. Using the RS approach reduced the computational burden significantly. It also allowed us to find the best set of control parameters for the GA (population size, crossover probability, and mutation probability). *McKinney and Lin* [1994] implemented their GA on CM5 parallel computers with various numbers of processors. They used different crossover and mutation probabilities for the different problems addressed but offered no guidelines for selecting these probabilities.

Fitness Function

Input: pumping rates

Output: fitness value

if(sum of pumping rates > size) return (1.0)

CMAX = f_C(pumping rates)

PW = present worth of installation, pumping, and treatment costs (in millions of dollars)

if(*CMAX* > C^{clean}) *PW* = *PW**(1+100*(*CMAX*-C^{clean}))

return(1000.0/*PW*)

Note: *PW* ranges between 0.5 and 50.0.

Figure 5. Evaluation of fitness for second formulation.

We used binary coding wherein the pumping rate from each well is represented by L digits of the chromosome. For example, when we tried to optimize the pumping rates from five extraction wells, the chromosome length was $5L$. The chromosome length, L, is determined from the desired representation accuracy. For example if the pumping rate from one well can range between P^L and P^U and the desired accuracy is ε, then

$$L = \frac{\log\left(1 + \frac{|P^U - P^L|}{\varepsilon}\right)}{\log 2} \tag{14}$$

where the logarithm is taken to any base. For example, when P^U is 800, P^L is 0, and the required accuracy is 0.5, then the chromosome length is 11. If we have five such pumping rates, the final chromosome length is 55. Notice that different pumping rates can have different accuracy values if desired. Longer chromosomes can be used to the desired accuracy at the expense of more run time for the GA. We used $\varepsilon = 0.5$ gpm (1.9 L/min) for all scenarios. The pumping rates ranged between 0 and 800 for scenarios C, D, and E and between 0 and 1200 for scenarios A and B. Therefore L had a value of 11 for the former scenarios and 12 for the latter scenarios.

Control parameters selection greatly affects the answer computed by the GA. However, there are no published general guidelines for selecting these parameters. Many studies have attempted to evaluate parameter values that work well under a variety of conditions [*De Jong*, 1975; *Schaffer et al.*, 1989]. However, their results are problem specific and depend on how the GA is implemented. A major advantage of our proposed methodology is that the size of the study area affects only the time required to evaluate the response functions. Therefore, after the response functions are evaluated, the GA takes very little time to find the best set of pumping rates. This allowed us to use the GA for a very large number of control parameter selections.

Table 3. Polynomial Coefficients and Exponents for Scenarios A and B

Polynomial Coefficients (Equation (13))	Scenario A	Scenario B
β_0	38.886	0.2757
$\beta_{1,1}(\alpha_{1,1}, \gamma_{1,1})$	−0.790 (1.064, 0.000)	9.174 (0.035, 0.000)
$\beta_{2,2}(\alpha_{2,2}, \gamma_{2,2})$	−1.000 (1.036, 0.000)	0.345 (0.590, 0.000)
$\beta_{3,3}(\alpha_{3,3}, \gamma_{3,3})$	···	−0.232 (0.5021, 0.000)
$\beta_{1,2}(\alpha_{1,2}, \gamma_{1,2})$	−0.004 (0.811, 2.450)	−4.030 (0.248, 0.009)
$\beta_{1,3}(\alpha_{1,3}, \gamma_{1,3})$	···	0.001 (0.000, 2.776)
$\beta_{2,3}(\alpha_{2,3}, \gamma_{2,3})$	···	−0.001 (0.406, 2.481)

Pumping rates in the polynomial equation are scaled by dividing their magnitude by 10,000.

Table 4. Results for Scenarios A2 and B2

	A2		B2	
	GA	NLP	GA	MINLP
Optimal pumping rates, gpm				
EX1	620	617	817	837
EX2	180	183	785	933
EX3	···	···	0	60
Total pumping, gpm	800	800	1602	1830
Present worth of costs, 10^6 dollars	1.136	1.141	2.467	3.203

One gpm = 5.4504 m^3/d.

At least 60 sets of control parameters (population size, crossover probability, and mutation probability) were tested for each problem. The results indicate that the population size should be between 100 and 200. Our experience is that larger population sizes require extra time but do not affect the solution. However, if the number of wells is large or if only a relatively small subspace provides a feasible solution, then a larger population size might be needed.

In this study the best crossover and mutation probabilities are 0.7–1.0 and 0.06–0.08, respectively. Generally, a crossover probability less than 0.7 always provided an inferior answer. A mutation probability greater than 0.08 increased the number of infeasible evaluations without improving the final answer. The GA performed most poorly when the mutation probability was zero. This is expected since mutation prevents the GA from getting locked at local optima.

The previous discussion only provides general guidelines for selecting control parameters' values. The mentioned values should be used as a starting point and should be revised. Different values might result in better answers for other problems. When a response surface is used, little effort is needed in trying different sets of control parameters for a given problem.

6. Mathematical Programming

For formulation 1 the optimization problem has linear (equations (2) and (3)) and nonlinear (equation (4)) constraints. This is a nonlinear programming (NLP) problem for which several robust solvers are available [*Drud*, 1985; *Murtagh and Saunders*, 1987]. We used MINOS [*Murtagh and Saunders*, 1987]. MINOS has been used successfully for a wide range of groundwater management problems [e.g., *Cunha et al.*, 1993; *Gharbi and Peralta*, 1994; *Peralta et al.*, 1995; *Takahashi and Peralta*, 1995; *Matsukawa et al.*, 1991; *Reichard*, 1995].

For formulation 2 in addition to the linear and nonlinear constraints, the optimization model has binary variables, $IP(\hat{e})$ (equation (5)). The resulting optimization problem is a mixed-integer nonlinear (MINLP) optimization problem. Available MINLP solvers are not as reliable as those for NLP and other mathematical programming problems [*Viswanathan and Grossmann*, 1990]. We used the DICOPT++ solver developed at Carnegie Mellon University [*Kocis and Grossmann*, 1989; *Viswanathan and Grossmann*, 1990]. The MINLP algorithm inside DICOPT++ is based on the outer-approximation algorithm. DICOPT++ solves a series of NLP subproblems and MIP (mixed-integer programming) master problems. To solve the subproblems, DICOPT++ uses external optimization algorithms. In this study we used MINOS [*Murtagh and Saunders*, 1987] to solve the NLP subproblems and OSL [*IBM Corporation*, 1991] to solve the MIP master problems.

To be able to compare the GA results with those of NLP and MINLP, we tried both direct minimization as well as reciprocal maximization. We also used the constraints directly and as penalties added to the objective function (as done in the GA). For all tested problems NLP or MINLP found better answers by direct minimization when constraints were used directly. This is expected because using the reciprocal introduces unnecessary nonlinearity into the optimization problem. In the next section we report only the best answer found by NLP (or MINLP).

7. Results

Results for scenarios A and B are shown in Figures 2 and 4. For the NLP problem of scenarios A1 and B1, both the GA and NLP found the global minimum solution. However, for the MINLP problems of scenarios A2 and B2, the GA found a better solution than MINLP (Table 4). As explained below, the GA generally performed better than NLP and MINLP for all tested scenarios.

In scenario C1 the GA's minimum CMAX is 1.451 ppb, while CMAX for the NLP solution is 1.504 ppb. This indicates that the answer found using NLP is a local minimum. Similar results were found for scenarios D1 and E1. Tables 5 and 6 summarize the results for the C and E scenarios, respectively.

Figure 6 shows the contaminant concentration contours after the pumping strategies of scenario C1 are implemented. The difference between the two strategies is unclear. Although the GA resulted in a strategy with a lower value of CMAX, the NLP strategy required one less well and resulted in concentrations that are almost identical from a practical viewpoint.

The results shown in Figure 6 reflect a fact noted in the discussion of Figure 4. In Figure 4 there is a wide flat "valley" around the optimal solution. Although the pumping rates differed greatly in that valley, CMAX was essentially the same. A similar behavior is exhibited in Figure 6, where the pumping rates are different but the resulting concentrations are very similar. However, this is not the case for cost minimization for which MINLP and GA produced greatly different results.

Figure 7 shows how cost is accumulated over the planning period after the optimal strategies of scenario C2 are implemented. Over the entire planning period, the MINLP pumping strategy costs about 32% more than the GA's strategy.

In scenarios D1 and D2, where the injection rates were not fixed, the answers that were obtained were not better than the answers for scenarios C1 and C2. This was expected because

Table 5. Results for the C Scenarios

	C1		C2	
	GA	NLP	GA	MINLP
Optimal pumping rates, gpm				
EX1	608	693	656	1061
EX2	512	486	485	525
EX3	651	800	39	27
EX4	67	21	23	0
EX5	162	0	0	0
Total pumping, gpm	2000	2000	1203	1613
CMAX, ppb	1.451	1.504	···	···
Present worth of costs, 10^6 dollars	···	···	2.307	3.054

One gpm = 5.4504 m^3/d.

Table 6. Results for the E Scenarios

	E1		E2	
	GA	NLP	GA	MINLP
Optimal pumping rates, gpm				
EX1	1076, 1124	1199, 1230	685, 730	1197, 1245
EX2	252, 373	336, 456	272, 382	292, 375
EX3	78, 210	465, 314	0, 0	0, 0
EX4	12, 0	0, 0	0, 0	0, 0
EX5	582, 293	0, 0	388, 300	0, 0
Total pumping, gpm	2000, 2000	2000, 2000	1345, 1412	1489, 1620
CMAX, ppb	8.526, 2.673	9.459, 3.034	...	...
Present worth of costs, 10^6 dollars	...	...	1.628, 0.624	1.773, 0.905

One gpm = 5.4504 m^3/d. Each cell contains two values, for the first and second planning periods, respectively.

the fixed injection well locations are not close enough to change groundwater flow near the plume center.

For the GA the best answer was always obtained before generation 250. However, we terminated the GA after at least 500 generations for all tested problems. For a few problems we terminated the GA after 10,000 generations. This never improved the solution for any tested problem.

8. Summary and Conclusions

The GA performed as well as or better than mathematical programming (in terms of the objective's numerical value) for all tested problems when response functions were used for each. Only for the simplest problem was mathematical programming able to find the same answer as the genetic algorithm. Furthermore, since response functions dramatically reduce the computational effort compared to all embedded approaches, the GA approach with response functions is recommended for similar problems.

Other advantages of the GA include the simplicity of implementation, speed, and the simple incorporation of integer variables within the optimization problem. The best set of control parameters for the genetic algorithm was found informally by using several sets of control parameters. A population size of about 150, a crossover probability of about 0.85, and a mutation probability of about 0.08 resulted in the best answers for almost all tested problems within less than 300 generations. The use of the response surface (RS) to represent the simulation constraints allows selection of the best set of control parameters.

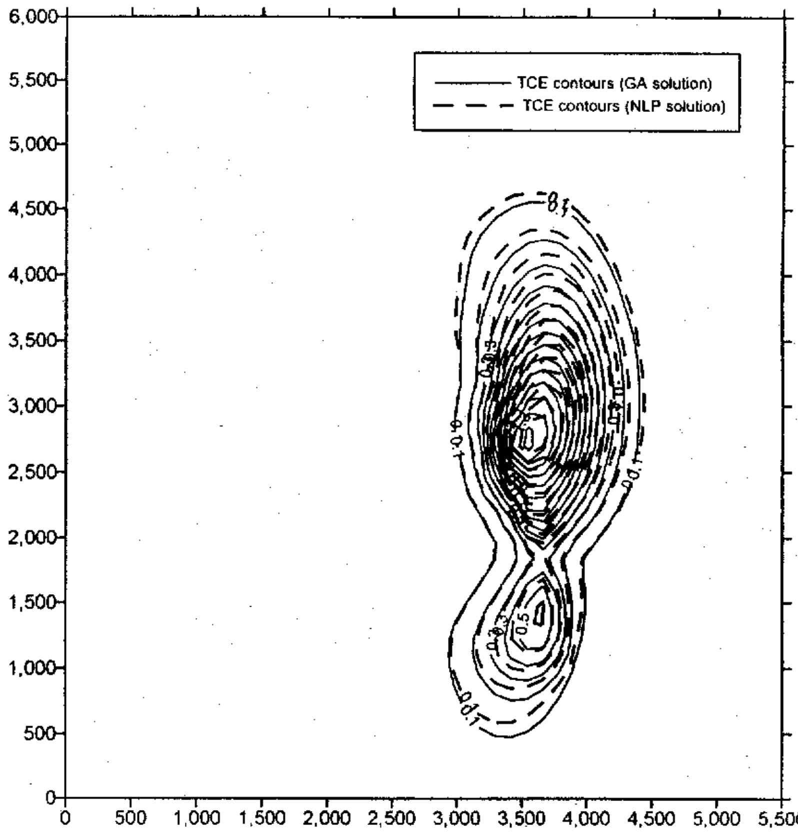

Figure 6. TCE contours after implementing optimal strategies for scenario C1.

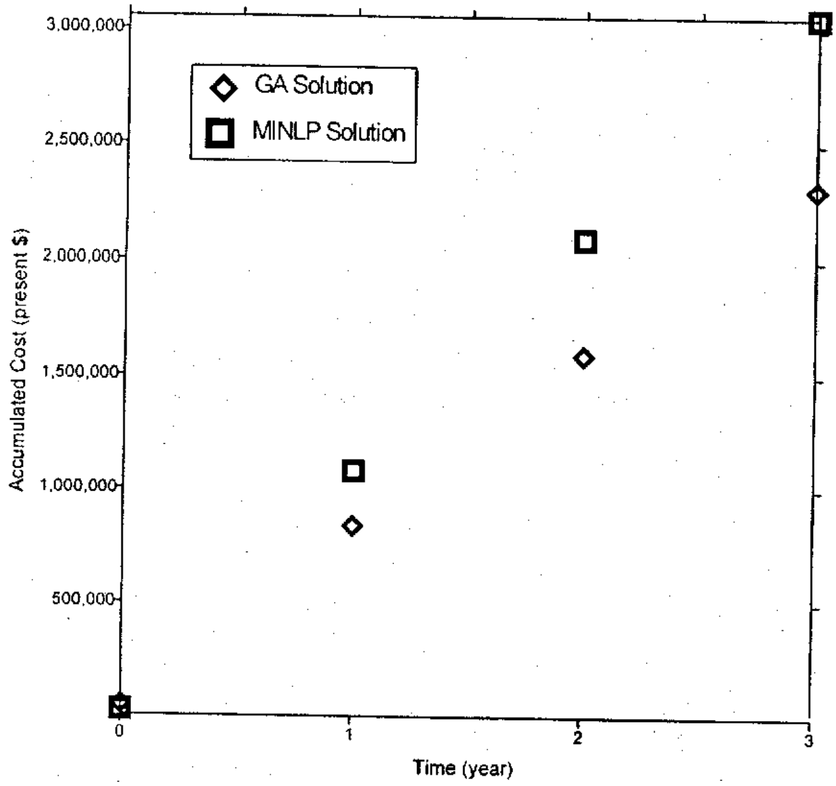

Figure 7. Accumulated cost versus time for scenario C2.

Since control parameter values have a great effect on the GA performance, careful control parameter selection is more important if the GA needs significant CPU time to solve the optimization problem. This situation arises in groundwater management when the embedding method is used to formulate the simulation constraints. Therefore control parameter selection is more important if the embedding method is used.

The functional form we used for the RS is merely one that performed well for all tested scenarios. Other functions might be better for other situations, especially when the number of wells increases.

For the cases evaluated in this study the GA performance was excellent. However, for more complex problems other operators can be investigated to enhance the GA performance. Niche methods, which keep solutions from different regions of the decision space, can be used to generate several optimal solutions and reduce the chances of premature convergence to local minima. Other operators, such as reordering operators, sexual determination, and elitism, introduce diversity into the population to introduce a similar effect. Other variations of tournament selection can be useful for different problems or when a large number of potential wells is used in the optimization problem formulation.

Although the methods presented in this paper are developed for aquifer cleanup problems, the methodology and formulation can be applied to other mixed integer nonlinear optimization problems.

Acknowledgment. This research was supported by the Utah Agricultural Experiment Station, Utah State University, Logan, Utah 84322-4810. Approved as journal paper 7154.

References

Ahlfeld, D. P., J. M. Mulvey, G. F. Pinder, and E. F. Wood, Contaminated groundwater remediation using simulation, optimization, and sensitivity theory, 1, Model development, *Water Resour. Res.*, *24*(3), 431–441, 1988.

Ahlfeld, D. P., Two-stage groundwater remediation design, *J. Water Resour. Plann. Manage.*, *116*(4), 517–529, 1990.

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

Atwood, D. F., and S. M. Gorelick, Hydraulic gradient control for groundwater contaminant removal, *J. Hydrol.*, *76*, 85–106, 1985.

Cieniawski, S. E., J. W. Eheart, and S. Ranjithan, Using genetic algorithms to solve a multiobjective groundwater monitoring problem, *Water Resour. Res.*, *31*(2), 399–409, 1995.

Cooper, G., R. C. Peralta, and J. J. Kaluarachchi, Optimizing separate phase light hydrocarbon recovery from contaminated unconfined aquifers, *Adv. Water Resour.*, *21*, 339–350, 1998.

Cunha, M. C. M. O., P. Hubert, and D. Tyteca, Optimal management of groundwater system for seasonally varying agricultural production, *Water Resour. Res.*, *29*(7), 2415–2426, 1993.

Davis, L. (Ed.), *Handbook of Genetic Algorithms*, Van Nostrand Reinhold, New York, 1991.

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

Draper, N. R., *Applied Regression Analysis*, John Wiley, New York, 1981.

Drud, A. S., A GRG code for large sparse dynamic nonlinear optimization problems, *Math. Programm.*, *31*, 153–191, 1985.

EA Engineering, Science and Technology, Groundwater modeling, technical report, *USAF Contract F41624-92-D-8005*, Lafayette, Calif., 1994.

Ejaz, M. S., and R. C. Peralta, Modeling for optimal management of agricultural and domestic wastewater loading to streams, *Water Resour. Res.*, *31*(4), 1087–1096, 1995.

Gharbi, A., and R. C. Peralta, Integrated embedding optimization applied to Salt Lake valley aquifers, *Water Resour. Res.*, *30*(3), 817–832, 1994.

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

Goldberg, D. E., A note on Boltzmann tournament selection for genetic algorithms and population-oriented simulated annealing, *Complex Syst.*, *4*, 445–460, 1990.

Gorelick, S. M., A review of distributed parameter groundwater management methods, *Water Resour. Res.*, *19*(2), 305–319, 1983.

Gorelick, S. M., C. I. Voss, P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright, Aquifer reclamation design: The use of contaminant transport simulation combined with nonlinear programming, *Water Resour. Res.*, *20*(4), 415–427, 1984.

Hoffman, F., Groundwater remediation using "smart pump and treat," *Groundwater*, *31*(1), 98–106, 1993.

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

IBM Corporation, *Optimization Software Library, Guide and Reference*, release 2, 3rd ed., Kingston, N. Y., 1991.

Joines, J. A., and C. R. Houck, On the use of non-stationary penalty functions to solve nonlinear constrained optimization problems with GAs, paper presented at First IEEE Conference on Evolutionary Computation, Inst. of Electr. and Electron. Eng. Neural Network Counc., Orlando, Fla., June 27–29, 1994.

Kocis, G. R., and I. E. Grossmann, Computational experience with DICOPT solving MINLP problems in process systems engineering, *Comput. Chem. Eng.*, *13*, 307–315, 1989.

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

Marquis, S. A., and D. Dineen, Comparison between pump and treat, biorestoration, and biorestoration/pump and treat combined: Lessons from computer modeling, *Ground Water Monit. Rev.*, spring, 105–119, 1994.

Marryott, R. A., Optimal groundwater remediation design using multiple control technologies, *Groundwater*, *31*(1), 98–106, 1996.

Marryott, R. A., D. E. Dougherty, and R. L. Stollar, Optimal groundwater management, 2, Application of simulated annealing to a field-scale contamination site, *Water Resour. Res.*, *29*(4), 847–860, 1993.

Matsukawa, J., B. A. Finney, and R. Willis, Conjunctive-use planning in Mad River basin, California, *J. Water Resour. Plann. Manage.*, *118*(2), 115–132, 1991.

McDonald, M. G., and A. W. Harbaugh, A three-dimensional finite-difference groundwater model, *U.S. Geol. Surv. Open File Rep. 83-875*, 1988.

McKinney, D. C., and M.-D. Lin, Genetic algorithm solution of groundwater management models, *Water Resour. Res.*, *30*(6), 3775–3789, 1994.

McKinney, D. C., and M.-D. Lin, Approximate mixed-integer nonlinear programming methods for optimal aquifer remediation design, *Water Resour. Res.*, *31*(2), 847–860, 1995.

Mercer, J. W., D. C. Skipp, and D. Griffin, Basics of pump-and-treat groundwater remediation technology, *Rep. EPA/600/8-90/003*, U.S. Environ. Prot. Agency, Washington, D. C., 1990.

Murtagh, B. A., and M. A. Saunders, *MINOS 5.1 User's Guide, Rep. SOL 83-20R*, Stanford Univ., Stanford, Calif., 1987.

Peralta, R. C., and R. Ward, Short-term plume containment: Multiobjective comparison, *Groundwater*, *29*(4), 526–535, 1991.

Peralta, R. C., H. Azarmnia, and S. Takahashi, Embedding and response matrix techniques for maximizing steady-state ground-water extraction: Computational comparison, *J. Ground Water*, *29*(3), 357–364, 1991.

Peralta, R. C., J. Solaimanian, and G. R. Musharrafiah, Optimal dispersed groundwater contaminant management: MODCON method, *J. Water Resour. Plann. Manage.*, *121*(6), 490–498, 1995.

Reichard, E. G., Groundwater-surface water management with stochastic surface water supplies: A simulation optimization approach, *Water Resour. Res.*, *31*(11), 2485–2865, 1995.

Ritzel, B. J., J. W. Eheart, and S. Rajithan, Using genetic algorithms to solve a multiple objective groundwater pollution containment problem, *Water Resour. Res.*, *30*(5), 1589–1603, 1994.

Rizzo, D. M., and D. E. Dougherty, Design optimization for multiple management period groundwater remediation, *Water Resour. Res.*, *32*(8), 2549–2561, 1996.

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.

Schaffer, J. D., R. A. Caruana, L. J. Eshelman, and R. Das, A study of control parameters affecting online performance of genetic algorithms for function optimization, in Schaffer, J. D. (ed.), *Proceedings of the Third International Conference on Genetic Algorithms*, edited by J. D. Schaffer, Morgan Kaufmann, San Francisco, Calif., 1989.

Staudte, R. G., and S. J. Sheather, *Robust Estimation and Testing*, John Wiley, New York, 1990.

Takahashi, S., and R. C. Peralta, Optimal perennial yield planning for complex nonlinear aquifers: Methods and examples, *Adv. Water Resour.*, *18*, 49–62, 1995.

Viswanathan, J., and I. E. Grossmann, A combined penalty function and outer approximation method for MINLP optimization, *Comput. Chem. Eng.*, *14*, 769–782, 1990.

Xiang, Y., J. F. Sykes, and N. R. Thomson, Alternative formulations for optimal groundwater remediation design, *J. Water Resour. Plann. Manage.*, *121*(2), 171–181, 1995.

Zheng, C., A modular three-dimensional transport model for simulation of advection, dispersion and chemical reactions of contaminants in groundwater systems, Robert S. Kerr Environ. Res. Lab., U.S. Environ. Prot. Agency, Ada, Oklahoma, 1990.

A. H. Aly, Department of Biological and Irrigation Engineering, Utah State University, Logan, UT 84322-4105. (alaa@ssol.agirrig.usu.edu)

R. C. Peralta, Department of Biological and Irrigation Engineering, Utah State University, Logan, UT 84322-4105. (peralta@cc.usu.edu)

(Received March 19, 1998; revised December 22, 1998; accepted December 23, 1998.)