
CHAPTER 8

GROUNDWATER REMEDIATION DESIGN USING SIMULATED ANNEALING

Richard L. Skaggs

*Pacific Northwest National Laboratory
Richland, Washington*

Larry W. Mays

*Department of Civil and Environmental Engineering
Arizona State University,
Tempe, Arizona*

8.1 INTRODUCTION

8.1.1 What Is Simulated Annealing?

There has been an emergence in the use of combinatorial methods such as simulated annealing in groundwater management during the past 10 years. Although previous studies demonstrated the feasibility of using these methods, a general finding was that computational processing requirements were inordinately high relative to gradient-based methods. An enhanced annealing algorithm was developed and used to demonstrate the potential for greatly improving the computational efficiency of simulated annealing as an optimization method for groundwater management applications. The algorithm incorporates “directional search” and “memory” capabilities. Selecting search directions based on better understanding of the current neighborhood of the configuration space was shown to improve algorithm performance. Also, memory concepts derived from the tabu search method show particular promise for improving the rate and quality of convergence. Performance of the enhanced annealing method was evaluated and the resultant management method was demonstrated using an example from the literature.

In recent years, groundwater quality management models (consisting of groundwater flow and contaminant transport simulation models coupled with optimization) have been developed to aid in groundwater remediation design. The simulator allows hydrogeologists to realistically estimate the effects of water withdrawal, injection, and contaminant movement under alternative remediation conditions. The optimization algorithms used in these groundwater management models have included linear programming and nonlinear optimization techniques, and more recently less conventional combinatorial optimization methods including neural networks (Rogers and Dowla, 1994), genetic algorithms (Ritzel et al., 1994), and simulated annealing (Dougherty and Marryott, 1991; Marryott et al., 1993).

Building on these and other research efforts, the overall objective of the research presented here is the development of a methodology focused on evaluation of alternative groundwater contamination cleanup technologies and management strategies. The methodology couples flow and contaminant transport simulation and optimization to enhance decision-making effectiveness when considering multiple issues such as technical, environmental, regulatory, and financial risks. The focus of the work is on development of an optimization method that identifies near-global optima without exorbitant computational resource requirements. The optimization method included in the methodology is enhanced simulated annealing, simulated annealing modified to include the concepts of memory and directional search.

8.2 CHAPTER EIGHT

Simulated annealing has been shown to locate global or near-global optima with a high degree of reliability. However, an important offsetting disadvantage is the relatively large number of required objective function evaluations. This is a particularly acute problem in groundwater management when hundreds to thousands of objective function evaluations may be necessary to solve a given problem, and each objective function evaluation is based on a flow and transport simulation requiring up to several minutes of computer processing time. Thus, an enhanced annealing methodology was developed to significantly reduce the number of objective functions required in the optimization procedure.

8.1.2 Background

Since the late 1970s, there has been a considerable body of work directed at the development of management models to assist decision-makers in selecting the most reliable and cost-effective methods for remediation of contaminated groundwater. The general problem, as viewed by Wanakule et al. (1986), is one of discrete time optimal control where variables describing the aquifer system (e.g., heads, concentrations, etc.) are direct functions of the control variables (e.g., pumping rates). Management models generally consist of an optimization method linked to groundwater flow and solute transport models as subroutines to evaluate the cost and feasibility of alternative remediation designs. The majority of the optimizers used are linear, nonlinear, or differential dynamic programming optimal control methods. Recently, efforts have investigated the performance of other types of optimizers from the field of combinatorial optimization.

Dougherty and Marryott (1991) provide an overview of the simulated annealing algorithm and guidance on its application to groundwater management. They note the simplicity with which simulated annealing is implemented and coupled to flow and transport simulators. They apply their algorithm to several problems ranging from a highly idealized dewatering problem to one combining injection/extraction wells with a slurry trench. The dewatering problem is used to demonstrate relative performance of several variations of the algorithm and the iterative concept of *zooming* to reduce the number of objective function evaluations. *Zooming*, a refinement method in which a sequence of refined estimates of the objective function is solved, is found to significantly reduce the computation time but reduce the likelihood of locating a global minimum.

The pump-and-treat remedial design process is modeled by linking a simulated annealing algorithm with a flow-contaminant transport simulator. This application was used to demonstrate the advantages of problem-specific heuristics in reducing the objective function calls (e.g., screening alternatives based on different components of the objective function, and retaining objective function values for a limited number of previously attempted configurations). The authors also suggested the flow and transport problem could be solved only approximately in the early stages of the solution without significantly affecting results. With a fourth example they demonstrated the ease with which multiple technologies can be addressed by evaluating a pump-and-treat design coupled with a slurry trench. A comparison of simulated annealing with the MINOS (Murtagh and Saunders, 1980) algorithm suggested that CPU requirements for annealing increase at a slower rate than for the gradient-based method as the level of contaminant removal is increased.

In a sequel to the paper discussed above, Marryott et al. (1993) apply simulated annealing to an actual contaminated field site. Simulated annealing coupled with a flow and transport simulator is used to solve two problems. The first is a regulatory constraint formulation in which concentration levels in the aquifer are reduced below regulatory standards by the end of a designated planning period. The second is a velocity constraint formulation in which the goal is plume containment. Results from the evaluations show that the first formulation produces lower minimum costs and the second requires less computational effort. They indicate that although reducing the number of objective function evaluations at each temperature of the cooling schedule can greatly reduce the computational effort, it also decreases the probability of achieving the optimal solution.

The authors conclude that the computational effort required for simulated annealing, although large, is comparable to other nonlinear optimization techniques. And, although the simulated annealing method applied to groundwater management shows promise in reliably locating near-global

optima for groundwater management problems, it is not yet mature and extensions and enhancements are needed.

8.1.3 Statement of the Problem

The solution process for simulated annealing is typically composed of several steps. Beginning at an elevated control parameter value, a completely stochastic search is conducted. Initially, sequential moves within the configuration space are large in magnitude as is the relative probability of making an uphill move (for a minimization problem). From initiation of the search to convergence, both the magnitude and direction of the move are blindly controlled by probabilistic functions. As the control parameter is decremented, the magnitude of the moves and the likelihood of moving uphill decrease exponentially at the same relative rate until convergence is achieved. In developing and selecting future moves, the simulated annealing algorithm contains no mechanisms that explicitly consider information about past moves (regions of relative success or failure). Nor can it consider the relative merit of one direction over another when evaluating candidate moves from a current location (e.g., gradients or projections). In other words, all moves are dictated by the current status of the annealing process and comparison of the current objective function value with that of proposed moves to randomly selected candidates.

The potential value in considering directional preferences and past moves in making search decisions can be deduced from a typical simulated annealing search pattern. Early, when the search range is large and the probability of moving uphill is high, large swings in objective function values are common. Almost all proposed moves, whether uphill or downhill, are taken. Later in the search, after only a few hundred iterations, the search has converged and narrowed to a smaller region providing only minor variations in objective function value. Thus, a significant percentage of the latter iterations are expended in a relatively small region of the configuration space. This is attributable to inherent features of the annealing process:

1. During the latter part of the search the potential change (i.e., search range) for all decision variables is uniformly small.
2. The likelihood of changing all variables is the same, regardless of which variables might have the greatest potential impact on the objective function value.

If the magnitude of movement could be increased for variables to which the objective function value is least sensitive (i.e., move into another region of the configuration space), the likelihood of locating a potential improvement and escaping a local minimum could be increased. The probability of wasting iterations on candidate moves representing little or no potential improvement could be reduced, thus increasing the efficiency of the search.

Also, near the end of the search a large number of iterations are expended in a small region of the configuration space and the likelihood of reencountering a given configuration becomes relatively high. When this occurs, no new information can be gained from reevaluating the objective function at these locations. In addition to individual sites, entire regions of the encountered configuration space may demonstrate little or no promise of containing a global or near-global optimum. It follows that the ability to remember recently encountered configurations or even sequences of iterations that proved unfruitful could, therefore, further reduce wasted (unfruitful) objective function evaluations.

8.2 ENHANCED ANNEALING

Simulated annealing is a stochastic computational method derived from statistical mechanics. As observed by Kirkpatrick et al. (1983), "there is a deep and useful connection between statistical mechanics (the behavior of systems with many degrees of freedom in thermal equilibrium at a finite temperature) and multivariate or combinatorial optimization (finding the minimum of a given function

8.4 CHAPTER EIGHT

depending on many parameters).” Kirkpatrick et al. (1983) first demonstrated its applicability to large combinatorial optimization problems such as wire routing and component placement in computer design. Subsequent to its development, simulated annealing has proven effective in solving a multitude of hard optimization problems (i.e., locate near-global optima) containing multiple local optima, while being straightforward and efficient to implement (Davis and Steenstrup, 1987). In contrast to gradient-based methods, simulated annealing does not require calculation of derivatives, thereby eliminating the need for the objective function and constraints to be continuous and differentiable. Incorporating the concepts of directional search and memory into the simulated annealing algorithm results in an algorithm referred to as *enhanced annealing* (Skaggs, 1995; Skaggs et al., 2001a; Skaggs et al., 2001b).

Simulated annealing gets its name from and is based on the analogy between the simulation of the annealing of solids and solving large combinatorial optimization problems. A solid is annealed by increasing its temperature so that its molecules are randomly arranged in the liquid phase, followed by slow cooling to allow the molecules to arrange themselves in the low-energy ground state of a corresponding lattice. If the thermal ability is lost at a sufficiently slow rate, the molecules are able to align themselves into a completely ordered crystalline structure. This configuration is the state of minimum energy for the system. If the cooling occurs too rapidly, the system does not achieve the highly ordered state, but ends up in a meta-stable, higher-energy state.

Because the number of atoms in a given system is extremely large, statistical mechanics is used to stochastically analyze the aggregate properties of the material. Simulating the attainment of thermal equilibrium of a solid for a fixed temperature can be accomplished using a Monte Carlo method proposed by Metropolis et al. (1953). At each temperature value T , a system allowed to achieve thermal equilibrium has the probability of being in energy state E given by the *Boltzmann probability distribution*,

$$P\{\mathbf{E} = E\} = \frac{1}{Z(T)} \exp\left(-\frac{E}{k_B T}\right) \quad (8.1)$$

where $Z(T)$ = partition function depending on temperature T
 k_B = Boltzmann constant
 $\exp(-E/k_B T)$ = Boltzmann factor

Equation (8.1) specifies that a system in thermal equilibrium at temperature T has its energy probabilistically distributed among all different energy states, $E_j \in E$. Using this equation, it is possible to predict the thermodynamic behavior of an idealized natural system as it is annealed. At high temperatures, the system is constantly changing and the probability is relatively high for all energy states. As the system is cooled, the probability of changing to higher-energy states decreases. However, even at low temperatures, there is still a finite chance that the system will transform to a higher-energy state, and thus escape from a local energy minimum in favor of finding a better, more global one.

Computationally, given a current state of the solid characterized by the positions of the particles, a randomly generated small displacement is applied to a randomly selected particle. If the difference in energy ΔE , between the current state and the perturbed state is negative, the perturbed state is accepted as the new state. If $\Delta E \geq 0$, then the probability of accepting the new state is determined by Eq. (8.1). This acceptance rule (function) for new states is known as the Metropolis criterion. In applying this to combinatorial optimization, the configurations of the optimization problem are analogous to the states of the solid whereas the cost function takes the place of energy. Temperature simply becomes a control parameter for the algorithm.

Several steps are required in the implementation of simulated annealing. Initial values of the control parameter, c_0 , and the cost function, C_0 , are determined. A specified number L of Metropolis loops are then completed. Each loop consists of generating a new configuration, calculating its associated cost function value C_1 , and applying the Metropolis criterion to determine whether the new configuration is accepted (i.e., C_0 is replaced by C_1) or rejected. After L loops have been completed the control parameter is adjusted and a new Metropolis loop is begun. Algorithms used for each of the key steps and the associated parameters are the following.

Initial value of the control parameter, c_0 . For an idealized cooling schedule c_0 should be selected such that almost all transitions are selected. The approach by Vidal (1993) computes c_0 according to Eq. (8.2) where m_0 is a preset number of transitions for which m_1 produce $DC \leq 0$, m_2 produce $DC > 0$, $\Delta C^{(+)}$ is the average increase in C over the m_2 transitions, and χ is equal to a value close to 0.90.

$$c_0 = \overline{\Delta C^{(+)}} \left[\ln \left(\frac{m_2}{m_2 \chi - m_1 (1 - \chi)} \right) \right]^{-1} \quad (8.2)$$

Decrements in the control parameter, c_0 . A commonly used rule to ensure quasi-equilibrium is achieved from one Metropolis loop to the next is simply

$$c_{k+1} = \alpha_{k+1} c_k \quad k = 0, 1, 2, \dots, \quad (8.3)$$

where α ranges from 0.5 to 0.99 but generally is close to 1.0 (Laarhoven and Aarts, 1987). Chardaire and Lutton (1993) propose a slightly different approach that begins by presetting the number of control parameter decrements T such that Eq. (8.3) holds and

$$\alpha^T = 0.1 \quad (8.4)$$

which, for T equal to 25 would make α equal to about 0.91. However, it is suggested that T should be increased with decreasing values of the control parameter according to

$$T_{k+1} = T_k \omega \quad \text{with } 1.0 < \omega < 1.05 \quad (8.5)$$

Markov chain length, L . Underlying the approach to selecting L is the notion that to achieve global minimization, quasi-equilibrium must be achieved at the completion of each Metropolis loop. Typically, the value of L is empirically determined, but Eq. (8.6), as recommended by Anderson and Vidal (1993), can serve as a guide where n is the number of decision variables.

$$L = 2(n - 1)n \quad (8.6)$$

Neighbor generation. The strategy used to make the transition from the current configuration to a neighboring configuration tends to be problem specific and dependent on how the solution space is configured. The major requirement is that transition from a given configuration to any other configuration should be possible in a finite number of moves. However, as suggested by Laarhoven and Aarts (1987), the neighbor generation mechanism should be biased toward “large” transitions at high values of the control parameter (i.e., produce large variations in the objective function value). And at low values of c , when large increases are often rejected, the bias should be toward small transitions. As a guide they suggest transitions should produce changes in the objective function approximately equal to the control parameter.

Dougherty and Marryott (1991) recommend an excursion limiting approach which simply varies the value of each of the decision variables by $\pm l$ levels from its current value. At the initial value of c , the magnitude of the excursion is defined by

$$\pm l = \pm s_l = \pm \frac{M}{2} \quad (8.7)$$

where s_l varies from $M/2$ to a lower limit equal defined as

$$s_l = \max \left[s_l \exp \left(\frac{C_0 - C_k}{C_k} \right), s_{\min} \right] \quad (8.8)$$

and M is the number of alternatives available for each decision variable. Thus, a transition is accomplished by adjusting each decision variable by an amount equal to the integer nearest

8.6 CHAPTER EIGHT

$$l = s_l \text{random}[0,1] \quad (8.9)$$

where C_0 and C_k are the initial and current values of the objective function, respectively. s_{\min} is equal to 0.51.

Convergence criteria. The annealing process is terminated when the rate of change in the average objective function values falls below a specified level as determined by

$$\overline{C}_k - k^{-1} \sum_{k=3}^k C_k \leq \epsilon \quad (8.10)$$

where ϵ is a small positive number (say 0.05) and k equals 8.

8.3 ANNEALING WITH DIRECTIONAL SEARCH

An important aspect of traditional nonlinear optimization methods is the use of gradient operators at a point to determine the search direction. For a continuous and continuously differentiable function $f(\mathbf{x})$ at point x , the gradient is defined as:

$$\nabla f(\mathbf{x}) = \left[\frac{\partial f}{\partial \mathbf{x}} \right] = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \dots, \frac{\partial f}{\partial x_n} \right)^T \quad (8.11)$$

and $\nabla f(\mathbf{x})$ represents the direction of steepest descent. In the combinatorial optimization problem addressed by simulated annealing, the conditions of continuousness and differentiability are not necessarily met; thus, selection of directions based on function gradients is not feasible.

However, in lieu of gradients, simulated annealing can be manipulated through its dependence on probabilities. Standard simulated annealing, when selecting a proposed transition from a given point, modifies all decision variables through the use of a probability function related to the current value of the control parameter c and a uniform probability distribution. The excursion length s_{\min} is equal for all the decision variables as is the exponential rate at which the excursion length decreases from its original value of $M/2$. Thus, the probability of a given magnitude dislocation at a particular stage of the annealing process is equal for all decision variables. An alternative approach would be, at strategic instances within the search, to adjust the excursion lengths of selected variables to increase the likelihood of moving to a potentially more desirable configuration.

It was observed by Ingber (1993) that in conducting annealing searches for many D -dimensional parameter space problems, due to physical considerations, different parameters might have different finite ranges and different annealing time-dependent sensitivities. This observation led to the consideration of independent annealing schedules for individual parameters in his adaptive simulated annealing method. Consequently, Ingber (1993) implemented a *reannealing* procedure whereby the range over which insensitive parameters are being searched is stretched out. In his algorithm the magnitude of search for a given parameter is a function of the annealing temperature wherein the annealing temperature is related to annealing time by

$$T_i(k_i) = T_{0i} \exp(-c_i k_i^{1/D}) \quad (8.12)$$

where k_i = annealing time step for parameter i

k = zero initially, incremented by 1 at completion of each temperature step

T_{0i} = initial annealing temperature for parameter i

c_i = tuning parameter

D = problem dimensionality

The reannealing is accomplished by periodically rescaling the annealing time for parameter α_i in terms of the sensitivities s_i calculated at the current value of the cost function L ,

$$s_i = \frac{\partial L}{\partial \alpha^i} \quad (8.13)$$

Thus, the annealing time is rescaled according to

$$T'_{ik} = T_{ik} \left(\frac{s_{\max}}{s_i} \right) \quad (8.14)$$

and

$$k'_i = \left(\frac{\ln(T_{ik}/T'_{ik})}{c_i} \right) \quad (8.15)$$

The effect of the reannealing is, for those parameters that are insensitive at the current configuration, to increase their annealing temperature and thereby increase the range of their search. Similarly, the components of the simulated annealing algorithm can be modified to incorporate the reannealing procedure.

The first step in incorporating reannealing into the simulated annealing algorithm is to include, along with the control parameter c , additional control parameters c_i where i varies from 1 to the number of decision variables. Thus, in the annealing process, the acceptance function is annealed according to control parameter c . And the decision variables are annealed independently according to respective values of c_i . At appropriate times during the annealing process, decision variable sensitivities are determined by

$$s_i = \frac{\partial C}{\partial Q_i} \cong \frac{C_{\text{scale}} - C_0}{Q_{\text{scale}} - Q_0} \quad i = 1, \text{NW} \quad (8.16)$$

where s_i = sensitivity of objective function to decision variable i at current configuration
 C_0 = current value of objective function computed for vector of decision variable values Q_0
 C_{scale} = value of objective function computed for vector of decision variables Q_0 with Q_i incremented by DQ
 NW = number of decision variables

Given the sensitivities from Eq. (8.16), the decision variable control parameters are then rescaled according to

$$c_i = \frac{s_{\max}}{s_i} \quad i = 1, \text{NW} \quad (8.17)$$

and

$$c_i \leq c_0 \quad i = 1, \text{NW} \quad (8.18)$$

where c_i = control parameter value for decision variable i
 s_i = sensitivity for decision variable i
 s_{\max} = maximum sensitivity for all decision variables
 c_0 = initial control parameter value
 NW = number of decision variables

The annealing schedule for the simulated annealing algorithm has a parameter T that is analogous to the annealing time parameter k used by Ingber (1993). T varies throughout the annealing process

8.8 CHAPTER EIGHT

and, along with w , controls the annealing rate. Thus, values of T_i must be determined for each of the rescaled c_i 's according to the following equations:

$$k_i = \text{NINT} \left[\frac{\ln(c_i/c_0)}{\ln(0.1)^{(10/T_0 \cdot \omega)} } \right] - 1 \quad i = 1, \text{NW} \quad (8.19)$$

and

$$T_i = T_0 \omega^{k_i} \quad i = 1, \text{NW} \quad (8.20)$$

where k_i = nearest integer to rescale value of T_i relative to initial input value T_0 ; w = input annealing schedule parameter; and c_p , c_0 , and NW are as previously defined.

Given the rescaled values of c_i and T_p the decision variable control parameters are annealed as before. In addition, the excursion lengths are also rescaled by

$$s_{ii} = s_{i0} \exp \left(\frac{c_i - c_0}{c_0} \right), \quad i = 1, \text{NW} \quad (8.21)$$

where s_{ii} = rescaled excursion length for decision variable i ; s_{i0} = initial excursion length for each decision variable; and c_p , c_0 , and NW are as previously defined.

Finally, at the completion of each Markov chain loop, s_{ii} is adjusted based on the current value of c_i by replacing values of C_0 and C_k in Eq. (8.22) with c_0 and c_p respectively, as follows:

$$s_{ii} = \max \left[s_{ii} \exp \left(\frac{c_0 - c_i}{c_i} \right), s_{\min} \right] \quad i = 1, \text{NW} \quad (8.22)$$

where s_{ip} , c_0 , c_i , s_{\min} , and NW are as previously defined.

8.4 ANNEALING WITH MEMORY

It has been observed here and by other authors [e.g., Greene and Supowit (1986) and Ingber (1993)] that a significant portion of the total computational effort for standard simulated annealing is expended at low temperatures during the latter stages of the search. The bulk of this effort is spent evaluating proposed moves that are subsequently rejected. As the search proceeds the range of the search is reduced along with the likelihood of making an uphill move, and a significant chance arises of a given configuration being proposed multiple times. There is also no retention of information from previous moves. In other words, the move from a particular location is based solely on the relative values of the objective functions and the control parameter for the current and proposed move as dictated by the Boltzmann function. Finally, near the end of the search there can be a tendency to cycle over a limited number of moves, leaving and returning to the local minimum. It's noted that the use of rescaling, as was demonstrated in the previous section, increases the search range for at least a subset of the decision variables. A benefit of this is an increase in the number of accessible configurations (e.g., diversification of candidate configurations) from a given location and thereby reducing the chance of repeats. However, the literature related to the tabu search method (TS) suggests that the efficiency of local search procedures can be improved by memory processes.

The tabu search method was developed by Glover (1989, 1990, 1994b, 1995) and has been used to obtain improved solutions to problems in scheduling, sequencing, resource allocation, investment planning, telecommunications, and many other areas. However, there are no known applications of TS to groundwater management problems. Derived originally by introducing memory into the scatter search method (Glover, 1994a), TS is a combinatorial optimization approach for exploring solution space beyond local optimality. In contrast to memoryless methods (e.g., simulated annealing and genetic algorithms), TS is founded on the premise that intelligent problem solving must incorporate

adaptive memory and responsive exploration. It is presumed that a bad strategic choice can contribute more to the search than a good random choice. The following highlights some of the key TS features. For a comprehensive discussion of the many alternative TS processes, the reader is referred to Glover (1995). An extensive anthology of TS applications and empirical experience is provided by Hammer (1993).

Implementation of TS begins similarly to other combinatorial search methods wherein some function $C(\mathbf{x})$ is optimized over a solution space $S = \{\mathbf{x} | \mathbf{x} = (x_1, x_2, \dots, x_m)\}$ where \mathbf{x} is the state vector. TS proceeds iteratively, moving from one solution to the next until the termination criteria are achieved. Each \mathbf{x} has a neighborhood $N(\mathbf{x}) \in S$ and each solution $\mathbf{x}' \in N(\mathbf{x})$. However, TS extends the search by modifying $N(\mathbf{x})$ as the search proceeds, replacing it by another neighborhood $N^*(\mathbf{x})$. A key element of TS is the use of memory to determine $N^*(\mathbf{x})$ and thereby organize the manner in which S is explored.

The solutions admitted to $N^*(\mathbf{x})$ are determined by any of several TS mechanisms. One, from which TS derives its name, labels solutions encountered as tabu and forbids their inclusion in $N^*(\mathbf{x})$ for a specified duration. Application of tabu status is intended to promote strategically aggressive search trajectories. In effect, original objective function evaluations are complemented by tabu evaluations to either encourage or discourage the selection of certain type solutions for inclusion in $N^*(\mathbf{x})$.

TS uses both explicit and attributive memory. Explicit memory retains complete solutions, typically elite or high-quality solutions encountered during the search or their neighbors. Attributive memory, on the other hand, records changes in solution attributes associated with moves. Both types of memory are applied as functions of event recency, event frequency, and differentiation between short and long term. They are applied in ways to concentrate focus on good regions or good solution features and exploring promising new regions.

The two basic types of TS memory are generally characterized as short term and long term. The most commonly used short-term memory is called recency-based memory and simply tracks solution attributes that have changed in the immediate past. Tabu status is applied to selected attributes that occur in recent solutions. Solutions that contain tabu attributes are thereby excluded from $N^*(\mathbf{x})$ and cannot be revisited while they retain their tabu status. The duration (i.e., number of iterations) that an attribute remains tabu-active is referred to as tabu tenure. Tabu tenure can be varied for different attributes or combinations of attributes as well as for different stages of the search. A flow-chart describing the implementation of short-term memory is presented in Fig. 8.1. The procedure begins with an initial solution \mathbf{x} . The neighborhood $N^*(\mathbf{x})$ is generated, excluding solutions containing tabu attributes. From this list, the best solution \mathbf{x}' is selected and a move is made. The tabu list is then updated and a check for termination is made. If the termination criteria are not met (e.g., exceed k_{\max} , no solutions remaining in the neighborhood, etc.), a new neighborhood is generated around the new \mathbf{x} , and the process continues.

Variations of the above include using aspiration criteria that would overrule tabu status for high-quality solutions (e.g., best yet encountered) or graduated penalties that would allow for varying degrees of tabu status. Glover (1995) notes that the short-term TS memory components are sufficient to obtain high-quality solutions for many problems. However, he also states that TS can be made significantly stronger by introducing longer-term memory components. Long-term memory considerations are primarily based on the frequency of events. Transitional frequencies track how often selected attributes change, whereas residence frequencies track relative durations with which attributes occur in generated solutions. Two important long-term memory considerations are intensification strategies and diversification strategies. Intensification strategies are designed to influence choices toward move combinations and solution attributes that historically produce good solutions. An example would be, subsequent to convergence at a local minimum, use of restarting procedures that incorporate good attributes from previous solutions.

Diversification strategies, in contrast, drive the search into new regions. They are often based on restricting attribute choices to those infrequently used. A basic example is that referred to as strategic oscillation whereby the search is driven toward and away from selected boundaries of feasibility by manipulating the objective function (e.g., adjusting penalties and incentives). A strategy that links diversification and intensification is path re-linking. This approach retains elite solutions in

8.10 CHAPTER EIGHT

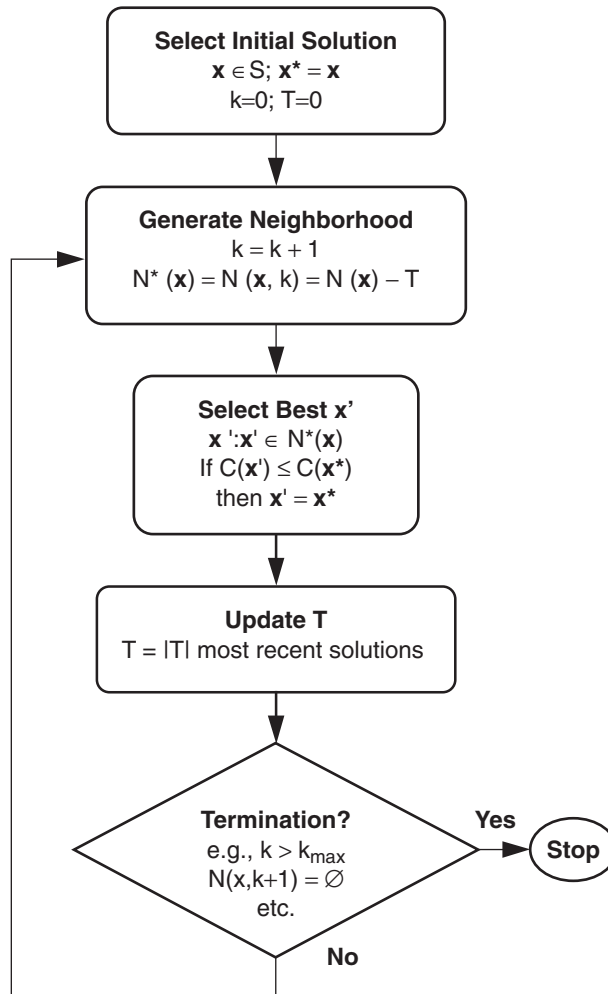


FIGURE 8.1 Flowchart of tabu search short-term memory process.

long-term memory. It is implemented by exploring trajectories connecting one or more of the elite solutions. Starting at an initiating solution, moves are selecting by introducing attributes from guiding solutions (also from the elite set) with the goal of creating a good attribute composition in the current solution.

From the description of TS, some conceptual parallels and differences between TS and simulated annealing are apparent. Although TS makes decisions relative to a neighborhood containing multiple alternative solutions, simulated annealing always makes decisions pertaining to a single proposed move. Both methods employ mechanisms for diversification and intensification. TS utilizes a variety of memory-based processes to balance diversification and intensification throughout the search, whereas simulated annealing tends to focus on diversification early (high uphill move probability and wide search range) in the annealing process, and more on intensification (low uphill move probability and limited search range) during the latter stages of the search. It is noted

that the directional search capability discussed earlier does provide for continued diversification throughout the search.

The collection of similarities and differences suggests that simulated annealing could potentially be improved by incorporating elements of memory. Although it was beyond the scope of the effort presented here to exhaustively explore the numerous possibilities, some simpler TS memory processes were adapted for use by the enhanced annealing algorithm presented.

8.5 MEMORY COMPONENTS

Three memory components were selected for inclusion in the enhanced annealing algorithm:

1. Short-term recency memory of last NT1 moves
2. Short-term recency memory of an attribute for last NT2 iterations
3. Long-term record of all configurations visited after 1000 iterations

The first two components are inspired by similar TS processes. Retention of the last NT1 moves for a duration of NT1 iterations (i.e., a circular list wherein as one configuration is added to the list another is dropped) allows for the avoidance of cycling near the end of the annealing process. Although the first tabu list is in effect throughout the search, it only becomes functional near the end of the search when the search range is small and the likelihood of returning to a recently visited site is high. The expected benefits from this mechanism are twofold. First, the prevention of cycling should enhance the rate and/or quality of convergence by in turn preventing the expansion of iterations within loops that have little or no probability of containing a more optimal solution. And, computational resources are not wasted on reevaluating objective functions.

The second tabu list is intended as an intensification strategy. Subsequent to each move, the direction of change in total pumpage is recorded. For NT2 iterations subsequent to the move, configurations that would reverse the direction of change in total pumpage are tabu. In other words, if the last move was the result of an increase in pumpage, then for the next NT2 iterations, only those configurations reflecting an increase in pumpage would be considered. Again, dual benefits are expected from this strategy. The avoidance of reversals in changes in total pumpage will provide persistence in the search direction. If the search is heading toward a local optimum, it can't be derailed by a stochastically generated uphill move until the search has an opportunity to explore the current direction more deeply. In addition to accelerating the convergence process, for those proposed moves violating the tabu criteria, objective functions are not evaluated.

The third memory component, although not a TS-derived process, relies on long-term memory. It consists simply of retaining a condensed record of all configurations visited and their objective function values. When one of the configurations is reencountered, instead of reevaluating the objective function, it is simply located in memory. The tradeoff for this memory component is the savings in computational resources for the sites visited more than once (i.e., not calling the simulator), against the time required to search the record for every proposed configuration. The records are scanned using a binary search approach. The configurations are stored in a condensed form using a simple form of hashing as described by Woodruff and Zemel (1993). The hashing function used is

$$h_0 = \sum_{i=1}^n z_i x_i \quad (8.23)$$

where h_0 = hashed configuration

z_i = precomputed vector of pseudo-random integers in the range 1 to 4×10^9

x_i = configuration vector defining decision variables

8.12 CHAPTER EIGHT

Results of the hashing are stored as 4-byte characters. The probability of a collision (i.e., two different vectors being encountered with the same hash function value) is extremely remote. For example, for a construction site dewatering problem, given a scenario with 10 wells and 16 alternative flow rates for each well, there are 16^{10} configuration permutations. Therefore, each of the 4 billion hash function values represents approximately 275 different configurations. If all different encountered configurations were stored over 100 Monte Carlo solutions up to 100,000 configurations, the joint probability of encountering two of the 275 matches for a given hash function value is on the order of 6×10^{-10} .

8.6 OVERALL ENHANCED ANNEALING ALGORITHM

A flowchart depicting the enhanced annealing algorithm, having both directional search and memory capabilities, is shown in Fig. 8.2. As discussed above, modifications to the simulated annealing algorithm are as follows:

1. After each new configuration is generated, a check is made to determine if the configuration is tabu (i.e., recently visited or tabu total pumpage). If the configuration is tabu, another configuration is generated without computing the objective function. If the configuration is not tabu, a check is made to determine if it has been previously visited. If it has, the objective function value is retrieved from the record. Otherwise, the objective function is evaluated. The annealing process then proceeds as usual.

2. If the number of iterations is greater than NRSL rescaling is implemented as described in Fig. 8.3. Otherwise, the control parameters for the acceptance function, individual decision variables, and the excursion length are adjusted. The excursion lengths are adjusted individually. In addition, they are adjusted as a function of the initial control parameter value and the current decision variable control parameter value.

8.7 APPLICATION OF ALGORITHM: N-SPRINGS SITE DESCRIPTION

8.7.1 Problem Description

Groundwater contamination from past and present industrial, agricultural, and commercial activities is one of the leading environmental challenges of our time. It is estimated that from 300,000 to 400,000 sites in the United States may contain contaminated groundwater. Past estimates suggest that the cost of cleanup for these sites over the next 30 years could amount to as much as \$1 trillion (National Research Council, NRC, 1994). Resultant risks to public health and ecosystems, and the associated high costs of remediation make it essential that aquifer management strategies be developed in the most reliable and cost-effective manner achievable.

The enhanced annealing algorithm has been developed and used in the new methodology for groundwater remediation design. The flowchart of this new groundwater management methodology is illustrated in Fig. 8.4. Essentially the remediation design is a discrete-time optimal control problem that is solved by interfacing a flow and transport simulator with the simulated annealing optimizer.

N-Springs is located at Hanford, Washington and is the location of the 1301-N Liquid Waste Disposal Facility (LWDF) which was used for disposal of radiologically contaminated water associated with pass-through cooling, spent fuel storage, and other activities related to operation of the N Reactor (U.S. DOE, 1995). The study area, located adjacent to the Columbia River, is shown in Fig. 8.5. The LWDF is a long zigzagging trench running parallel to and about 800 ft from the Columbia River. From 1963 until 1985, the LWDF received discharges containing both radiological and nonradiological

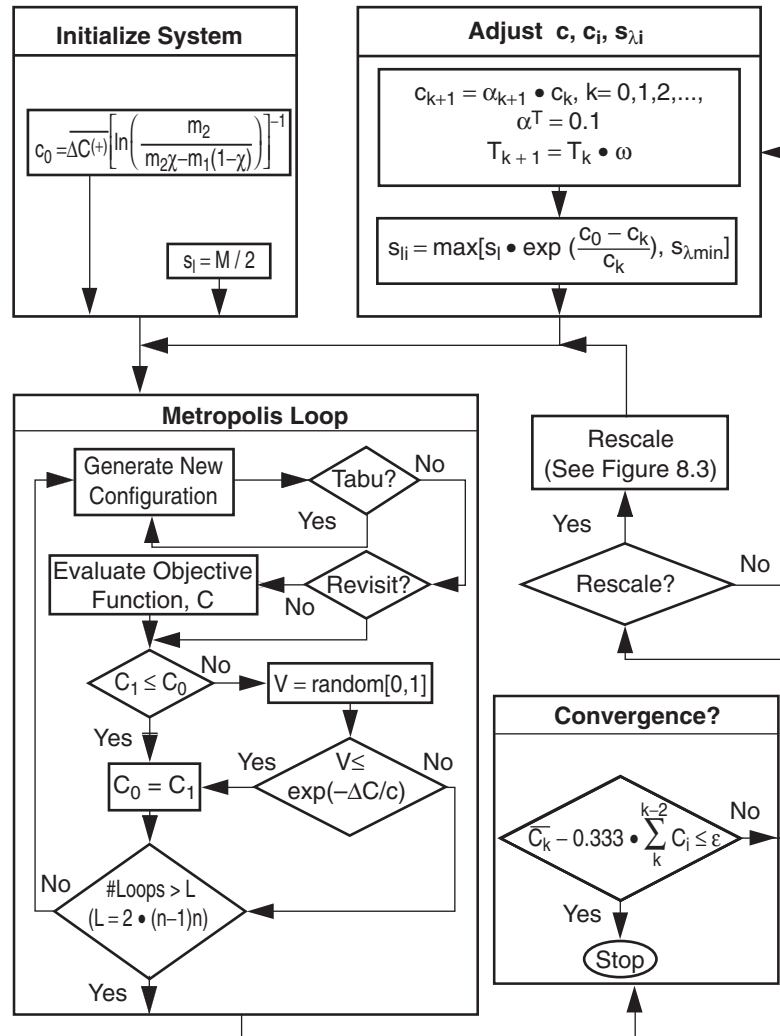


FIGURE 8.2 Enhanced annealing with memory and directional search.

contaminants at a rate of approximately 2100 gal/min. Almost immediately after receiving water, mobile contaminants from the LWDF were observed in springs along the banks of the Columbia River. Beginning in 1980, ^{90}Sr was detected in the groundwater near the river. It is estimated that 2997 Ci of ^{90}Sr were discharged to the facilities at N-Springs of which ~ 46 Ci were released to the Columbia River and 1085 Ci have decayed away. The majority of the radiological inventory remaining at the N-Springs area is located in the sediments underlying the LWDF including 75.5 Ci within the unconfined aquifer. Of this amount, 75 Ci are estimated to be adsorbed to the sediments and 0.5 Ci are dissolved in the groundwater.

Contours depicting the areal concentration distribution of ^{90}Sr in the groundwater at N-Springs are shown in Fig. 8.6. The plume covers approximately 55 acres having maximum concentrations on the order of 4000 pCi/l. The highest concentrations are shown to be immediately beneath and down-gradient (i.e., toward the river) from the LWDF.

8.14 CHAPTER EIGHT

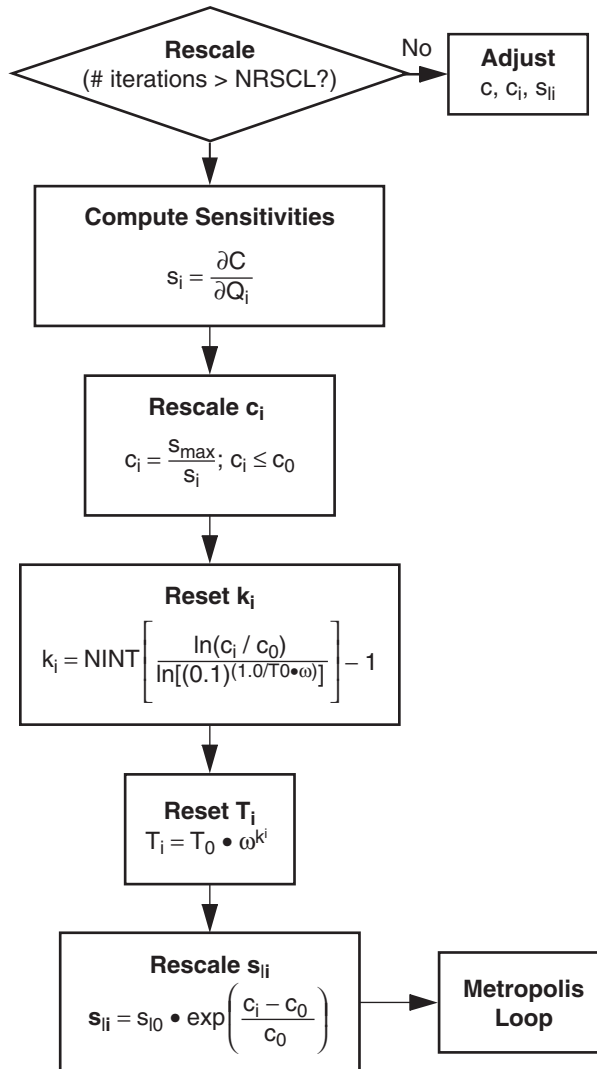


FIGURE 8.3 Flowchart of rescaling process (see Fig. 8.2).

8.7.2 Demonstration Scope

Alternative physical barrier and pump-and-treat designs were evaluated as part of an expedited response action to substantially reduce the seepage of ^{90}Sr into the Columbia River. The goal for the systems evaluated was to extend the travel path for the ^{90}Sr , taking advantage of its slow migration rate. The K_d for ^{90}Sr is 15.5 mL/g, resulting in a retardation factor of 100.2. Thus, increasing the path length to the river can dramatically enhance natural radioactive decay. Three barrier designs having lengths of 2000, 3000, and 4000 ft were considered. All fully penetrate the unconfined aquifer and are positioned approximately 30 ft from and parallel to the Columbia River. In conjunction with the barriers, the U.S. DOE study analyzed the performance of three different pump-and-treat systems.

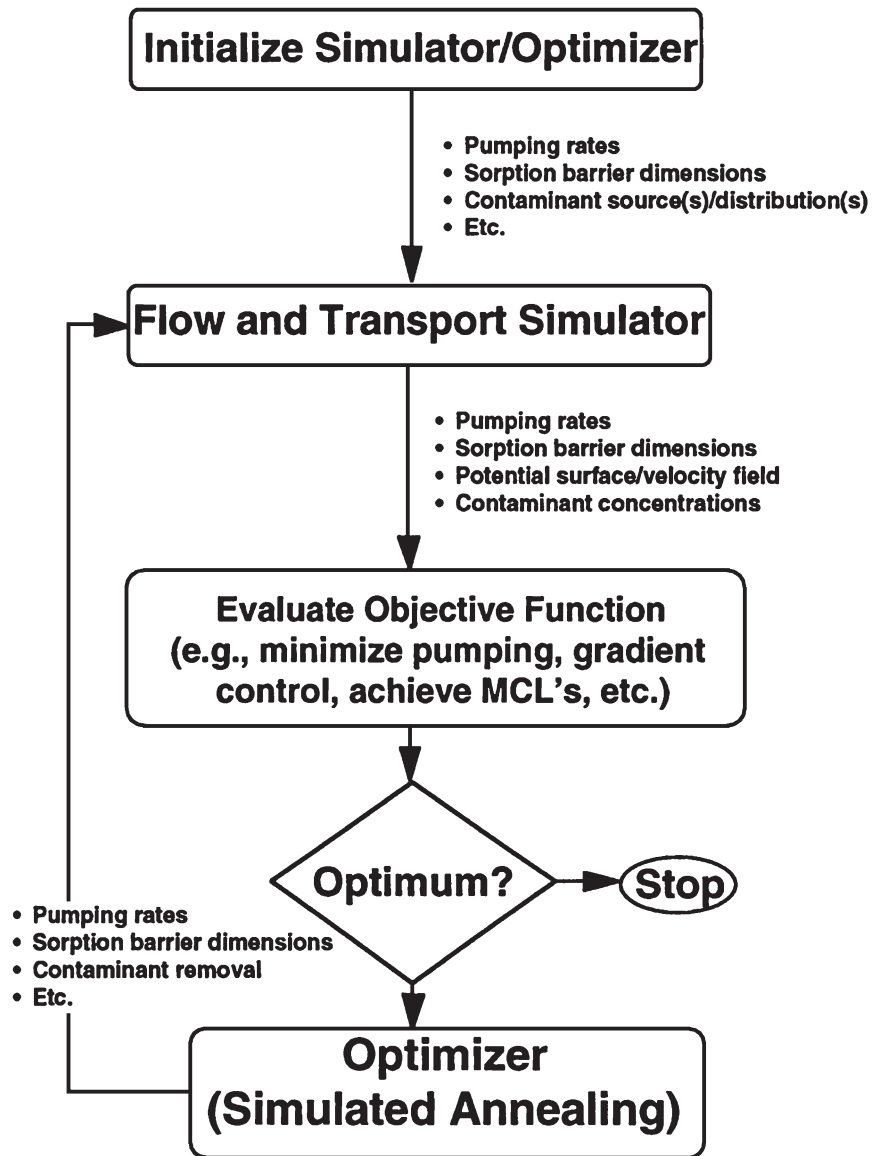


FIGURE 8.4 Flowchart of groundwater management methodology.

The combinations of barrier/pump-and-treat designs evaluated are summarized in Table 8.1. As indicated, the performance of 11 individual configurations was assessed. The 50-gal/min (gpm) system used one injection and one extraction well each operating at 50 gal/min. The 100-gpm system used two injection and two extraction wells operating at 50 gpm and the 180-gpm system used two injection wells operating at 90 gpm and three extraction wells operating at 60 gpm. The total rates of extraction and injection were equal such that there was no net withdrawal from the aquifer. Locations

8.16 CHAPTER EIGHT

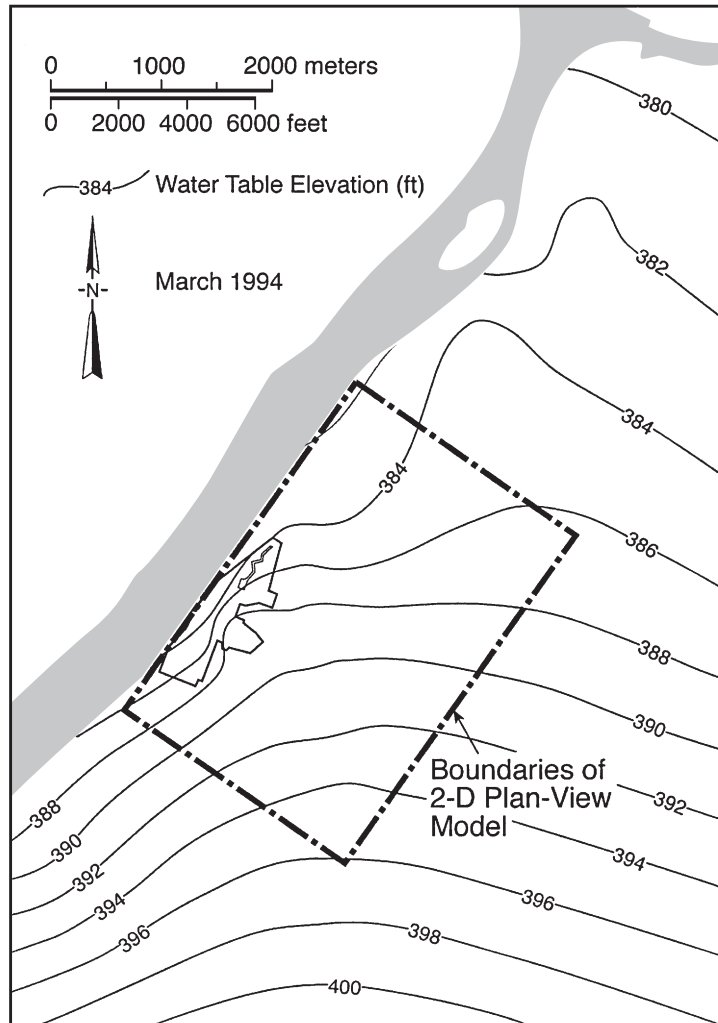


FIGURE 8.5 Plan view of N-springs study area. (From U.S. DOE, 1995.)

for the endpoints for the barrier designs and the extraction and injection wells analyzed in the U.S. DOE (1995) study are shown in Fig. 8.6.

The emphasis of the U.S. DOE (1995) study was in evaluating the performance of the barriers with and without pump-and-treat. The study did not report results for any systems using pump-and-treat alone. The findings from the study indicate that locating an extraction system in the area of highest ^{90}Sr produces only a minor decrease in total loading of ^{90}Sr to the Columbia River compared to loadings with a barrier. The barriers sufficiently increase the travel paths and times of the highly retarded ^{90}Sr , that release to the river is dramatically decreased. The findings of the U.S. DOE (1995) study are summarized in Table 8.2. The results indicate that without remediation, the releases to the river over a 10-yr period total 3.2 Ci. With 2000-ft, 3000-ft, and 3800-ft barriers in place, the releases decrease to 0.8 Ci, 0.1 Ci, and 0.0 Ci, respectively. With the well systems in place, the total for the

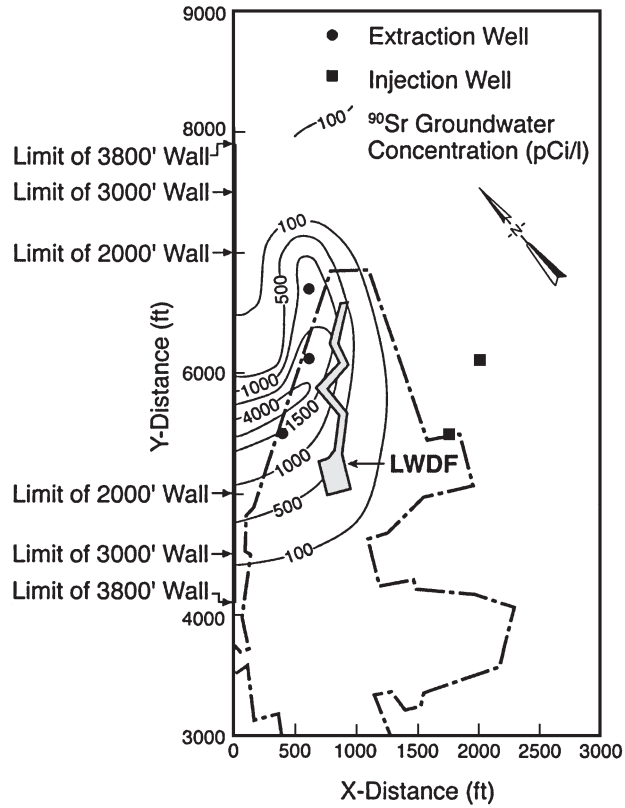


FIGURE 8.6 Areal distribution of ⁹⁰Sr in the groundwater at the N-Springs area.

TABLE 8.1 Summary of Barrier and Pump-and-Treat Designs Evaluated by U.S. DOE

Wall length, ft	Number of wells	50 gal/min	100 gal/min	180 gal/min
0	X	—	—	—
2000	X	X	—	X
3000	X	X	—	X
4000	X	X	X	X

Source: From U.S. DOE (1995).

2000-ft wall shows no change for a 50-gpm pumping system and a 25 percent reduction with a 180-gpm pumping system. Pumping had no apparent effects on the results for the 3000-ft and 3800-ft barrier releases.

These results strongly indicate that with a barrier in place, little if any incremental benefit is derived from adding the specified pump-and-treat configurations to the containment system. Therefore, the focus of the groundwater management methodology demonstration is on selecting an optimal design for eliminating ⁹⁰Sr releases to the Columbia River over a 10-yr period with or without a barrier in place. The possible configurations for the design will include no barrier or one of the

8.18 CHAPTER EIGHT

TABLE 8.2 Summary of 10-Year ⁹⁰Sr Releases (Ci) to the Columbia River for Barrier and Pump-and-Treat Designs by U.S. DOE

Design configuration	Estimated 10-year ⁹⁰ Sr release, Ci
No barrier	3.2
2000-ft barrier	0.8
2000-ft barrier with wells (50 gpm)	0.8
2000-ft barrier with wells (180 gpm)	0.6
3000-ft barrier	0.1
3000-ft barrier with wells (50 gpm)	0.1
3000-ft barrier with wells (180 gpm)	0.1
3800-ft barrier	0.0
3800-ft barrier with wells (50 gpm)	0.0
3800-ft barrier with wells (100 gpm)	0.0
3800-ft barrier with wells 180 gpm)	0.0

Source: From U.S. DOE (1995).

three lengths analyzed by the U.S. DOE study, and one or more extraction or injection wells selected from 28 candidate locations. Details of the flow and transport models and formulation of the optimization problem are discussed below.

8.7.3 Flow and Transport Model

The model area for the demonstration is the same used by the U.S. DOE (1995) analysis as shown in Fig. 8.5. The model encompasses the 8000-ft by 12,000-ft study area. As shown, the prevailing hydraulic gradient is almost due north, from the lower right-hand corner of the model area to the upper left-hand corner. As groundwater approaches the river it turns northwesterly. Over the study area the gradient varies from 0.0005 to 0.003, producing average groundwater velocities between 0.1 to 2 ft/day in the vicinity of the LWDF. The uppermost stratigraphic unit at the site is the Hanford formation, which consists of uncemented gravels with sand and silt interbeds. The Hanford formation extends to just above the water table and transitions to the Ringold formation.

The U.S. DOE (1995) study was based on a two-dimensional finite-difference code with the 8000- by 12,000-ft study area discretized using a variable grid spacing consisting of 58 nodes in the *y*-direction and 38 nodes in the *x*- direction. The flow model assumed steady-state and transport was determined using particle tracking based on computed groundwater pathlines and accounting for retardation and radioactive decay. The site was discretized using a 35 by 68 element grid. In the vicinity of the ⁹⁰Sr plume (from 3000 to 9000 ft in the *y*-direction and 0 to 3000 ft in the *x*-direction) a subgrid, regularly spaced with element dimensions of 100 by 100 ft was used. Outside the plume area, element dimensions increased from 250 to 1500 ft. An overlay of the subgrid over the model area is shown in Fig. 8.7. The model parameters used for the demonstration were either those used in the U.S. DOE (1995) study or inferred from information provided. The parameters used are summarized in Table 8.3. The hydraulic conductivity value is 261 ft/day, the geometric mean of five pumping test results ranging from 100 to 500 ft/day. Barriers are represented in the model by simply reducing the hydraulic conductivity in selected elements to 0.0001 ft/day. The porosity used was 0.25. The aquifer bottom was assumed to be at elevation 365.0, which results in an average aquifer thickness of about 25.0 ft. The boundaries were all assumed to be fixed-head boundaries with values varying linearly between elevations estimated from the water table map in Fig. 8.5. Elevations at the northeast and southeast boundaries were 386.6 and 396.0 ft, respectively. Elevations along the river varied from 381.3 to 384.4 ft.

Initial conditions for the transport analyses were based on an element-by-element discretization of the plume shown in Fig. 8.7. Travel times to the river were determined considering advection only. The travel times were then adjusted for retardation based on the following equation

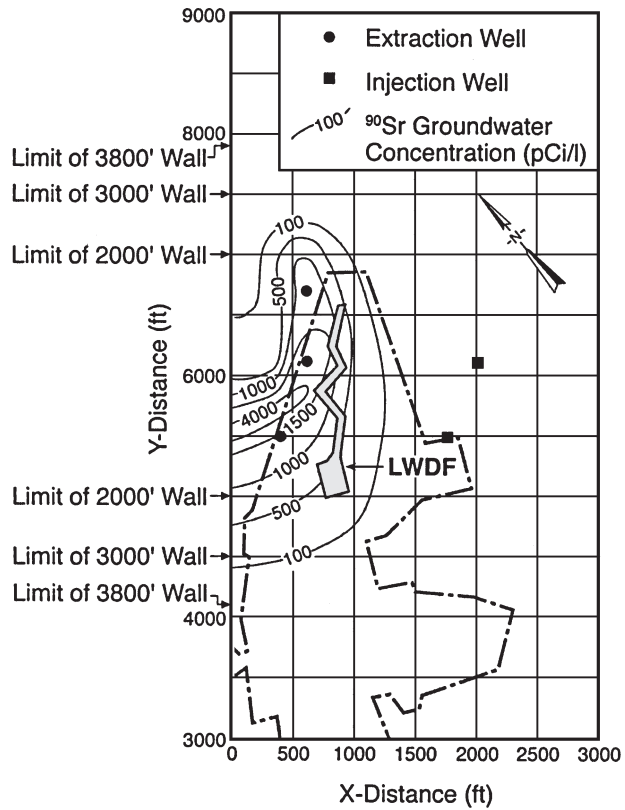


FIGURE 8.7 Overlay of 100 ft by 100 ft subgrid on the study area.

TABLE 8.3 Summary of Parameters for N-Springs Area Model

Parameter	Value
Hydraulic conductivity, K (ft/day)	261.0
Aquifer thickness, b (ft)	~25.0
Porosity, n	0.25
^{90}Sr distribution coefficient, K_d (mL/g)	15.5
Retardation factor	100.2
^{90}Sr half-life (yrs)	28

$$R_f = \frac{v}{v_c} = 1 + \frac{\rho_b}{\eta} K_d \quad (8.24)$$

where R_f = retardation factor

v and v_c = groundwater velocity and contaminant velocity, respectively

ρ_b = bulk distribution

η = porosity

K_d = distribution coefficient

8.20 CHAPTER EIGHT

Mass was then adjusted for radioactive decay using

$$\text{Mass} = \text{mass } 0.5^{(T/H_L)} \quad (8.25)$$

where T is the contaminant travel time and H_L is the half-life for ^{90}Sr .

The flow and transport models for the N-Springs area were validated based on comparisons with results obtained in the U.S. DOE (1995) study for flow and ^{90}Sr transport with no barrier and no wells. A particle released at x - y coordinates of (2000 ft, 6000 ft) had a computed groundwater travel time of 1579 days. The comparable value obtained from pathlines determined in the U.S. DOE study, is approximately 1500 days. Total release of ^{90}Sr to the Columbia River in 10 years was estimated to be 3.6 Ci. The 10-year release determined by the U.S. DOE study was 3.2 Ci. These results suggest that the groundwater management methodology flow and transport models are reasonably simulating ^{90}Sr transport in the N-Springs area.

8.7.4 Optimization Problem Formulation

The management problem under consideration for the N-Springs area is selection of a least-cost containment system that will reliably prevent any release of ^{90}Sr into the Columbia River for a 10-year period. Factors evaluated include construction costs for a barrier and/or well(s), treatment costs for any extracted contaminated groundwater, plus maintenance and operating costs for both extraction and injection wells. In addition, the net groundwater withdrawal should be zero (i.e., total extraction rate equals total injection rate). The configuration space includes 28 candidate well locations as shown in Fig. 8.8. There are 12 possible flow rates for each of the wells: five injection rates (-125 , -100 , -75 , -50 , and -25 gpm), zero, and six extraction rates (25, 50, 75, 100, 125, and 150 gpm). As with the U.S. DOE study, four possible barrier configurations were considered: no barrier and candidate lengths of 2000, 3000, and 3800 ft. The endpoints for each of the barriers are noted in Fig. 8.9.

Approximate costs for the Hanford site are summarized in Table 8.4. In addition to the cost elements, other considerations are penalties associated with not achieving the stated objective of zero ^{90}Sr releases to the river, and not balancing the extraction and injection rates. The concerns about releasing radionuclides to the Columbia River are severe, so a high penalty is necessary to ensure that the search emphasize solutions that will achieve that objective. Also, the need for additional treatment or disposal of excess extracted water, or the need to develop a source for additional injection water, make the balancing of extraction and injection rates also very important.

Based on the above, the mathematical formulation for the optimal design of an N-Springs containment system is as follows:

$$\min \sum_{i=1}^{N_w} C_{Q^+} Q_i^+ + \sum_{i=1}^{N_w} C_{Q^-} Q_i^- + \sum_{i=1}^{N_w} I_i + C_B L_B + P_1 \Delta Q + P_2 \text{mass} \quad (8.26)$$

where C_{Q^+} = cost per gpm for treating extracted groundwater for each extraction well i

Q_i^+ = pumping rate (gpm) for extraction well i

C_{Q^-} = cost per gpm for groundwater injection for each injection well i

Q_i^- = pumping rate (gpm) for injection well i

I_i = construction costs for well i if $Q_i \neq 0$

C_B = cost/ft of barrier construction

L_B = barrier length (ft)

P_1 = penalty for not achieving balanced extraction and injection rates

ΔQ = absolute value of difference in total injection and withdrawal rates (gpm)

P_2 = penalty for release of ^{90}Sr to the river

mass = computed mass of ^{90}Sr released to the river (Ci)

In addition to the estimated costs, another concern in selecting coefficient values for Eq. (8.26) is scaling the components of the cost function such that the desired emphasis is achieved, and there is adequate contrast in cost function values between good solutions and bad solutions. With this in mind,

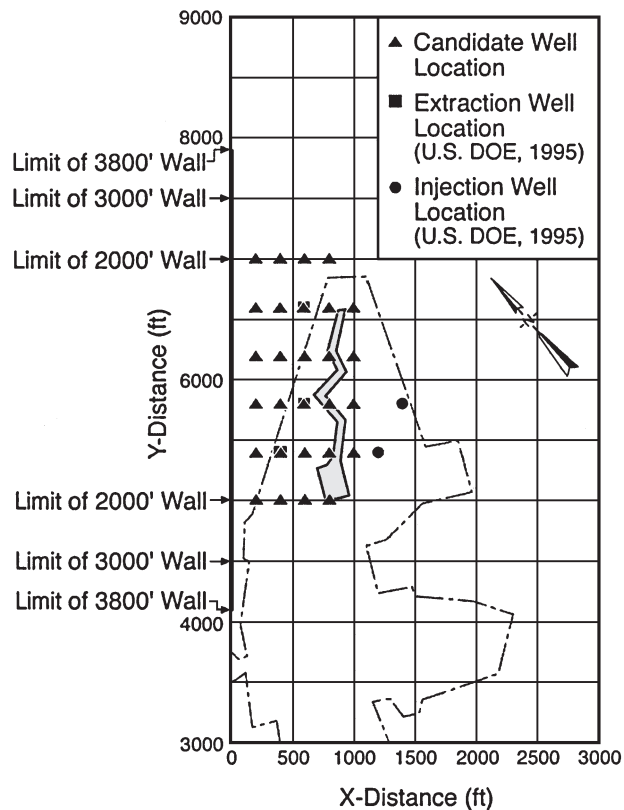


FIGURE 8.8 Well and barrier locations for N-Springs area.

the actual unit costs for well construction, groundwater treatment, and barrier construction were reduced by a factor of 1000. The coefficient for injection was set equal to 1.0, and the penalty factors $P1$ and $P2$ were both set equal to 5000. The resultant values are listed in Table 8.4.

The optimal solution obtained by the groundwater management methodology consists of a 3000-ft barrier with a single withdrawal well and a single injection well. The solution required 65,250 iterations, 30,883 objective function evaluations, and 1.5 CPU processing hours. The withdrawal well has a pumping rate of 25 gpm and is located at coordinates (6200 ft, 400 ft), on the downstream edge of the 4000 pCi/L contour of the ^{90}Sr plume. The injection well, also pumping at 25 gpm, is located almost immediately upgradient of the withdrawal well at coordinates (6600 ft, 800 ft). The water table produced by the design is shown in Fig. 8.9. The vertical scale is exaggerated, so the figure clearly depicts the disruption in the general gradient produced by the 3000-ft barrier at the river. This disruption significantly increases the travel time for ^{90}Sr to the river. In addition, the withdrawal well at the downstream edge of the plume is well positioned to intercept the highest concentration flows or at least incrementally impede their migration. The effect of the injection well immediately upgradient of the withdrawal well is likely to redirect the high concentration flows toward the withdrawal well.

Computed contaminant releases to the river are virtually zero until year 11 when the cumulative mass release is 0.001 Ci. The objective value cost elements associated with the design are summarized in Table 8.5. Given that the withdrawal and injection pumping rates are balanced, the only cost elements are installation costs for two wells, treatment costs for 25 gpm capacity, and construction of the 3000-ft barrier for a total of \$17.6 million. Applying the same cost factors to the U.S. DOE (1995) analysis of

8.22 CHAPTER EIGHT

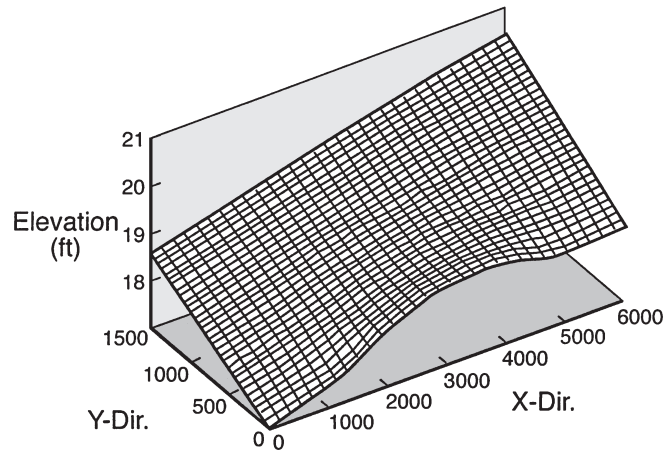


FIGURE 8.9 N-Springs area water table produced by optimal design.

TABLE 8.4 Summary of Estimated Construction and Operation Costs for ^{90}Sr at N-Springs

Cost element	Parameter	Value
Well construction: \$50,000/well	I_i	50.0
Groundwater treatment: \$220,000/gpm	C_{Q^+}	220.0
Injection well operation	C_{Q^-}	1.0
Barrier construction: \$4,000/ft	C_B	4.0
Penalty for excess extraction/withdrawal	P_1	5,000
Penalty for ^{90}Sr releases	P_2	5,000

TABLE 8.5 Summary of Optimal Design Costs

Cost element	Optimal design	U.S. DOE design
Well construction: \$50,000/well	\$0.1 million (2 wells)	\$0.1 million (2 wells)
Groundwater treatment: \$220,000/gpm	\$5.5 million (25 gpm)	\$11.0 million (50 gpm)
Barrier construction: \$4,000/ft	\$12.0 million (3,000 ft)	\$12.0 million (3,000)
Penalty for excess extraction/withdrawal	\$0.0	\$0.0
^{90}Sr releases	0.0 Ci	0.1 Ci
Total cost	\$17.6 million	\$23.1 million

the system consisting of a 3000-ft barrier and 50 gpm pumping (the lowest pumping rate analyzed) the comparable total cost would be \$23.1 million. In addition, the U.S.DOE design results in the release of 0.1 Ci of ^{90}Sr to the river within the 10-year study period. Although the wells are pumping at higher rates, their locations are less effective in intercepting or diverting the migration of ^{90}Sr .

8.8 SUMMARY AND CONCLUSIONS

The reliability of simulated annealing in locating global or near-global optima is a major advantage. Other noted advantages are its flexibility in considering complex, highly nonlinear or discontinuous

objectives and/or constraints, and the ease with which it can be interfaced with application-specific simulators. However, the limited experience to date in applying simulated annealing to groundwater management problems highlights the large number of objective function evaluations required for convergence to a quality solution. Enhanced annealing departs from the purely random search approach inherent in simulated annealing by incorporating directional search and memory capabilities. These new capabilities compound the improvements in computational efficiency previously achieved, demonstrating the potential gains that accrue from combining concepts of related combinatorial optimization methods. Memory concepts derived from the tabu search method show particular promise for improving the rate and quality of convergence. Also, selecting search directions based on better understanding of the current neighborhood of the configuration space was shown to improve algorithm performance.

The capabilities of the groundwater management methodology with enhanced annealing were demonstrated by applying it to an actual contaminated groundwater site. The analysis considered optimal design of a contaminated groundwater containment system using pump-and-treat in conjunction with physical barriers. The flow and transport simulators proved to adequately represent important processes at the study site and optimal results were obtained that are consistent with previously reported findings. The application confirmed the ease with which simulated annealing can be adapted to new problem formulations and/or objective and constraint sets.

REFERENCES

- Anderson, K. and R. V. V. Vidal (1993). "Solving the Quadratic Assignment Problem," in *Applied Simulated Annealing*, René V. V. Vidal, ed., Springer-Verlag, New York.
- Chardaire, P. and J. L. Lutton (1993). "Using Simulated Annealing to Solve Concentrator Location Problems in Telecommunication Networks," in *Applied Simulated Annealing*, René V. V. Vidal, ed., Springer-Verlag, New York.
- Davis, L. and M. Steenstrup (1987). "Genetic Algorithms and Simulated Annealing: An Overview," in *Genetic Algorithms and Simulated Annealing*, L. Davis, ed., Morgan Kaufman, Los Altos, Calif.
- Dougherty, D. E. and R. A. Marryott (1991). "Optimal Groundwater Management, 1. Simulated Annealing," *Water Resources Research*, 27(10):2493–2508.
- Glover, R. (1989). "Tabu Search—Part I," *ORSA Journal on Computing*, 1(3):190–206.
- Glover, R. (1990). "Tabu Search—Part II," *ORSA Journal on Computing*, 2:4–32.
- Glover, F. (1994a). "Genetic Algorithms and Scatter Search: Unsuspected Potentials," *Statistics and Computing*, 4:131–140.
- Glover, F. (1994b). "Tabu Search for Nonlinear and Parametric Optimization (with Links to Genetic Algorithms)," *Discrete Applied Mathematics*, 49:231–255.
- Glover, F. (1995). "Tabu Search Fundamentals and Uses," CAAI Research Report, University of Colorado.
- Greene, J. W. and K. J. Supowit (1986). "Simulated Annealing Without Rejected Moves," *IEEE Transactions on Computer-Aided Design*, CAD-5(1):221–228.
- Hammer, P. L., ed. (1993). "Tabu Search," *Annals of Operations Research*, 41.
- Ingber, L. (1993). "Simulated Annealing: Practice Versus Theory," *Mathematical Computer Modeling*, 18(11):29–58. Kaufmann, Los Altos, Calif.
- Kirkpatrick, S., C. D. Gelatt, Jr., and M. P. Vecchi (1983). "Optimization Simulated Annealing," *Science*, 220(4598):680.
- Laarhoven, P. J. M. van (1988). *Theoretical and Computational Aspects of Simulated Annealing*, Centre for Mathematics and Computer Science, Amsterdam, the Netherlands.
- Laarhoven, P. J. M. van and E. H. L. Aarts (1987). *Simulated Annealing: Theory and Applications*, Kluwer Academic, Norwell, Mass.
- Marryott, R. A., D. E. Dougherty, and R. L. Stollar (1993). "Optimal Groundwater Management, 2. Application of Simulated Annealing to a Field-Scale Contamination Site," *Water Resource Research*, 29(4):847–860.
- Metropolis, N., A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller (1953). "Equation of State Calculations by Fast Computing Machines," *Journal of Chemical Physics*, 21:1087–1092.

8.24 CHAPTER EIGHT

- Murtagh, B. A. and M. A. Saunders (1980). "MINOS/AUGMENTED User's Manual I." *Tech. Rep.* 80-74, 51 pp., Syst. Optimization Laboratory Department of Operations Research, Stanford University, Stanford, Calif.
- National Research Council (1994). *Alternatives for Ground Water Cleanup*, National Academy Press, Washington D.C.
- Ritzel, B. J., J. W. Eheart, and S. Ranjithan (1994). "Using Genetic Algorithms to Solve a Containment Problem," *Water Resource Research*, 30(5):1589-1603.
- Rogers, L. L. and F. U. Dowla (1994). "Optimization of Groundwater Remediation Using Artificial Neural Networks with Parallel Solute Transport Modeling," *Water Resource Research*, 30(2):457-481.
- Skaggs, R. L. (1995). "Enhanced Annealing Applied to Groundwater Restoration Management," Ph.D. Dissertation, 260 pp., Department of Civil Engineering, Arizona State University, Tempe, Ariz.
- Skaggs, R. L., L. W. Mays, and L. W. Vail (2001a). "Simulated Annealing with Memory and Directional Search for Ground Water Remediation Design," *Journal of the American Water Resources Association*, 37(4): 853-866.
- Skaggs, R. L., L. W. Mays, and L. W. Vail (2001b). "Application of Enhanced Annealing to Ground Water Remediation Design," *Journal of the American Water Resources Association*, 37(4): 867-875.
- U.S. Department of Energy (1995). "Modeling Evaluation of N-Springs Barrier and Pump-and-Treat System," DOE/RL-94-132, Rev. 0, Richland, Wash.
- Vidal, R. V. V. (1993). "Optimal Partition of an Interval—The Discrete Version," in *Applied Simulated Annealing*. R. V. V. Vidal, ed., Springer-Verlag, New York.
- Wanakule et al. (1986).
- Woodruff, D. L. and E. Zemel (1993). "Hashing Vectors for Tabu Search," *Annals of Operations Research*, 41.