1.1A GENETIC ALGORITHMS AND THEIR APPLICATIONS IN ENVIRONMENTAL SCIENCES

Sue Ellen Haupt *
Randy L. Haupt

Utah State University, Logan, UT

1. INTRODUCTION

The genetic algorithm (GA) is finding wide acceptance in many disciplines. This paper introduces the elements of GAs and their application to environmental science problems.

The genetic algorithm is an optimization tool that mimics natural selection and genetics. The parameters to be optimized are the genes, which are strung together in an array called a chromosome. A population of chromosomes is created and evaluated by the cost function, with the "most fit" chromosomes being kept in the population while the "least fit" ones are discarded. The chromosomes are then paired so they can mate, involving combining portions of each chromosome to produce new chromosomes. Random mutations are imposed. The new chromosomes are evaluated by the cost function and the process iterates. Thus the parameter space is explored by a combination of combining parts of the best solutions as well as extending the search through mutations. The trade-offs involved in selecting population size, mutation rate, and mate selection are briefly discussed below.

The key to using GAs in environmental sciences is to pose the problem as one in optimization. Many problems are quite naturally optimization problems, such as the many uses of inverse models in environmental science. Other problems can be manipulated into optimization form by careful definition of the cost function, so that even nonlinear differential equations can be approached using GAs. Examples of both the natural type as well as those contrived into an optimization form are presented.

GAs are well suited to many optimization problems where more traditional methods fail. Some of the advantages they have over conventional numerical optimization algorithms are that they:

- Optimize with continuous or discrete parameters,
- Don't require derivative information,
- Simultaneously search from a wide sampling of the objective function surface,
- Deal with a large number of parameters,
- Are well suited for parallel computers,
- Optimize parameters with extremely complex objective function surfaces,
- Provide a list of semi-optimum parameters, not just a single solution,
- May encode the parameters so that the optimization is done with the encoded parameters, and
- Works with numerically generated data, experimental data, or analytical functions.

These advantages outweigh the GAs' lack of rigorous convergence proofs.

In the following sections we give a short overview of how the GA works, briefly review some of the ways that GAs have been used in environmental science, and present an example application that demonstrates the strength of the GA on an inverse problem.

2. INTRODUCTION TO GENETIC ALGORITHMS

John Holland is often referred to as the "father of genetic algorithms." He developed this brand of genetic programming during the 1960's and 1970's and his work is described in

* Corresponding author address: Sue Ellen Haupt, Department of Mechanical and Aerospace Engineering, 4130, Utah State University, Logan, UT 84322-4130; e-mail: suehaupt@ece.usu.edu

his book (Holland 1975). His student, David Goldberg, popularized the method by solving a difficult problem involving the control of gas-pipeline transmission for his dissertation (see Goldberg 1989). Since that time, they have been applied to a wide variety of problems, including those described above.

The following explanation follows the flow chart in Figure 1. The first step is defining an objective function with inputs and outputs. A binary GA encodes the value of each input parameter (e.g. *a, b, c, d*) as a binary number. The parameter values are then placed side-by-side in an array known as a chromosome.

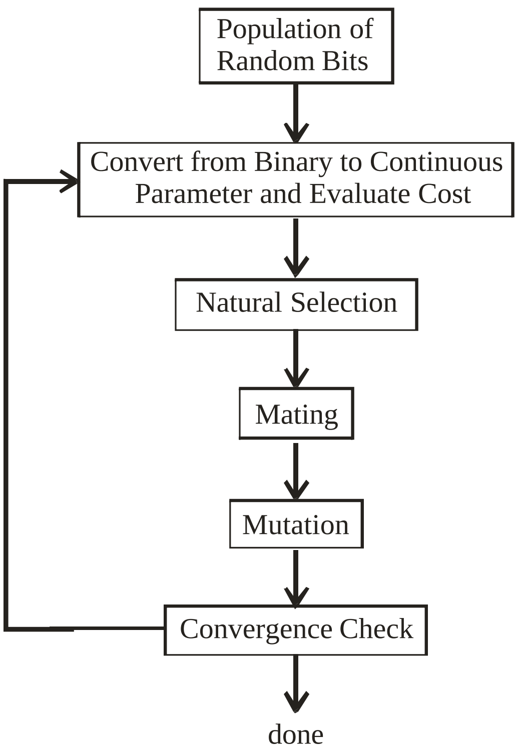

Figure 1: Flow chart of Binary Genetic Algorithm

A population is a matrix with each row representing a chromosome. The algorithm begins with a population consisting of random ones and zeros (see Figure 2).

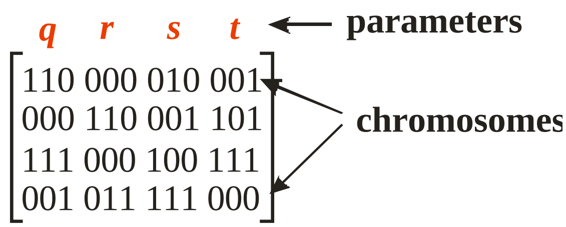

Figure 2. Initial population of binary coded parameters.

These random binary digits translate into guesses of values of the input parameters. Next, the binary chromosomes are converted to continuous values which are evaluated by the objective function. Mating takes place between selected chromosomes. Mates are randomly selected with a probability of selection greater for those chromosomes yielding desirable output from the objective function (tournament or roulette wheel selection). Offspring (new chromosomes) produced from mating inherit binary codes from both parents. A simple crossover scheme randomly picks a crossover point in the chromosome. Two offspring result by keeping the binary strings to the left of the crossover point for each parent and swapping the binary strings to the right of the crossover point, as shown in Figure 3. Crossover mimics the process of meiosis in biology. Mutations randomly convert some of the bits in the population from "1" to "0" or visa versa. The objective function outputs associated with the new population are calculated and the process repeated. The algorithm stops after finding an acceptable solution or after completing a set number of iterations.

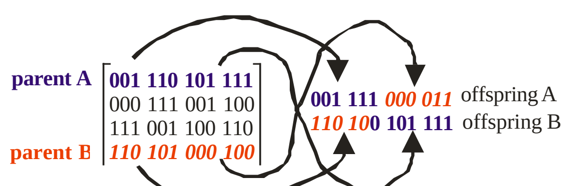

Figure 3. Crossover during the mating process.

Selecting the best population size, mating scheme, and mutation rate is still an area of controversy. Haupt and Haupt (1998, 2000) address some of these issues. Since the GA is a random search, a certain population size and mutation rate can give considerably different answers for different independent runs. A GA run will give a good answer found from a wide exploration of the search space but not necessarily the best answer.

Most real world optimization problems have multiple objectives. Multiple objectives can be handled by weighting and adding the fitness from each objective. Multi-objective optimization does not have a single optimum solution relative to all objectives. Instead, there are a set of optimal solutions, known as Pareto-optimal or non-inferior solutions. A Pareto GA attempts to find as many Pareto-optimal solutions as possible, since all these solutions have the same cost.

3. USES OF GENETIC ALGORITHMS IN ENVIRONMENTAL SCIENCE

There is a recognized need for better methods of optimization in the environmental sciences. For instance, many different problems involve fitting a model to observed data. Sometimes the data is a time series while other times it is an observed environmental state. Often, some general functional forms are known or surmised from the data. But frequently, the goal is to fit model parameters to optimize the match between the model and the data. Practitioners often go the next step and use the model to make predictions. The need for new tools involving Artificial Intelligence (AI) techniques, including Genetic Algorithms, is noted by Hart, et al. (1998) among others.

One example of fitting a model to observed data using a GA is reported by Mulligan and Brown (1998). They use a GA to estimate parameters to calibrate a water quality model. They used nonlinear regression to search for parameters that minimize the least square error between the best fit model and the data. They found that the GA works better than more traditional techniques plus noted the added advantage that the GA can provide information about the search space, enabling them to develop confidence regions and parameter correlations. Some other work related to water quality includes using GAs to determine flow routing parameters (Molian and Loucks 1995)) solving ground water management problems (McKinney and Lin 1994, Rogers and Dowla 1994, Ritzel, et al. 1994), sizing distribution networks (Simpson, et al. 1994), and calibrating parameters for an activated sludge system (Kim, et al. 2002).

Managing groundwater supplies has found AI and GAs useful. Peralta and collaborators have combined GAs with neural networks and simulated annealing techniques to combine the advantages of each. Aly and Peralta (1999a) used GAs to fit parameters of a model to optimize pumping locations and schedules for groundwater treatment. They then combined the GA with a neural network (NN) to model the complex response functions within the GA (Peralta and Aly 1999b). Shieh and Peralta (1997) combined Simulated Annealing (SA) and GAs to maximize efficiency and well use the easily applied parallel nature of the GA. Most recently, Fayad (2001) together with Peralta used a Pareto GA to sort optimal solutions for managing surface and groundwater supplies, together with a fuzzy-penalty function while using an Artificial Neural Network (ANN) to model the complex aquifer systems in the groundwater system responses.

Another example is the successful application of a GA to classification and prediction of rainy day versus non-rainy day occurrences by Sen and Oztopal (2001). They used the GA to estimate the parameters in a third order Markov model.

An example from geophysics is determining the type of underground rock layers. Since it is not practical to take core samples of sufficient resolution to create good maps of the underground layers, modern techniques use seismic information or apply a current and measure the potential difference which gives a resistance. These various methods produce an underdetermined multimodal model of the Earth. Fitting model parameters to match the data is regarded as a highly nonlinear process. Genetic algorithms have found recent success in finding realistic solutions for this inverse problem (Jervis and Stoffa 1993; Jervis, et al. 1996, Sen and Stoffa 1992a,b; Chunduru, et al. 1995, 1997; Boschetti, et al. 1995, 1996, 1997; Porsani, et al. 2000). Minister, et al. (1995) find that evolutionary programming is useful for locating the hypocenter of an earthquake, especially when combined with simulated annealing.

Another inverse problem is determining the source of air pollutants given what is known about monitored pollutants. Additional information includes the usual combination (percentages) of certain pollutants from different source regions and predominant wind patterns. The goal of the receptor inverse models is to target what regions, and even which sources contribute the most pollution to a given receptor region. This process involves an optimization. Cartwright and Harris (1993) suggest that a genetic algorithm may be a significant advance over other types of optimization models for this problem when there are many sources and many receptors.

Evolutionary methods have also found their way into oceanographic experimental design. Barth (1992) showed that a genetic algorithm is faster than simulated annealing and more accurate than a problem specific method for optimizing the design of an oceanographic experiment. Porto, et al. (1995) found that an evolutionary programming strategy was more robust than traditional methods for locating an array of sensors in the ocean after they have drifted from their initial deployment location.

Finally, Charbonneau (1995) gives three examples of uses of a genetic algorithm in astrophysics: modeling the rotation curves of galaxies, extracting pulsation periods of Doppler velocities in spectral lines, and optimizing a model of hydrodynamic wind.

4. EXAMPLE APPLICATION

Many of the applications reviewed above use a GA to fit parameters to a model based on data, we choose to demonstrate the utility of the GA on a specific inverse problem. In particular, we will begin with time series data from the predator-prey model (also known as the Lotka-Volterra equations), namely:

$$\frac{dx}{dt} = ax - bxy \qquad (1)$$
$$\frac{dy}{dt} = -cy + dxy$$

where x is the number of prey and y the number of predators. The prey growth rate is a while the predator death rate is c. Parameters b and d characterize the interactions. Equations (1) were integrated using a fourth order Runge Kutta with a time step of 0.01 and parameters a =1.2, b=0.6, c= 0.8, and d=0.3. The time series showing the interaction between the two appears as Figure 4. This time series serves as the data for computing the inverse models below.

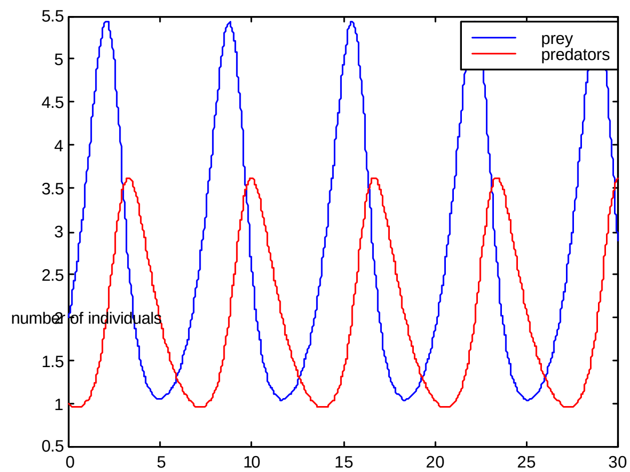

Figure 4. Time series showing predator and prey variations over time according to equation (1).

The phase space plot is Figure 5 where we see the limit cycle between the predators and the prey.

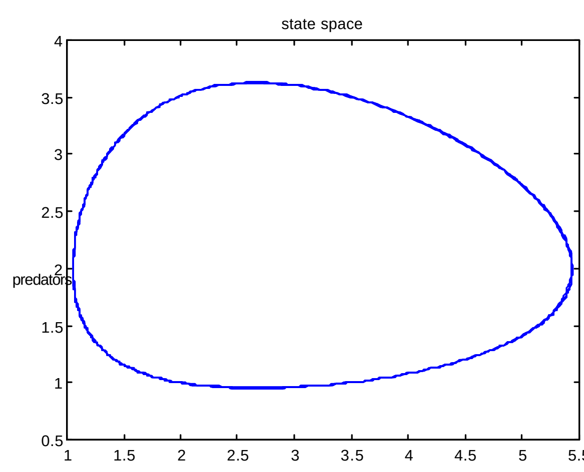

Figure 5. State space showing predator-prey interactions.

A standard linear least squares model fit would be of the form:

$$s_t = Ls + C \quad (2)$$

where s is a vector incorporating both x and y, L is a linear matrix operator, and C is the additive constant. This simple linear form is easily fit using standard analytical techniques to minimize the least square error between the model and data. The least squares fit to the linear model appears in Figure 6. We note that the agreement is quite poor, as one would expect given that the system (1) is highly nonlinear. With no nonlinear interaction available, the number of prey grows while the number of predators remains stationary.

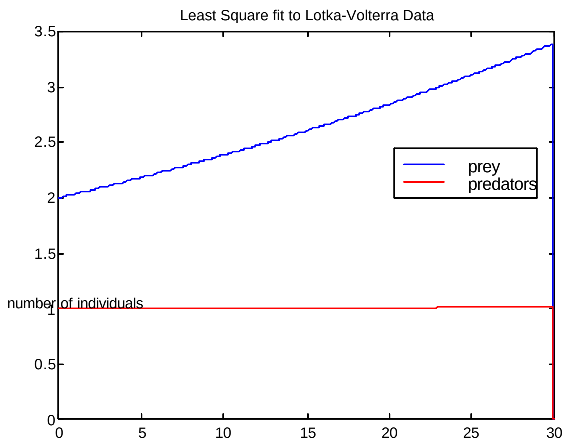

Figure 6. Least squares time series fit to predator-prey model.

To obtain a more appropriate nonlinear fit, we now choose to model the data with a nonlinear model:

$$s_t = Ns^T s + Ls + C \quad (3)$$

We now allow nonlinear interaction through the nonlinear third order tensor operator, N. Although one can still find a closed form solution for this nonlinear problem, it involves inverting a fourth order tensor. For problems larger than this simple two-dimensional one, such an inversion is not trivial. Therefore, we choose to use a genetic algorithm to find parameters which minimize the least square error between the model and the data. The GA used an initial population size of 200, a working population size of 100, and a mutation rate of 0.2. A time series of the solution as computed by the GA appears in Figure 7. Note that although the time series does not exactly reproduce the data, the oscillations with a phase shift of roughly a quarter period is reproduced. The wavelength is not exact and the amplitudes grow in time, indicating an instability. This instability is likely inherent in the way that the model is matched. However, the reproduction of such a difficult nonlinear system is amazing given the comparison to traditional linear models.

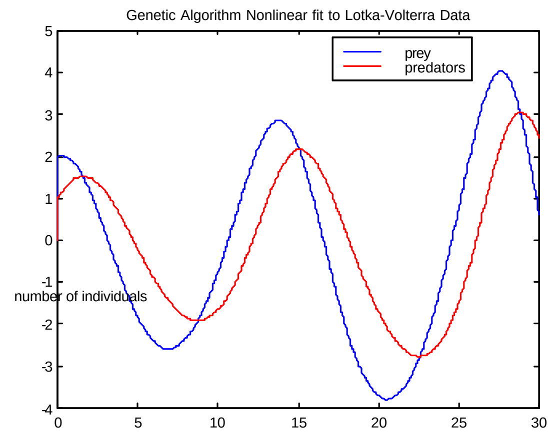

Figure 7. Time series of predator-prey interactions as computed by the genetic algorithm.

The state space plot appears in Figure 8. Once again, the limit cycle is not actually reproduced. The nonlinear model instead appears unstable and slowly grows. However, the comparison with the linear least squares model resulted in merely a single slowly growing curve (not shown). The GA nonlinear model was able to capture the cyclical nature of the oscillations.

Finally, Figure 9 shows the convergence of the GA for a typical run of fitting the nonlinear model (3) to the data. Note that due to their random nature, the results of the GA are never exactly the same. In particular the convergence plots will differ each time. However, it is amazing how the results are so reliable.

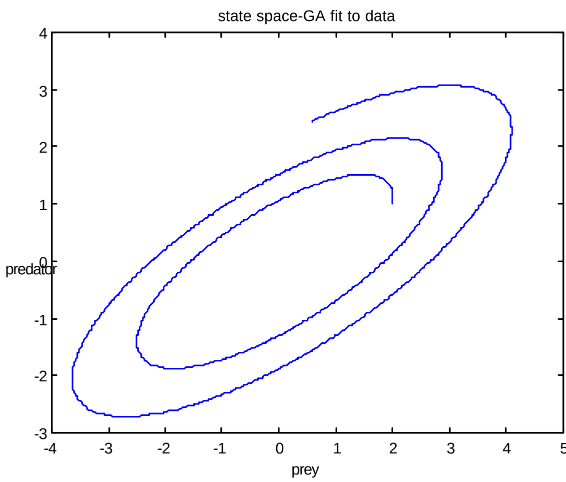

Figure 8. The predator-prey relation in state space as computed by the nonlinear model with parameters fit by the GA.

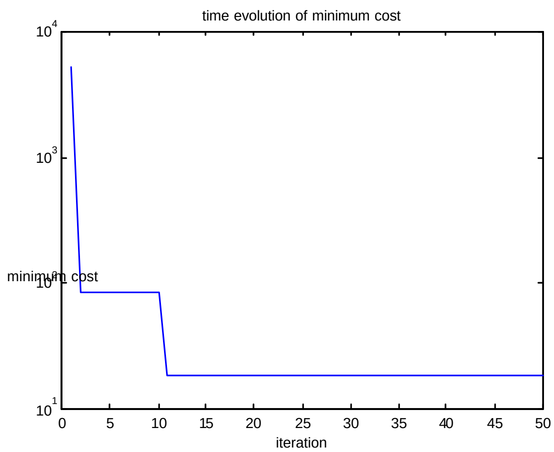

Figure 9. Evolution of the minimum cost for the GA fit to the nonlinear model parameters.

6. CONCLUSIONS

We have shown that genetic algorithms are not only an effective way of solving optimization problems, but they can also be rather fun to apply. They have begun to find their way into applications in the environmental sciences as cited above, but their strengths have only begun to be tapped. We have demonstrated here how versatile these algorithms are at finding solutions where other methods often fail. We saw that for a simple two-dimensional nonlinear system describing predator-prey relations, the GA was able to fit the parameters of a nonlinear model so that the attractor was much better produced than by a traditional linear least squares fit. Although the match is not perfect, the nonlinear GA model captured the essence of the dynamics.

Here, we have only discussed binary genetic algorithms and their most direct applications to optimization problems. The companion paper (Haupt 2003) describes the version of the GA encoded in terms of floating point numbers and describes its application in more complex problems. We show there how to pose boundary value problems in terms amenable to minimization and show how genetic algorithms can be effective at finding solutions to highly nonlinear partial differential equations. In additions, we show variations of the inverse type problem described here where a highly nonlinear system of equations can be stochastically modeled if the parameters are fit using a GA.

The hope is that this work has whet the reader's appetite and that the GA will find its way into other interesting problems. Our goal is to inspire other environmental scientists to try the GA on problems that arise in optimization.

REFERENCES

Aly, A.H. and R.C. Peralta, 1999a: Comparison of a genetic algorithm and mathematical programming to the design of groundwater cleanup systems, *Water Resources Research*, **35**(8), pp. 2415-2425.

Aly, A.H. and R.C. Peralta, 1999b: Optimal design of aquifer clean up systems under uncertainty using a neural network and a genetic algorithm, *Water Resources Research*, **35**(8), pp. 2523-2532.

Barth, H., 1992: Oceanographic Experiment Design II: Genetic Algorithms, *Journal of Oceanic and Atmospheric Technology*, **9**, 1992, pp. 434-443.

Boschetti, F., Dentith, M.C. and List, R. D., 1995: A staged genetic algorithm for tomographic inversion of seismic refraction data. *Exploration Geophysics* **26**, pp 331-335.

Boschetti, F., Dentith, M.C. and List, R. D., 1996: Inversion of seismic refraction data using genetic algorithms. *Geophysics* **61**, pp 1715-1727.

Boschetti, F., Dentith, M.C. and List, R., 1997. Inversion of potential field data by genetic algorithms. *Geophysical Prospecting.***45**, pp 461-478.

Cartwright, H.M. and S.P. Harris, 1993: Analysis of the Distribution of Airborne Pollution using Genetic Algorithms, *Atmospheric Environment*, Part A, **27A**, pp. 1783-1797.

Chambers, L. ed., 1995: *Genetic Algorithms, Applications Volume I,* New York: CRC Press.

Charbonneau, P., 1995: 'Genetic Algorithms in Astronomy and Astrophysics, *The Astrophysical Journal Supplement Series*, **101**, 309-334.

Chunduru, R.K., M.K. Sen, P.L. Stoffa, and R. Nagendra, 1995: 'Non-linear Inversion of Resistivity Profiling Data for some Regular Geometrical Bodies,' *Geophysical Prospecting*, 43, pp. 979-1003.

Chunduru, Raghu K., Mrinal K. Sen, and Paul L. Stoffa, 1997: Hybrid optimization for geophysical inversion, *Geophysics*, **62**(4), 1196–1207.

Fayad, H. 2001: Application of neural networks and genetic algorithms for solving conjunctive water use problems, Ph.D. Dissertation, Utah State University, 152 pp.

Goldberg, D.E. 1989: *Genetic Algorithms in Search, Optimization, and Machine Learning*, New York: Addison-Wesley.

Hart, J., I. Hunt, V. Shankararaman, 1998: Environmental management systems – a role for AI?, ECAI 98 W7 Binding Envronmental Sciences and AI.

Haupt, R.L. and S.E. Haupt. 1998: *Practical Genetic Algorithms*, John Wiley & Sons, New York, 177pp.

Haupt, R.L. and S.E. Haupt, 2000: Optimum population size and mutation rate for a simple real genetic algorithm that optimizes array factors, *Applied Computational Electromagnetics Society Journal*, Vol**.** **15**, No. 2.

Haupt, S.E. 2003: Genetic Algorithms in Geophysical Fluid Dynamics, AMS Conference on Artificial Intelligence, Paper P1.7.

Holland, J.H., 1992: Genetic algorithms, *Sci. Amer*., July, pp. 66-72.

Jervis, M., M.K. Sen, and P.L. Stoffa, 1996: Prestack Migration Velocity Estimation using Nonlinear Methods, *Geophysics,* **60**, pp. 138-150.

Kim, S., H. Lee, J. Kim, C. Kim, J. Ko, H. Woo, and S. Kim, 2002: Genetic algorithms for the application of Activated Sludge Model No. 1, *Water Science and Technology*, **45** (4-5), pp. 405-411.

McKinney, D.C. and M.-D. Lin, 1993: Genetic algorithm solution of ground water management models, Water Resources Research, 30(6), pp. 3775-3789.

Michalewicz, Z. 1992: *Genetic Algorithms + Data Structures = Evolution Programs*, New York: Springer-Verlag.

Minister, J-B. H., N.P. Williams, T.G. Masters, J.F. Gilbert, and J.S. Haase, 1995: Application of evolutionary programming to earthquake hypocenter determination, in *Evolutionary Programming: Proc. Of the Fourth Annual Conference on Evolutionary Programming*, pp. 3-17.

Mohan, S. and Loucks, D.P., 1995: Genetic algorithms for estimating model parameters, *Integrated Water Resour. Plng. For the 21st Century, Proc. Of the 22nd Annu. Conf*, ASCE, Cambridge, MA.

Mulligan, A.E. and L.C. Brown, 1998: Genetic algorithms for calibrating water quality models, *J. of Environmental Engineering*, pp. 202-211.

Porsani, M. J., P. L. Stoffa, M. K. Sen, and R. K. Chunduru, 2000: Fitness functions, genetic algorithms and hybrid optimization in seismic waveform inversion, *J. Seismic Explor*., **9**, 143-164, 2000

Porto, V.W., D.B. Fogel, and L.J. Fogel, 1995: Alternative neural network training methods, *IEEE Expert Syst*, June.

Ritzel, B.J., J.W. Eheart, and S. Rajithan, 1994: Using geteic algorithms to solve a multiple objective groundwater pollution containment problem, *Water Resources 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 Resources Research*, **30**(2), pp. 457-481.

Sen, M.K. and P.L. Stoffa, 1992: Rapid Sampling of Model Space using Genetic Algorithms: Examples from Seismic Waveform Inversion, *Geophys. J. Int.*, **108**, pp. 281-292.

Sen, M.K. and P.L. Stoffa, 1992: Genetic Inversion of AVO, *Geophysics: The Leading Edge of Exploration*, pp. 27-29.

Sen, M.K. and P.L. Stoffa, 1996: Bayseian Inference, Gibbs' Sampler and Uncertainty Estimation in Geophysical Inversion, *Geophysical Prospecting*, **44**, pp. 313-350.

Sen, Z. and A. Oztopal, 2001: Genetic algorithms for the classification and prediction of precipitation occurrence, *Hydrological Sciences*, **46**(2), pp. 255-268.

Shieh, H-J. and R.C. Peralta, 1997: Optimal system design of in-situ bioremediation using genetic annealing algorithm. In Ground Water: *An Endangered Resource*, Proceedings of Theme C, Water for a changing global community, 27th Annual Congress of the International Association of Hydrologic Research, pp 95-100.

Simpson, A.R., Dandy, G.C. and L.J. Murphy, 1994: Genetic algorithms compared to other techniques for pipe optimization, *J. Water Resour. Plng. And Mgmt.*, **120**(4), pp. 423-443.