A Multiscale Island Injection Genetic Algorithm for Optimal Groundwater Remediation Design

Meghna Babbar[1], Barbara Minsker[2], and David E. Goldberg

[1]Graduate Research Asst., Dept of Civil & Environmental Engineering, University of Illinois-Urbana, 205. N. Mathews Ave, Urbana, IL 61801; PH (217) 333-1657; mbabbar@uiuc.edu
[2]Asst. Professor, Dept of Civil & Environmental Engineering, University of Illinois-Urbana, 205. N. Mathews Ave, Urbana, IL 61801; PH (217) 333-9017; minsker@uiuc.edu

Abstract

Genetic algorithms have been shown to be powerful tools for solving a wide variety of water resources optimization problems. Applying these approaches to complex, large-scale applications, which is usually where these methods are most needed, can be difficult due to computational limitations. Large grid sizes are often needed for solving field-scale groundwater remediation design problems. Fine grids usually improve the accuracy of the solutions, but they also pose major bottlenecks in the computational efficiency of the algorithms. In this paper, we present multiscale parallel genetic algorithms that can be used to improve the performance of groundwater management problems that use multiscale grids. Application of the methods to a case study will be presented at the conference.

Introduction

Use of optimization techniques along with simulation models to identify cost effective designs for groundwater containment and remediation; has gained considerable popularity in the recent past. *Chan* (1994), *Culver and Shoemaker* (1997), *Karatzas and Pinder* (1996), *Lee and Kitanidis* (1991), *McKinney and Lin* (1996), *Minsker and Shoemaker* (1998), *Misirli and Yazicigil* (1997), *Ritzel et al* (1994), *Rizzo and Dougherty* (1996), *Sawyer et al* (1995), *Sun et al* (1996), *Tiedeman and Gorelick* (1993), and *Zhen and Uber* (1996) have implemented various optimization methods for groundwater management, including nonlinear and linear programming, optimal control, genetic algorithms, simulated annealing and cutting plane methods.

An important constituent of such coupled search processes is the numerical simulation model that analyzes the performance of candidate remediation designs. The degree of accuracy of numerical fate and transport models used in groundwater management models is a function of many factors, including grid size. Finer grids usually improve accuracy, but can increase computational efforts substantially. Some work has been done in the past to study the effect of multiscale grid approximations on optimization models [see *Albert and Goldberg*, 2001, *Liu and Minsker*, 2001]. *Albert et al* used a simple genetic algorithm (SGA) to explore the use of coarse grid approximations for solving a numerical integration problem. *Liu et al* (2001) devised a full multiscale approach for in-situ bioremediation design using a derivative-based optimization method called SALQR. *Miller and Goldberg* (1996) found that genetic algorithms with noisy fitness functions (i.e., uncertainty in the objective function or constraints) behave better when supplied with a coarse level of information rather than a fine level of information.

In this paper we explore the potential for two new multiscale parallel approaches to improve the performance of SGA based optimization models using a groundwater remediation design problem. The new approaches being investigated are:

a) Multiscale single population parallel genetic algorithm with a simple master-slave configuration.
b) Multiscale multi-population island injection genetic algorithm.

Details on these approaches are given below; a test of their performance on a case study will be presented at the conference.

Optimization Techniques

This section provides an overview of the methods used in this study.

Simple Genetic Algorithms

Simple Genetic Algorithms (SGA) have proven to be good optimization tools for various water resources problems [*Aly and Peralta, 1999, Cieniawski et al., Dandy and Engelhardt, 2001, 1995, Ritzel et al, 1994, Smalley et al, 2000, Wang, 1991, Wang and Zheng, 1997*]. Their main advantage over conventional gradient based methods lies in their ability to solve discrete, non-convex, and discontinuous problems [*Goldberg, 1989*]. SGAs emulate natural selection and genetics to produce optimal designs. They work with coded decision variables known as 'chromosomes', and search from a population of possible designs ("individual") using the information provided by the objective function ("fitness function"). Using three probabilistic operators - reproduction, crossover, and mutation the SGA evolves the population to find better solutions until it converges to the optimal or near-optimal solution.

GA theory shows that GAs find good solutions by assembling building blocks, which are high quality "pieces" of chromosomes. Goldberg, Deb & Clark (1992) have emphasized the importance of population sizing for a GA to have enough initial supply of building blocks. The population size can be estimated using the equation developed by *Harik et al* [1997] which is based on a random walk model for the initial supply of building blocks (BBs) and the gambler's ruin model for representing the interaction of competing BBs:

$$N \geq -2^{k-1} \ln(a) \frac{s_{bb}\sqrt{p\ (m-1)}}{d} \qquad (1)$$

where,
N = population size,
k = BB order,
a = probability of failure (less than 5%),
d = signal difference,
s_{bb} = standard deviation or noise interference between competing BBs,

In this equation, the term '$s_{bb}\sqrt{p\ (m-1)}$' is the standard deviation of fitness s_f for the entire population (*Reed et al*, 2001).

Other parameters, like crossover rate, mutation rate, convergence time, tournament size, etc., that are crucial to the design of an efficient SGA, were set using the methodology described by *Reed et al* (2000).

Master Slave Parallel Genetic Algorithms

Engineering problems are often computationally intensive, with each fitness function evaluation requiring a large amount of processor time. The simplest way to address this problem is a master slave parallelization, in which the 'master' processor distributes the fitness function evaluations among several 'slave' processors (*Cantu'-Paz et al*, 1999). The slave processors evaluate the fitness function and return the information back to the master processor who does all of the GA operations (see Figure 1). Master-slave GAs search the decision space in the same way as a SGA, but can be solved more quickly.

The two most important limiting components of the entire execution time are the computational time for each fitness evaluation and the communication time between the master and slave processors. Since communication between the master and each slave occurs every generation, the cost of communicating individuals creates a tradeoff between computation and communication costs. Therefore, master-slave GAs are more effective for problems that have computationally intensive fitness function evaluations. *Grefenstette* (1995) found that a complex robot learning task gave him 80% efficiency, vis-a-vis 50% efficiency for an easier task.

A control over the communication delays can be attained by using optimal number of processors. *Cantu'-Paz* [1999] estimated the optimal number of processors for a master-slave GA to be approximately:

$$P^* = \sqrt{\frac{n\,T_f}{T_c}} \tag{2}$$

where,
n = size of population,
T_f = Time required to evaluate fitness of each individual,
T_c = Time required to communicate between processors.

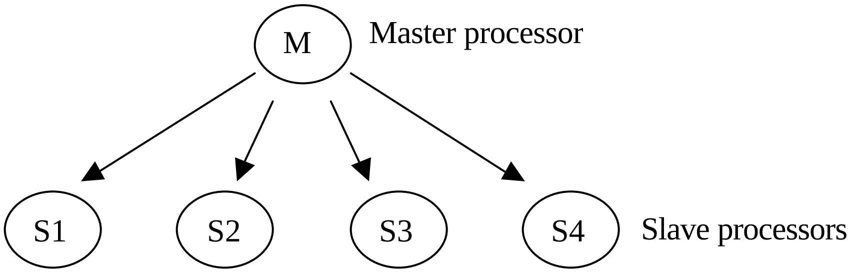

Figure 1. Master and Slave parallel processors

The master-slave SGA starts by searching the decision space with the ‘16 by 8’ (coarse) grid case. The fitness function evaluations with this grid will be noisy (i.e., inaccurate) because of the coarse grid approximation. The GA population also creates noise in evaluating building block fitness (σ_f) because a finite population is used. In this work we hypothesize that when the noise in fitness (σ_f) approaches the grid noise, no further improvements in the quality of the solution will be possible without errors and it is better to shift to a finer grid that can reduce grid noise by working on the more accurate version of the problem. In our experiments we test for various possible stages during the entire run when the shift to the finer grid is done, depending on the relation between the noise in fitness for the population and the constant grid noise (equation 3).

For the remediation problem, an estimation of grid noise is made by evaluating fitness of 1000 random individuals using both the ‘16 by 8’ and ‘48 by 24’ grid sizes. A relative grid noise for the ‘16 by 8’ grid size with respect to the ‘48 by 24’ grid size is calculated by estimating the variance of the error in fitness for the 1000 individuals. For the purposes of this study, the ’48 by 24’ grid is assumed to be error-free, but the same approach could be used to continue to finer grids.

$$Grid\ noise = s_n^2 = \sum_{i=1}^{N}\left[\frac{(E_{avg} - E_i)^2}{N-1}\right] \tag{3}$$

where,

N = population size (i.e. 1000),

Ei = error in fitness of each individual for ’16 by 8’ grid case with respect to the ’48 by 24’ grid case,

E_{avg} = average of error terms

Island Injection Genetic Algorithms

Island injection genetic algorithms are multi-population parallel GAs that are also called ‘Distributed’ GAs. Unlike a simple master slave parallel GA, they consist of groups of master-slave GAs (“islands”) that work with different populations and occasionally migrate individuals between each island. Each master-slave island searches its own space for feasible solutions. The islands can be interconnected through various topologies: Bidirectional rings, unidirectional rings, hypercubes, tree structure, etc. [*Cantu’-Paz* ,1999; *Malott et al*, 1996; *Eby et al*, 1999]. Different strategies for deciding migration rates and frequencies can be adopted. *Tanese et al* (1987), *Whitley et al* (1990), *Eby et al* (1999) and *Punch et al* (1995) have experimented with different fixed migration frequencies between different islands, where as *Cantu’-Paz* (1999) migrated individuals only after a certain convergence criterion for the entire deme was met.

Similar to the work done by *Malott et al* (1996) and *Eby et al* (1999), the topology of the iiGA used in this study consists of three islands that search for solutions within their own space for the same problem, but at different resolutions. Two islands work on the problem at lower resolution (in our case, a coarse grid) and they periodically inject solutions (for fine tuning) to the island working at higher resolution. The two ‘coarse’ islands also exchange solutions between each other at periodic intervals to push each other towards the global optima. Since the coarse islands work on a more ‘inexact’ representation of the numerical grid, they take much less time to compute the solutions. The good 'inexact' solutions passed into the 'finer' island push the search going on in this high resolution island towards a more accurate representation of the solution. See Figure 2 for the architecture.

*Cantu'-P*az (1999) found that as migration rates increased, the probability of the population converging sooner also increased. He also observed that above a 10% migration rate there was not much improvement in the convergence. In this study, three different migration rate schemes have been used for comparison purposes:

a) Migration rates between all islands are equal to 10%
b) high migration rate (10%) between low resolution islands and low migration rate (5%) from low islands to high resolution island.
c) Low migration rate (5%) between low resolution islands and high migration rate (10%) from low islands to high resolution island.

Migration between all islands takes place after every 10 generations. Best individuals migrate from one island to the other and replace the worst individuals there. Though transferring of best individuals should result in exchanging of good building blocks, excessive migration could lead to premature convergence. This possible effect will also be observed during the experiments.

The GA parameters used for all three islands are identical. They all work with the entire chromosome and search the entire decision space. The only difference between them lies in the resolution of the fitness function, population sizes and random number seeds. The number of optimal processors used for each island is calculated using equation (2).

Initially three experiments, using population sizes for all islands equal to 0.5, 1, and 1.5 times the population size calculated for the SGA, are performed. Migration rates of 10% (migration rate scheme (a)) between all islands are implemented for these experiments. Once the optimal population size for the islands are assessed, the remaining 2 migration rate schemes are applied to identify any change in performance of the GA.

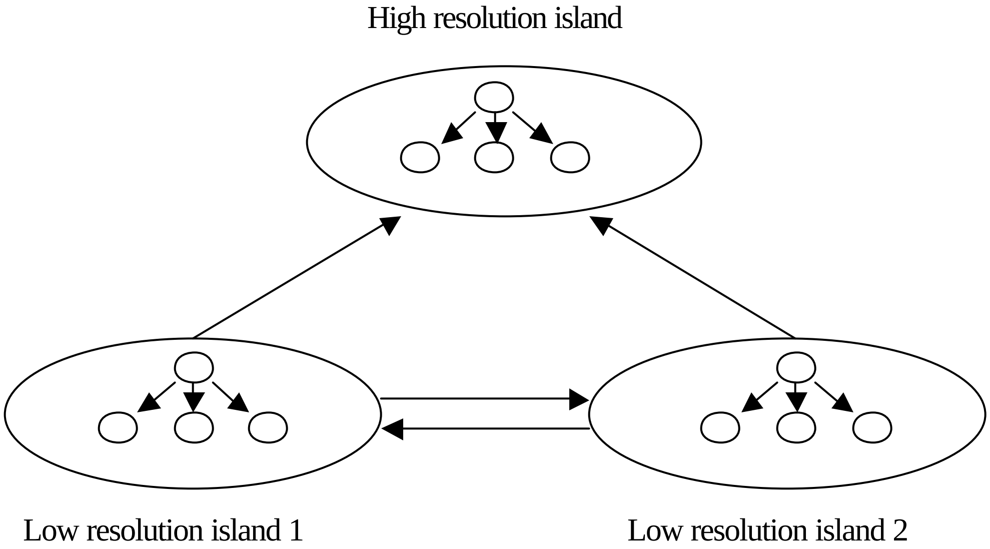

Figure 2. Island Injection topology

Experiments and Conclusions

Experiments for the above research are currently being performed. Complete results and analysis will be presented during the conference.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant No. BES 99-03889. This material is also based upon work supported by, or in part by, the U. S. Army Research Office under grant numbers DAAD19-00-1-0389 and DAAD19-001-1-0025.

References

Albert L.A., and Goldberg, D.E., Efficient Evaluation Genetic Algorithms under Fitness Functions, IlliGAL Report No. 2001024, July 2001.

Cantu'-Paz, E., A survey of Parallel Genetic Algorithms, *Calculateurs Paralleles, Reseaux et Systems Repartis,* Vol. 10, No. 2, pp. 141-171, Paris: Hermes, 1998.

Cantu'-Paz, E., Designing efficient and accurate parallel genetic algorithms, PhD thesis, 1999.

Chan, N., Partial infeasibility method of chance constrained aquifer management, *Journal of Water Resources Planning and Management*, 120(1), 70-89, 1994.

Clement, T. P. , RT3D - A modular computer code for simulating reactive multi-species transport in 3-Dimensional groundwater aquifers, *Battelle Pacific Northwest National Laboratory Research Report*, PNNL-SA-28967. (http://bioprocesses.pnl.gov/rt3d.htm.), 1997

Clement, T. P, Sun, Y., Hooker, B. S., and Petersen, J. N., Modeling multi-species reactive transport in groundwater, *Ground Water Monitoring and Remediation*, 18(2), 79-92, 1998.

Clement, T. P., Johnson, C. D., Sun, Y., Klecka, G. M., and Bartlett, C., Natural attenuation of chlorinated solvent compounds: Model development and field-scale application, *Journal of Contaminant Hydrology*, 42, 113-140, 2000.

Culver, T.B., and C.A. Shoemaker, Dynamic optimal ground-water reclamation with treatment capital costs, *Journal of Water Resources Planning and Management*, 123(1), 23-29, 1997.

Eby D., Averill R. C., Gelfand B., Punch W. F., Mathews O., Goodman E. D., An Injection Island GA for Flywheel Design Optimization, *Invited Paper, Proc. EUFIT '97, - 5th European Congress on Intelligent Techniques and Soft Computing***,** Sept., 97, 1997

Eby D., Averill R., Goodman E., and Punch W., The Optimization of Flywheels Using an Injection Island Genetic Algorithm, in Bentley, P. (ed.), *Evolutionary Design by Computers,* Morgan Kaufmann, San Francisco, 1999, pp.167-190.

Harik G. R., Cantu-Paz E., Goldberg D. E., and Miller B. L., The gambler's ruin problem, genetic algorithms and the sizing of populations, In *Proceedings of the 1997 IEEE Conference on Evolutionary Computation*, pp 7-12, IEEE press, New York, NY, 1997

Goldberg David E., *Genetic Algorithms in Search, Optimization and Machine Learning,* Addison–Wesley, New York, NY, 1989

Gopalakrishnan G., Minsker B., and Goldberg D.E., Optimal sampling in a Noisy Genetic Algorithm for Risk-Based Remediation Design, *Journal of Hydroinformatics*, in press, 2002.

Karatzas, G.P., and G.F. Pinder, The solution of groundwater quality management problems with a nonconvex feasible region using a cutting plane optimization technique, *Water Resources Research*, 32(4), 1091-1100, 1996.

Lee, S.I., and P.K. Kitanidis, Optimal estimation and scheduling in aquifer remediation with incomplete information, *Water Resources Research*, 27(9), 2203-2217, 1991.

Lin S-C., Goodman E.D., Punch W.F., Investigating Parallel Genetic Algorithms on Job Shop Scheduling Problems, *Evolutionar Programming VI, Proc. Sixth Internat. Conf., EP97, Springer Verlag, NY, P. J. Angeline, et al., eds.,Indianapolis*, pp.383-394, June, 97, 1997

Liu, Y., and B. S. Minsker, "Efficient multiscale methods for optimal in situ bioremediation design." *Journal of Water Resources and Planning Management*, in press, 2001.

Malott B., Averill R.C., Goodman E.D., Ding Y., Punch W.F., Use of Genetic Algorithms for optimal design of laminated composite sandwich panels with bending-twisting coupling, presented at *AIAA SDM (Structures, Dynamics and Materials)*, Apr 96, 1996

McDonald, M.G., and Harbaugh, A.W. (1988). "A modular three-dimensional finite-difference ground-water flow model." Techniques of Water Resources Investigations 06-A1, United States Geological Survey.

McKinney, D.C., and M.-D. Lin, Pump-and-treat ground-water remediation system optimization, *Journal of Water Resources Planning and Management*, 122(2), 128-136, 1996.

Miller B. L., and Goldberg D. E., Noise, Sampling and Efficient Genetic Algorithms, IlliGAL Report No. 97001, May 1997

Minsker, B.S., and C.A. Shoemaker, Dynamic optimal control of in situ bioremediation of groundwater, *Journal of Water Resources Planning and Management,* 124(3), 149-161, 1998.

Misirli, F., and H. Yazicigil, Optimal ground-water pollution plume containment with fixed charges, *Journal of Water Resources Planning and Management*, 123(1), 2-14, 1997.

Punch W.F., Averill R.C., Goodman E.D., Lin S-C., Ding Y., Design using Genetic Algorithms – Some results for laminated composite structures., IEEE expert, vol 10 (1), pg 42-49, February 95, 1995.

Reed P., Minsker B. S., and Goldberg D. E., Designing a competent simple genetic algorithm for search and optimization, *Water Resources Research*, 36(12), 3757-3761, 2000

Ritzel, B.J., J.W. Eheart, and S. Ranjithan, Using genetic algorithms to solve a multiple objective groundwater pollution containment problem, *Water Resources Research*, 30(5), 1589-1603, 1994.

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

Sawyer, C.S., D.P. Ahlfeld, and A.J. King, Groundwater remediation design using a three-dimensional simulation model and mixed integer programming, *Water Resources Research*, 31(5), 1373-1385, 1995.

Smalley J. B., Minsker B. S., and Goldberg D. E., Risk-based In Situ bioremeditation design using a noisy genetic algorithm, *Water Resources Research*, 36(20), 3043-3052, 2000.

Sun, Y.-H., M.W. Davert, and W.W.-G. Yeh, Soil vapor extraction system design by combinatorial optimization, *Water Resources Research*, 32(6), 1863-1873, 1996.

Whitley D., Starkweather T., Genitor II : A distributed genetic algorithm, *Journal of Experimental and Theoretical Artificial Intelligence*, 1990.

Tanese R., Parallel genetic algorithm for a hypercube, Grefenstette J.J., Ed., *Proceedings of the Second International Conference on Genetic algorithms*, 434-439. Morgan Kaufmann

Tiedeman, C., and S.M. Gorelick, Analysis of uncertainty in optimal groundwater containment capture design, *Water Resources Research*, 29(7), 2139-2153, 1993.

Zhen, C., and J.G. Uber, Reliability of remediation designs in presence of modeling error, *Journal of Water Resources Planning and Management*, 122(4), 253-261, 1996.