A Numerical Simulation of Groundwater Flow and Contaminant Transport on the Cray T3D and C90 Supercomputers

S. F. Ashby, W. J. Bosl, R. D. Falgout, S. G. Smith, A. F. B. Tompson, T. J. Williams

Lawrence Livermore National Laboratory
Livermore, California, USA

Abstract

Numerical simulations of groundwater flow and chemical transport through three-dimensional heterogeneous porous media are described. We employ two Cray supercomputers for different parts of the decoupled calculation: the flow velocity field is computed on the T3D massively parallel computer and the contaminant migration is simulated on the C90 vector supercomputer. We compare simulation results for subsurface models using homogeneous layers and heterogeneous layers, and we see that the heterogeneities have a profound impact on contaminant migration.

Introduction

Numerical simulations of fluid flow and chemical contaminant transport through porous subsurface formations play an increasingly important role in environmental remediation. Such simulations are used, for example, to analyze cleanup strategies for a particular site, as well as to manage a given procedure in the most cost-effective fashion, by determining the optimal pumping configuration in a pump-and-treat scenario, for instance. Simulations also can be used in risk assessment and in demonstrating regulatory compliance.

Realistic simulations must account for the heterogeneous nature of the three-dimensional subsurface medium. Although "homogenized" simulations can provide information about gross flow and transport behavior, they cannot reveal the complicated preferential flow paths that are present in heterogeneous media, and may not be able to represent fully the effects of nonlinear interactions that occur in localized regions. Thus, we advocate a strategy of detailed simulation as a complement to homogenized models and theories [7]. On a large site (several square kilometers and hundreds of meters deep), one must account for the effects of heterogeneities on the scale of 1-10m, and this requires the use of detailed spatial meshes (with 10^8 or more nodes).

In our direct numerical simulations, the heterogeneities are resolved in one or more statistical realizations of the subsurface formation. Hundreds of time-dependent simulations may be required as part of an exploratory investigation of various remediation options, or to evaluate the impact of different model parameters. Consequently, we need the capability to repeatedly run detailed simulations of large sites, and this need drives our use of supercomputers.

In this paper, we present a simple case study that illustrates these effects dramatically by comparing detailed simulations (more than one million spatial zones) for two subsurface models. In each case, the model is multilayered, consisting of seven distinct hydrostratigraphic units (which we call *geounits* for brevity): five layers, a clay region, and a fault zone. In the first simulation, we use homogeneous geounits; in the second simulation, we use heterogeneous geounits. The heterogeneities are represented by statistical random field models [7]. We simulate the migration of a dilute sorbing contaminant through a saturated medium (i.e., single phase). Here it is possible to decouple the flow and transport calculations. Flow velocities are computed via the solution of an elliptic PDE, and then passed to a particle-in-cell code, which uses them to simulate the contaminant transport. The flow computation is done on a Cray T3D massively parallel computer and the transport is done on a Cray C90 vector supercomputer.

Simulation Methodology

In this section, we describe our simulation strategy. Specifically, we discuss problem definition, groundwater flow modeling, and contaminant transport.

Subsurface model definition

The subsurface model is the key to problem specification. Geologists and hydrologists use hard data (gathered from monitoring wells and core samples) and soft data (via indirect measurements and inferred knowledge about a given site) to develop a conceptual model

Figure 1: *Subsurface model with homogeneous layers. The dark gray region represents a clay layer.*

Figure 2: *Same subsurface model as in Figure 1, but with heterogeneous layers.*

of the subsurface. This usually results in the identification of distinct geounits (hydrostratigraphic units), each having a specific soil type with certain properties, for example, hydraulic conductivity. Typically, there are clay layers, sand layers, and gravel lenses. This process is often aided by use of a front-end graphical user interface (such as GMS [14]), which allows the engineer or scientist to define the model interactively.

In many models, an effective mean value for the hydraulic conductivity may be assigned to the entire geounit [7], resulting in a homogeneous layer (Figure 1). In this paper, we use stochastic simulation to reproduce detailed variabilities (i.e., heterogeneities) within the layers (Figure 2). Specifically, we use Tompson's turning bands algorithm [9] to generate a spectral random field for the log-hydraulic conductivity ($\ln K$) distribution. This distribution often is modeled by [7, 8, 9, 12]

$$\ln K(\mathbf{x}) = F + f(\mathbf{x})$$

where F is a constant and $f(\mathbf{x})$ is a realization of a stationary, zero-mean, spatially correlated random field. The correlation function of f is anisotropic and assumed to be of the form

$$C(\mathbf{r}) = \sigma^2 \exp\left(-(r_x/\lambda_x)^2 - (r_y/\lambda_y)^2 - (r_z/\lambda_z)^2\right)^{1/2}$$

where $\mathbf{r} = (r_x, r_y, r_z)$ is a spatial separation lag vector. Input parameters to this model include the geometric mean K_G (L/T) for K ($K_G = \exp(F)$), the variance σ^2 of f, and correlation lengths (L) λ_x, λ_y, and λ_z. Each geounit may have its own set of geostatistics; the turning bands algorithm is run within each. In this way, we represent both the gross geologic features and the fine-scale heterogeneities. As we will see in the numerical simulations, these heterogeneities strongly influence contaminant migration. (See [4] for a discussion of the parallel implementation and performance of turning bands.)

Flow field computation

Our mathematical model of groundwater flow is derived from Darcy's Law and conservation of mass in a porous medium:

$$-\nabla \cdot (K \nabla h) - Q = 0 \qquad (1)$$

where h is the hydraulic head (L), K is the hydraulic conductivity of the medium (L/T), and Q is a source term (T^{-1}) used to represent pumping wells, for example. The hydraulic conductivity realization is obtained from a turning bands algorithm (applied to each layer) as described above. At present, the problem domain is assumed to be a parallelepiped; the boundary conditions may be Dirichlet, Neumann, or mixed. In the experiments below, we enforce no-flow (Neumann) conditions on the top and the bottom, and piece-wise linear hydraulic head (Dirichlet) conditions on the vertical faces.

We use a standard 7-point finite volume spatial discretization on a uniform mesh. After discretization, we obtain a large system of linear equations, $Ah = q$. The coefficient matrix A is symmetric positive definite and has the usual seven stripe pattern. The matrix has order $N = n_x \times n_y \times n_z$, where the n_i are the number of grid points in x, y, and z directions, respectively. For problems of interest, N is in the millions; the large number is dictated by the size of the physical site and the need to resolve heterogeneities adequately. Once the hydraulic head is computed, the velocity field can be calculated easily using a simple differencing scheme. This field is then passed to a transport code to simulate contaminant migration (see below).

The solution of the large linear system is computationally intensive and must be done efficiently and accurately. We are using a multigrid-preconditioned conjugate gradient (MGCG) solver. Although one could use multigrid as a stand-alone solver, we prefer to use it a preconditioner because it is easier to implement in this fashion. The outer CG iteration provides robustness, especially when the underlying PDE has a nearly discontinuous coefficient function, as in our case. In the current implementation of MGCG, the preconditioning step consists of a single standard *V-cycle* with weighted Jacobi as the smoother. Since our problems tend to be anisotropic (due to a skewed grid aspect ratio), we employ a *semi-coarsening* strategy in which the grid is coarsened in one spatial direction at a time. Because of the "discontinuous" nature of K, we use *operator-induced* prolongation and restriction (c.f. [1, 5, 6]). See [2] for a more complete description of this algorithm and its performance on the T3D for a variety of test problems.

To solve this problem on the T3D, we first distribute the problem data across a virtual 3D process grid consisting of $P = p \times q \times r$ processes. The grid points within a process are arranged as a 3D subgrid, and the code uses a nested loop to access these points. The computations are organized so as to avoid explicit data redistribution, thereby improving the code's efficiency. Although each process has a piece of the problem domain, we are *not* doing domain decomposition in the algorithmic sense. We are solving the *full* problem rather than independent subproblems. We have overlapped much of the code's communication with computations, thereby enhancing scalability on machines that permit this. At present, we have layered our own message-passing library, AMPS, on top of PVM, Chameleon, and the Reactive Kernel, which allows us to run our code on workstation clusters, the nCUBE/2, the IBM SP-1, and of course, the Cray T3D. This layer eventually will be replaced with the emerging MPI standard. See [4] for a discussion of the parallel implementation and performance of this code on the T3D.

Contaminant transport simulation

We now consider the transport of one or more dilute, reactive chemical contaminants within a steady-state groundwater flow field (as calculated by ParFlow, for example). Specifically, we consider neutrally buoyant chemical species, or constituents, distributed between the aqueous and solid phases of the porous medium. Solute mass transfer between the liquid and solid phases is assumed to be governed by reversible, equilibrium, sorption reactions. This means that the chemical mass is partitioned between the liquid and solid phases "instantaneously," or at least much faster than the time scale of transport (advective flow). As mentioned earlier, we may decouple the flow and transport stages of simulation because the problem is single phase and we are interested only in dilute contaminants.

The total mass balance equation for constituent i is

$$\frac{\partial \rho_i}{\partial t} + \nabla \cdot \left(\frac{\mathbf{v} \rho_i}{\mathcal{R}_i} \right) - \nabla \cdot \left(\varepsilon \mathsf{D}_i \cdot \nabla (\frac{\rho_i}{\varepsilon \mathcal{R}_i}) \right) = \frac{-\rho_i}{\varepsilon \mathcal{R}_i} \, Q \quad (2)$$

where $\rho_i(\mathbf{x}, t)$ is the total (aqueous plus solid) mass density of constituent i at a point in the subsurface formation (m/L^3), $\mathbf{v}(\mathbf{x})$ is the average pore fluid velocity (computed by ParFlow), $\mathsf{D}_i(\mathbf{v})$ is a hydrodynamic dispersion tensor (L^2/T), ε is the medium porosity $(\cdot)$, and $\mathcal{R}_i(\varepsilon, t)$ is a dimensionless retardation factor that describes how the total concentration (ρ_i) is related to the aqueous concentration (c_i) and local medium reactivity:

$$\rho_i = \varepsilon c_i \mathcal{R}_i. \quad (3)$$

The right hand side in (2) accounts for loss of chemical mass from a pumping sink of strength Q. Equations (2) and (3) are the fundamental transport equations we wish to solve (for ρ_i and c_i).

We use an efficient particle-grid method [8, 11, 12, 13] to solve these equations. The total concentration distribution for constituent i, ρ_i, is represented by a collection of N_i particles (each of mass m). The concentration at a given point in the subsurface is determined from the mass density of particles in a neighborhood about that point. The number of particles per unit mass specifies the "particle resolution." The greater this resolution, the more accurate the simulation will be, and the greater the computational expense. A random walk procedure is used to move and diffuse the particles through the subsurface. Particles may be removed from the system as a result of pumping within the domain, or if they cross a domain boundary. See [8, 10].

In the SLIM computer code [12], the continuum variables are represented as cell-centered quantities on the grid used in ParFlow (which computes the hydraulic head at node centers). During particle displacement, these quantities are gathered to particle positions using an interpolation formula [11]. Following particle displacement, the particle masses are scattered back to the grid cells using another interpolation formula, leading to a distribution of $\rho_i(\mathbf{x}, t)$ on the grid. Some particles may be removed to account for boundary conditions and mass sinks.

An initial "point source" is represented by assigning an appropriate number of particles to the desired cells.

Geounit	K_G	σ	λ_x	λ_y	λ_z
Top Layer	6.7	0.2	300.0	300.0	20.0
2nd Layer	5.3	0.6	250.0	250.0	25.0
3rd Layer	4.0	1.8	200.0	200.0	20.0
4th Layer	2.7	0.5	400.0	400.0	30.0
5th Layer	1.3	1.0	200.0	200.0	20.0
Clay Layer	0.004	0.0	—	—	—
Fault Zone	0.4	0.0	—	—	—

Table 1: *Geostatistical parameters used in each of the heterogeneous layers shown in Figure 2. (K_G units are ft/day and λ units are ft.)*

Boundary conditions at the edges of the grid domain may allow for outflow, reflection (no flux), or Dirichlet (constant concentration) conditions. Isolated pumping wells located within single grid cells remove particles at random as a function of the time step and pumping rate.

Simulation Results

In this section, we describe the results of two numerical simulations, the purpose of which is to study the influence of fine-scale heterogeneities on contaminant migration through a multilayered subsurface (as described above). Our test problem corresponds to a physical domain of size $12700 \times 12700 \times 630$ ft^3 with $N = 129 \times 129 \times 65$ grid points (over one million spatial zones). This implies an approximate grid spacing of $\Delta_x = 99.2$ ft, $\Delta_y = 99.2$ ft, and $\Delta_z = 9.8$ ft, yielding an approximate grid aspect ratio of 10:10:1. The geostatistical parameters used in each layer in the heterogeneous model (Figure 2) are given in Table 1. In the homogeneous model (Figure 1), the geometric means, K_G, are used throughout the entire layers (with $\sigma = 0$). We impose no-flow conditions on the top and bottom of the domain, and piece-wise linear Dirichlet boundary conditions on the four vertical sides (hydraulic head values ranged from 510 to 670 ft, depending on the side). The MGCG algorithm was halted once the C-norm of the relative residual was less than 10^{-9} (where C denotes the multigrid preconditioning operator).

We study the impact of a single pumping well (located in the center of the domain) on the migration of a sorbing chemical species for each of the homogeneous and heterogeneous models. (The contaminant reacts with the solid phase according to a linear sorption law.) The pumping action was simulated by fixing the hydraulic head at the appropriate grid cell to be 530 ft; capture of particles in the vicinity of the well was simulated via a probabilistic function. The simulation lasted 30,000 times steps ($\Delta t = 5$ days), and

Subsurface Model	Well	MGCG Iters	MGCG Time	Total Time
Homogeneous	no	19	58.0	167.7
Homogeneous	yes	19	58.0	167.3
Heterogeneous	no	57	163.7	403.4
Heterogeneous	yes	55	153.9	393.0

Table 2: *MGCG iterations and wall-clock time (in seconds) required to solve for the hydraulic head (on a $129 \times 129 \times 65$ spatial mesh).*

the well was turned on at step 4,500. To do this, we computed two flow fields: one without the well and one with the well. We assume that the transient effects of starting the well all occur within the span of a single time step. These flow fields were passed to SLIM, which made the switch at the prescribed time step. This was done for each of the subsurface models. We remark that this investigation is similar to the one described in [10], but there we employed only a two-layer subsurface medium.

Flow velocities

The MGCG iteration counts and timings (in seconds) required for each of the hydraulic head computations are summarized in Table 2. We give the MGCG time for solving the large, linear system, as well as the total time spent in the simulator. This latter time includes problem setup, which is expensive in the case of the heterogeneous model because the turning bands algorithm must be called for each layer. The process grid topology was $P = 1 \times 4 \times 4$. Notice that the heterogeneous problems take longer, which is due to ill-conditioning induced by the variability in K. The well has a negligible effect on the flow calculation, but of course, affects the resulting velocity field. The resulting velocity fields (with the pumping well) are shown in Figures 3 and 4.

Parallel performance on T3D

In Figure 5, we report scaled speedup for the matrix-vector multiplication, multigrid preconditioning, and MGCG routines for a 128-node Cray T3D (each node consists of a 150MHz DEC Alpha processor and 64MB of memory). Our message-passing library is layered on top of PVM; we expect much better results when we replace PVM with Cray's SHMEM library. For a more complete discussion of the parallel performance of these and other ParFlow routines, see [4]. Scaled speedup is defined to be T_1/T_P, where T_P is the time to solution on P processes where the total problem size is allowed to grow with P. In our experiments, each processor was given $2^{17} \approx 10^5$ grid points. Note that both the MG preconditioner and the MGCG algorithm appear to be fairly scalable, meaning that the time per iteration remains constant as we increase the problem size *and* number of processes in tandem. To calculate these graphs, it was necessary to vary the $p : q : r$ and $n_x : n_y : n_z$ ratios. In these experiments, we chose $p = 1$, which gives the best overall time to solution. As discussed in [4], the choice of n_x and p has a dramatic impact on node performance, which (at least in part) explains the erratic performance graphs. MFLOP rates were observed to vary by more than an order of magnitude depending on the number of grid points in the innermost loop of the matvec routine [4]. This is due to several factors, including cache issues, the inherent inefficiency of PVM, and a poor C compiler.

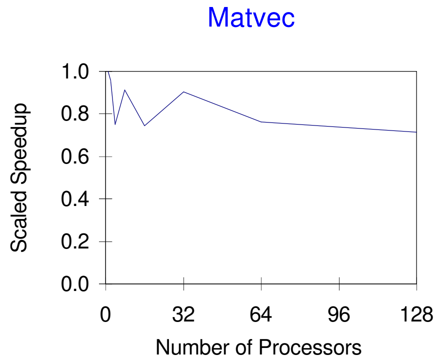
Figure 3: *Velocity magnitude for homogeneous model.*

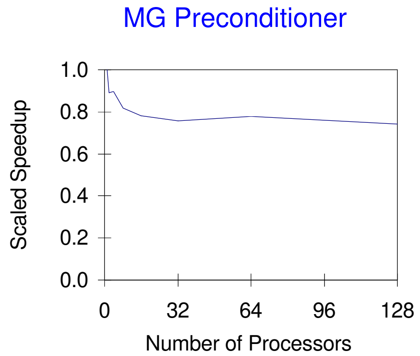
Figure 4: *Velocity magnitude for heterogeneous model.*

Chemical transport

After computing the flow fields, they were passed to SLIM for the contaminant migration study. Recall that in each simulation (homogeneous layers versus heterogeneous layers), we used two flow fields: one without the well and one with the well; the fields were switched at time step 4500. This simulates the effect of an undetected contaminant plume spreading for a period of time before pump-and-treat remediation is activated. The node-centered hydraulic heads computed by ParFlow were transformed into cell-centered velocity components using the same $129 \times 129 \times 65$ discrete spatial domain. The initial constant concentration was established by assigning particles to sixteen contiguous grid cells. We used approximately 140 particles per cell, with a total of 2240 particles in the system. This is a very coarse representation for illustrative purposes only; normally, many more particles (on the order of 10^5) would be used to achieve a reasonable solution.

The two simulations are summarized in Figures 6 and 7. In each figure, we show extent of the contaminant plume at four different times in the simulations. The first frame in each figure was taken at time step 1000 (before the well was turned on), the second frame at step 4500 (when the well started to pump), the third frame at step 15,000, and the fourth frame at step 30,000 (end of the simulation). Figure 6 illustrates the migration within the homogeneous model; Figure 7 is for the heterogeneous model. In the former, the pumping well appears to have cleaned up the contaminant fairly well, but in the latter, most of the contaminant is downstream of the well and still in the subsurface. The heterogeneities have resulted in preferential flow channels, which have caused the contaminant dispersion to be much more rapid than predicted using the simpler homogeneous model. Also, we remark that the layered subsurface resulted in interesting flow behavior. In both models, there was a lagging plume in the lower subsurface (where the hydraulic conductivities were smaller), and a leading plume in the upper portion of the subsurface (where the hydraulic conductivities are larger). In the heterogeneous model, the lagging plume was effectively cleaned up, but the leading portion escaped the well.





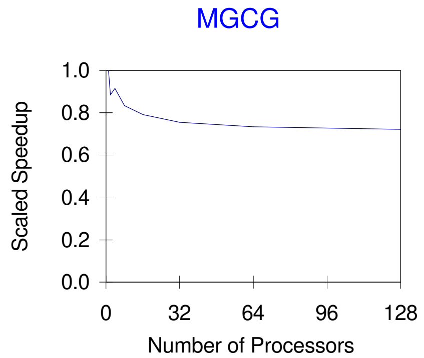


Figure 5: *Scaled speedup on the Cray T3D.*

Performance on the C90

We ran the SLIM simulations on a single processor of a Cray C90. Each simulation took about 160 seconds of CPU time, ran at an average of 144 MFLOPS, and showed an average vector length of 75 (128 is ideal). Particle-grid simulations are not generally known to achieve high MFLOP rates; they are bound by memory-access speed. These SLIM runs, for example, executed about 125 million vector memory references per second on average. The short vector length is caused in part by the small size of this problem (only 2240 particles, starting out in only 16 grid cells). We expect much better node performance when we run more detailed simulations (with 50,000 or more particles per constituent).

Profiling measurements on SLIM reveal that about 36% of the time went toward computing the total mass density values, which includes the accumulation (scatter–add) from the particles. Determining the liquid concentrations from the total mass densities accounted for 18% of the time. Moving the particles, including computation of the force values, took 16% of the time. Particle boundary condition enforcement and diagnostic breakthrough counting (not discussed here) took another 14%. Modeling the pumping well, a sort of specialized boundary condition, took 8%; the remaining time was spent in miscellaneous routines.

Acknowledgements

We acknowledge the valuable participation of Charles Baldwin, John Bell, Thomas Fogwell, Gwen Loosmore, and John Ziagos in the ParFlow project. In particular, we thank Dr. Thomas Fogwell for his helpful comments and guidance concerning multigrid.

The Cray T3D experiments described in this paper were run on the 128-node machine located in the Center for Computational Sciences and Engineering at Lawrence Livermore National Laboratory (LLNL) as part of the H4P Industrial Computing Initiative. Early code development was done on the Cray T3D located at the Pittsburgh Supercomputer Center. The Cray C90 experiments were run on the machine located at the National Energy Research Supercomputer Center at LLNL.

Figure 6: *Snap-shots in time: dispersion of contaminant plume assuming homogeneous layers.*

Figure 7: *Snap-shots in time: dispersion of contaminant plume assuming heterogeneous layers.*

This work was performed at LLNL under contract W-7405-ENG-48 for the U.S. Department of Energy (DOE). It was supported in part by the DOE Defense Programs Technology Transfer Initiative, the Laboratory Directed Research and Development program at LLNL, the Applied Mathematical Sciences program of the DOE Office of Scientific Computing, and by the Strategic Environmental Research and Development Program of the U.S. Department of Defense through the cooperation of the USAE Waterways Experiment Station (Vicksburg, MS).

References

[1] R. E. Alcouffe, A. Brandt, J. E. Dendy, and J. W. Painter, *The multi–grid method for the diffusion equation with strongly discontinuous coefficients*, SIAM J. Sci. Stat. Comput., 2 (1981), pp. 430–454.

[2] S. F. Ashby, R. D. Falgout, S. G. Smith, and T. W. Fogwell, *Multigrid preconditioned conjugate gradients for the numerical simulation of groundwater flow on the CRAY T3D*, Tech. Report UCRL-JC-118622, Lawrence Livermore National Laboratory, 1994. Submitted to the proceedings of the ANS International Conference on Mathematics and Computations, Reactor Physics, and Environmental Analyses, April 30–May 4, 1995, Portland, OR.

[3] S. F. Ashby, R. D. Falgout, S. G. Smith, and A. F. B. Tompson, *Modeling groundwater flow on MPPs*, in Proc. 1993 Scalable Parallel Libraries Conference, A. Skjellum and D. Resse, eds., IEEE Computer Society Press, 1994, pp. 17–25. Held at Mississippi State University, October 6–8, 1993. Also available as LLNL technical report UCRL-JC-115602.

[4] ——, *The parallel performance of a groundwater flow code on the CRAY T3D*, in Proc. Seventh SIAM Conference on Parallel Processing for Scientific Computing, Society for Industrial and Applied Mathematics, 1995, pp. 131–136. Held in San Francisco, February 15–17, 1995. Also available as LLNL technical report UCRL-JC-118604.

[5] J. E. Dendy, *Black box multigrid*, J. Comput. Phys., 48 (1982), pp. 366–386.

[6] T. W. Fogwell and F. Brankhagen, *Multigrid method for the solution of porous media multiphase flow equations*, in Nonlinear Hyperbolic Equations – Theory, Computation Methods, and Applications, vol. 24 of Notes on Numer. Fluid Mech., Vieweg, Braunschweig, 1989, pp. 139–148.

[7] L. W. Gelhar, *Stochastic Subsurface Hydrology*, Prentice Hall, 1993.

[8] A. F. B. Tompson, *Numerical simulation of chemical migration in physically and chemically heterogeneous porous media*, Water Resources Res., 29 (1993), pp. 3709–3726.

[9] A. F. B. Tompson, R. Ababou, and L. W. Gelhar, *Implementation of of the three-dimensional turning bands random field generator*, Water Resources Res., 25 (1989), pp. 2227–2243.

[10] A. F. B. Tompson, S. F. Ashby, R. D. Falgout, S. G. Smith, T. W. Fogwell, and G. A. Loosmore, *Use of high performance computing to examine the effectiveness of aquifer remediation*, in Proc. X International Conference on Computational Methods in Water Resources, A. Peters, G. Wittum, B. Herrling, U. Meissner, C. Brebbia, W. Gray, and G. Pinder, eds., vol. 2, Kluwer Academic Publishers, Dordrecht, 1994. Held in Heidelberg, Germany, July 19-22, 1994. Also available as LLNL technical report UCRL-JC-115374.

[11] A. F. B. Tompson and D. E. Dougherty, *Particle-grid methods for reacting flows in porous media with application to Fisher's Equation*, Applied Mathematical Modelling, 16 (1992), pp. 374–383.

[12] A. F. B. Tompson, E. Vomvoris, and L. W. Gelhar, *Numerical simulation of solute transport in randomly heterogeneous porous media: motivation, model development, and application*, Tech. Report 316, R. M. Parsons Laboratory, Dept. of Civil Engineering, Massachusetts Institute of Technology, Cambridge, MA, 1988. Also available as LLNL technical report UCID-21287 (1987).

[13] G. Uffink, *A random walk model for the simulation of macrodispersion in a stratified aquifer*, in Proc. of the IAHS Symposia, vol. HS2, 1983. IUGG 18th General Assembly, Hamburg.

[14] United States Department of Defense, *Groundwater Modeling System Reference Manual*, Engineering Computer Graphics Laboratory, Brigham Young University, Provo, UT, June 1994. Preliminary Draft.