

**PARALLELIZABLE METHODS FOR
MODELING FLOW AND TRANSPORT IN
HETEROGENEOUS POROUS MEDIA**

Myron B. Allen

Mark C. Curran

Proceedings

1992

WWRC-92-11

In

**Proceedings of the Oberwolfach Conference
on Porous Media**

Submitted by

**Myron B. Allen
Department of Mathematics
University of Wyoming
Laramie, Wyoming**

**Mark C. Curran
Applied and Numerical Mathematics Division
Sandia National Laboratories
Albuquerque, New Mexico**

Parallelizable Methods for Modeling Flow and Transport in Heterogeneous Porous Media

Myron B. Allen and Mark C. Curran

Groundwater contaminant modeling presents several challenges to the mathematician. Among these are the need to compute accurate water velocities and difficulties arising from fine-scale heterogeneities and sharp concentration fronts. This paper presents parallelizable numerical methods that address these challenges.

For groundwater flow, mixed finite-element models yield velocities comparable in accuracy to computed heads. However, large variations in hydraulic conductivity can cause iterative matrix solvers to converge slowly. The fine grids needed to resolve heterogeneities aggravate the poor conditioning. A parallelizable, multigrid-based iterative scheme for the lowest-order mixed method largely overcomes both sources of poor behavior.

For contaminant transport, finite-element collocation yields high-order spatial accuracy. The timestepping scheme combines a modified method of characteristics, which reduces temporal errors when advection dominates, with an alternating-direction formulation, which is "embarrassingly parallel" and has a favorable operation count.

1. Introduction

The equations governing steady flow of water in a two-dimensional, rectangular porous medium Ω have the following forms:

$$\begin{aligned} \mathbf{u} &= -K \nabla p \quad \text{in } \Omega, \\ \nabla \cdot \mathbf{u} &= f \quad \text{in } \Omega. \end{aligned} \tag{1}$$

Here $\mathbf{u} = (u^x, u^y)$, p , and f represent the Darcy velocity, hydraulic head, and source term, respectively. In natural aquifers, the hydraulic conductivity $K(x, y)$ varies in space depending upon the lithology of the host rock. We assume that K is bounded above and that $\inf K(x, y) > 0$.

The spatial variability, or heterogeneity, in K causes difficulties for mathematical modelers. In particular, two sources of poor conditioning often affect the linear systems that approximate the governing equations. One source is the need to use fine spatial grids to resolve the variations in the medium and the resulting variations in p and \mathbf{u} . The other is the variability in K itself, which affects the matrix entries of the linear system.

In this context, mixed finite-element methods have attracted much attention. These methods, together with appropriate choices of trial spaces, yield solutions for p and

\mathbf{u} that have the same order of accuracy as the grid mesh size $h \rightarrow 0$ (Douglas et al., 1983; Raviart and Thomas, 1977). Standard Galerkin and finite-difference formulations generally do not enjoy this property, since they require one to solve for p and then numerically differentiate to compute \mathbf{u} . Since velocities determine the main features of the contaminant transport, mixed methods are therefore better suited to the coupled flow-and-transport problem.

Contaminant transport poses another set of difficulties. Here, the governing equation takes the form

$$\partial_t c + \mathbf{u} \cdot \nabla c - \nabla \cdot (D_H \nabla c) = 0 \quad \text{in } \Omega, \quad (2)$$

where $c(\mathbf{x}, t)$ is the contaminant concentration and D_H represents the hydrodynamic dispersion tensor. This equation is formally parabolic.

In many applications, advection dominates, with the dissipative effects of hydrodynamic dispersion having only a small influence. In such regimes, Equation (2) exhibits hyperbolic behavior, and sharp fronts in contaminant concentration tend to persist. Low-order numerical methods, such as upstream-weighted finite-differences, smear these fronts. Even high-order methods typically fail to capture the fronts accurately unless one uses either globally or locally fine spatial grids. In two or three space dimensions, the computational effort associated with such grids can be onerous, especially on serial-architecture machines.

Finite-element collocation on cubic trial spaces offers high-order spatial accuracy, but, like other techniques, it yields unwieldy matrix equations in the multidimensional problems arising in practice. An alternating-direction algorithm similar to that proposed by Celia (1983) decomposes these unwieldy equations into parallelizable sets of smaller linear systems that can be solved with significantly fewer arithmetic operations. Moreover, the scheme is amenable to timestepping along approximate characteristic curves, a tactic that reduces the temporal truncation error (Russell, 1980).

This paper examines these numerical methods. For the flow equations (1), we consider an iterative scheme for solving the lowest-order mixed finite-element approximations on rectangular grids. The overall structure of the scheme, analyzed in detail by Allen et al. (1992), consists of an outer iteration, whose convergence rate is independent of h and of spatial variations in K , coupled with an inner iteration on an elliptic linear system. We use a highly parallelizable multigrid method to ensure that the inner iterations are rapid. For the transport equation (2), we examine an alternating-direction collocation (ADC) scheme that employs a modified method of characteristics and exhibits excellent parallelism (Allen and Khosravani, 1992).

2. The Mixed Finite-Element Method for Flow Equations

Consider Equations (1), subject to the boundary condition $p = 0$ on $\partial\Omega$. To discretize this system via the lowest-order mixed method, we construct a rectangular grid Δ on Ω having vertical grid lines at $x = x_0, x_1, \dots, x_m$ and horizontal grid lines at $y = y_0, y_1, \dots, y_N$. The mesh size of Δ is $h := \max\{x_i - x_{i-1}, y_j - y_{j-1}\}$. With Δ we associate trial spaces Q_x , Q_y , and V for the x -velocity u^x , the y -velocity u^y , and the hydraulic head p , respectively. The space Q_x contains functions that are piecewise linear in x and piecewise constant in y ; Q_y contains functions that are piecewise constant in x and piecewise linear in y , and V contains functions that are piecewise constant on

Δ . Crucial to the error estimates associated with these spaces is the fact that, if $\mathbf{v} \in Q_x \times Q_y$, then $\nabla \cdot \mathbf{v} \in V$ (Raviart and Thomas, 1977).

Each of these trial spaces has a tensor-product basis containing products of the usual one-dimensional basis functions for piecewise constant and piecewise linear interpolation. We associate a nodal value $p_{i,j}$ of head with the centroid of each cell $[x_{i-1}, x_i] \times [y_{j-1}, y_j]$ formed by the grid Δ , a nodal value $u_{i,j}^x$ of x -velocity with the midpoint $(x_i, y_{j-1/2})$ of each vertical cell edge, and a nodal value $u_{i,j}^y$ with the midpoint $(x_{i-1/2}, y_j)$ of each horizontal cell edge.

Given these trial spaces, the mixed formulation for Equations (1) is as follows: Find $\mathbf{u}_h \in Q_x \times Q_y$ and $p_h \in V$ such that

$$\begin{aligned} \int_{\Omega} \frac{\mathbf{u}_h \cdot \mathbf{v}}{K} dx dy - \int_{\Omega} p_h \nabla \cdot \mathbf{v} dx dy &= 0, \quad \forall \mathbf{v} \in Q_x \times Q_y, \\ \int_{\Omega} (\nabla \cdot \mathbf{u}_h - f) q dx dy &= 0, \quad \forall q \in V. \end{aligned} \quad (3)$$

This finite-dimensional system yields approximations \mathbf{u}_h and p_h whose global errors are both $O(h)$ in the norm $\|\cdot\|_{L^2(\Omega)}$ (Raviart and Thomas, 1977).

Under lexicographic ordering of equations and unknowns, Equations (2) yield a linear system having the following block structure:

$$\begin{bmatrix} A & N \\ N^T & 0 \end{bmatrix} \begin{bmatrix} U \\ P \end{bmatrix} = \begin{bmatrix} 0 \\ F \end{bmatrix}. \quad (4)$$

The vector U contains nodal values of the velocities u^x and u^y , and P contains nodal heads. The matrix A is symmetric and positive definite and has the block structure

$$A = \begin{bmatrix} A^x & 0 \\ 0 & A^y \end{bmatrix}.$$

The blocks A^x and A^y are tridiagonal, their entries being integrals of the form

$$\int_{\Omega} K^{-1} \varphi_k \varphi_\ell dx dy,$$

where φ_k, φ_ℓ are functions belonging to the basis for $Q_x \times Q_y$. In practice, we approximate these integrals using a two-point Gauss composite rule in each coordinate direction.

The matrix N has the block structure

$$N = \begin{bmatrix} N^x \\ N^y \end{bmatrix},$$

where N^x and N^y . These blocks mimic the usual difference approximations to $\partial/\partial x$ and $\partial/\partial y$. The vector F contains integrals involving the source function f . For a detailed specification of the entries in this linear system, we refer readers to Allen et al. (1992).

3. An Iterative Scheme for the Mixed Method

We solve the system (4) iteratively, using the following matrix splitting:

$$\begin{bmatrix} D & N \\ N^T & 0 \end{bmatrix} \begin{bmatrix} U \\ P \end{bmatrix}^{(k+1)} = \begin{bmatrix} 0 \\ F \end{bmatrix} + \begin{bmatrix} D - A & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} U \\ P \end{bmatrix}^{(k)}. \quad (5)$$

Here, D is a diagonal matrix, the simplest effective structure for which is the $\text{diag}(A)$. This scheme has convergence rate that is independent of mesh size and of variations in K . In fact, each iteration reduces the error by a factor no greater than $\frac{1}{2}$ (see Allen et al., 1992).

Computationally, the scheme (5) requires the following steps:

- (i) $G^{(k-1)} \leftarrow -F + N^T D^{-1}(D - A)U^{(k-1)}$.
- (ii) Solve $N^T D^{-1}NP^{(k)} = G^{(k-1)}$.
- (iii) $U^{(k)} \leftarrow D^{-1}(D - A)U^{(k-1)} - D^{-1}NP^{(k)}$.

Steps (i) and (iii) in this algorithm are cheap. Step (ii), however, requires more work, since $N^T D^{-1}N$ has the same pentadiagonal structure as the usual five-point finite-difference approximation to operators of the form $\nabla \cdot K \nabla$.

Instead of executing step (ii) exactly, we use a multigrid scheme to solve the pentadiagonal system approximately. Thus the matrix splitting serves as an “outer” iteration, while the multigrid cycles executed for step (ii) constitute an “inner” iteration. In particular, we perform several V-cycles to get an approximate value for $P^{(k)}$, then proceed to step (iii). Each V-cycle involves two Gauss-Seidel iterations at each level in a nest $\Delta = \Delta_0 \supset \Delta_1 \supset \dots \supset \Delta_L$ of successively coarser grids, the mesh size of Δ_k being $2^k h$. For the intergrid transfers, we use full weighting as a restriction operator and bilinear interpolation as a prolongation operator.

One attractive feature of the multigrid scheme is its amenability to parallel processing. Tuminaro and Womble (to appear), for example, discuss this advantage. By adopting a red-black ordering for the cells in each grid, we decompose each Gauss-Seidel relaxation sweep into two sets of calculations. In particular, we designate each cell $[x_{i-1}, x_i] \times [y_{j-1}, y_j]$ in a grid as *red* or *black*, depending on whether $i + j$ is even or odd. We update each of the red cells using old values in the black cells, then use the new red values to update the black cells. In any sweep, calculations for red cells are independent of each other. Updates for black cells are also mutually independent.

To implement the scheme on a distributed-memory machine, we arrange for each processor to manage a 32×32 -cell rectangular region, or *patch*, of the original fine grid. The relaxation sweep on any patch requires some values of latest iterates from the nearest-neighbor patches. Therefore, before executing a relaxation sweep, a processor must trade information about a “boundary layer” of nodal values with the processor that manages the nearest-neighbor patch. Therefore, the parallel implementation requires communication between processors before each “red” sweep and before each “black” sweep. This communication prevents ideal parallel speedups.

4. Computational Performance of the Mixed-Method Scheme

Allen et al. (1992) discuss the performance of the serial scheme in the presence of the following heterogeneous conductivity fields $K(x, y)$ on $\Omega = (0, 1) \times (0, 1)$:

$$K_I(x, y) = 1;$$

$$K_{II}(x, y) = e^{-x-y};$$

$$K_{III}(x, y) = \begin{cases} 1, & \text{if } x < y, \\ 0.1, & \text{if } x \geq y; \end{cases}$$

$$K_{IV}(x, y) = K_{II}(x, y) \cdot K_{III}(x, y);$$

$$K_V(x, y) = \begin{cases} 1, & \text{if } x < y, \\ 0.01, & \text{if } x \geq y. \end{cases}$$

The experiments involve grids with $h = 2^{-\ell}$, where $\ell = 4, 5, 6, 7, 8$. Each iteration of the solution scheme includes two V-cycles of the multigrid algorithm, where the coarsest grid in each cycle has mesh 2^{-1} , and the finest has mesh $2^{-\ell}$. Table 1 displays the convergence rates of the outer iteration versus coefficient and mesh size. The results confirm the theoretical bound of $\frac{1}{2}$ for the convergence rate.

Table 1: Convergence rates for the outer iteration of the flow-equation scheme using various coefficients and grids.

Coefficient	Grid mesh h				
	2^{-4}	2^{-5}	2^{-6}	2^{-7}	2^{-8}
K_I	0.4933	0.4988	0.4993	0.4995	0.4999
K_{II}	0.4966	0.4995	0.4988	0.4997	0.4999
K_{III}	0.4948	0.4982	0.4991	0.4998	0.4999
K_{IV}	0.4947	0.4980	0.4992	0.4998	0.4999
K_V	0.4939	0.4978	0.4989	0.4999	0.5000

To assess the scheme's parallelism, we examine its execution time on a 1024-processor nCube 2 having a hypercube architecture. To measure speedups, we examine execution times required on subcubes of the machine having dimension 0 (1 processor), 1 (2 processors), ..., 10 (1024 processors), running problems of proportionately larger size on larger subcubes. Each subcube is a set of processors linked by the shortest possible physical paths in the machine. Hence proper subcubes suffer essentially no disadvantage in the lengths of communication paths.

Table 2 shows timings for a sequence of runs involving a 32×32 -cell grid on the one-processor subcube, a 64×32 -cell grid on the two-processor subcube, a 64×64 -cell grid on the four-processor subcube, and so forth, up to a 512×512 grid on a 512-processor cube. Since the ratio of problem size to number of processors remains constant in this sequence, an algorithm possessing ideal parallelism would require the same execution time for all runs. In practice, interprocessor communication and computational overhead disrupt this ideal relationship.

Table 2 also shows the times associated with problem setup (initialization and matrix assembly) and interprocessor communication. Each run represents 20 outer iterations of the scheme (4), each iteration of which requires five V-cycles in step (ii). In practice, the outer iterations typically converge to within machine precision tolerances in fewer than 10 iterations, so practical runtimes are smaller, and setup time has a larger effect on speedup. Still, these timings suggest that the algorithm possesses excellent parallelism in addition to its good performance in the presence of heterogeneities and fine grids.

Table 2: Runtimes (seconds) for scaled groundwater flow problems on the nCUBE 2.

Number of processors	Setup time	Communication time	Total time
1	0.178	0.205	161.516
2	0.184	1.784	159.274
4	0.201	3.690	157.260
8	0.214	4.639	158.484
16	0.251	5.775	159.967
32	0.319	5.832	160.231
64	0.530	5.908	160.620
128	0.782	5.950	160.985
256	1.396	5.979	161.673
516	2.691	6.030	163.005

5. Collocation for the Transport Equation

We turn now to Equation (2), which governs contaminant transport. Of special interest are flow regimes in which advection is dominant, in the sense that, if L is the diameter of the spatial domain, then the Peclet number $\|\mathbf{u}\|_\infty L/D_H$ is much larger than unity. For such problems, it is useful to rewrite Equation (2) in terms of the material derivative $D_t := \partial_t + \mathbf{u} \cdot \nabla$ of the fluid-solute mixture. We get

$$D_t c - \nabla \cdot (D_H \nabla c) = 0. \quad (6)$$

Consider the following initial-boundary-value problem:

$$\begin{aligned} D_t c - \nabla \cdot (D_H \nabla c) &= 0, & (\mathbf{x}, t) \in \Omega \times (0, \infty), \\ c(\mathbf{x}, 0) &= c_I(\mathbf{x}), & \mathbf{x} \in \Omega, \\ c(\mathbf{x}, t) &= 0, & (\mathbf{x}, t) \in \partial\Omega \times (0, \infty). \end{aligned}$$

This problem models the movement of an initial contaminant plume $c_I(\mathbf{x})$, so long as the plume does not approach $\partial\Omega$.

To discretize this problem in space, we use finite-element collocation on piecewise Hermite bicubics, a standard method summarized, for example, in Curran and Allen

(1990). Let Δ be a rectangular grid partitioning Ω into rectangular elements bounded by adjacent grid lines $x = x_i$ and $y = y_j$. As before, h stands for the mesh size of this grid. Denote by M the trial space of all Hermite piecewise bicubics that vanish on $\partial\Omega$. The trial function $c_h \in M$ has the form

$$c_h = \sum_{i,j} \left(c_{ij} H_{00ij} + c_{ij}^{(x)} H_{10ij} + c_{ij}^{(y)} H_{01ij} + c_{ij}^{(x,y)} H_{11ij} \right),$$

where the functions $H_{pqij}(x, y)$ form a nodal basis for M (Prenter, 1975).

To determine the nodal unknowns in this expansion, we substitute c_h into the left side of Equation (6) and force the residual to vanish at a set of collocation points $\bar{\mathbf{x}}_m$. For optimal-order accuracy, we choose these points to be the 2×2 Gauss quadrature abscissae in each element Ω_i . This procedure yields a system of ordinary differential equations in time:

$$D_t c_h(\bar{\mathbf{x}}_m, t) - \nabla \cdot [D_H \nabla c_h(\bar{\mathbf{x}}_m, t)] = 0, \quad (7)$$

These equations determine the evolution of the unknown coefficients of c_h . We project the initial function c_I onto M via interpolation to get an initial function $c_h(\bar{\mathbf{x}}_m, 0)$.

We discretize Equation (4) temporally in two steps. First, following Russell [2], we approximate $D_t c_h$ using the modified method of characteristics (MMOC). This procedure leads to a difference expression of the form

$$D_t c_h(\bar{\mathbf{x}}_m) \simeq k^{-1} \left[c_h^{n+1}(\bar{\mathbf{x}}_m) - c_h^n(\mathbf{x}_m^*) \right],$$

where $c_h^n(\mathbf{x})$ denotes an approximate value of $c_h(\mathbf{x}, nk)$ and k is the time step. The point \mathbf{x}_m^* is a *backtrack* point, which we compute according to the method of characteristics for the purely advective version of Equation (2). Theoretically, if $(\mathbf{s}(t), t)$ is a parametrization of the characteristic curve $d\mathbf{x}/dt = \mathbf{u}$ passing through $\bar{\mathbf{x}}_m$, then

$$\mathbf{x}_m^* = \bar{\mathbf{x}} + \int_{t_{n+1}}^{t_n} \mathbf{u}(\mathbf{s}(t), t) dt.$$

In practice we compute \mathbf{x}_m^* approximately by solving $d\bar{\mathbf{x}}_m^*/dt = \mathbf{u}$, subject to the “final” condition $\mathbf{x}(t_n) = \bar{\mathbf{x}}_m$, using an Euler scheme.

The second step in discretizing Equation (7) is to use alternating-direction collocation. We perturb the discrete operator equations to obtain the following factoring along the x - and y -coordinate directions:

$$(1 + k\mathcal{L}_x)(1 + k\mathcal{L}_y)c_h^{n+1}(\bar{\mathbf{x}}_m) = c_h^n(\mathbf{x}_m^*) + \mathcal{O}(k^2). \quad (9)$$

Here, $\mathcal{L}_x = -\partial_x(D_H \partial_x)$ and $\mathcal{L}_y = -\partial_y(D_H \partial_y)$. By properly numbering the collocation equations and unknowns, one can reduce the equations (9) to an algebraic system that involves highly parallel sets of matrix equations, each of which has an inexpensive, one-dimensional structure.

6. Computational Aspects of ADC

Curran and Allen (1990) discuss efficient algorithms for solving the ADC equations on parallel-architecture computers. The computational problem is “embarrassingly parallel,” in the sense that it naturally decomposes into linear systems, having one-dimensional zero structure, that one can obviously solve concurrently. Speedup curves of slope greater than 0.8 are attainable on an Alliant FX/8 eight-processor machine.

Aside from parallelism, two features of the ADC-MMOC approach make it an attractive one. First, the method inherits high-order spatial accuracy from the standard collocation approach. Percell and Wheeler (1980) show that standard collocation on piecewise Hermite cubics has $\mathcal{O}(h^4)$ spatial accuracy for elliptic spatial operators. ADC attains this accuracy with “one-dimensional” matrices having bandwidth five.

Second, the use of MMOC reduces both the temporal truncation error and the number of degrees of freedom needed to resolve sharp fronts. Russell (1980) discusses these advantages. A related observation that MMOC essentially removes the advective term from the spatial operator, leaving only the diffusive operator to be discretized via collocation. This fact is appealing on numerical grounds, since we expect collocation on Hermite cubics to yield $\mathcal{O}(h^4)$ accuracy for Equation (2) in the parabolic case, when $D_H \neq 0$, but only $\mathcal{O}(h^3)$ accuracy in the hyperbolic case when $D_H = 0$ (see Dupont, 1973). With MMOC, the collocation procedure discretizes the part $-\nabla \cdot (D\nabla)$ of the spatial operator for which it is best suited, even when the other term $\mathbf{u} \cdot \nabla$ is physically dominant.

The ADC-MMOC scheme does not strictly conserve mass in the global sense

$$E_M(t_n) := \int_{\Omega} (c_h^n - c_h^0) dv + \sum_{\nu=0}^n k \oint_{\partial\Omega} (\mathbf{u}c_h^\nu - D\nabla c_h^\nu) \cdot \mathbf{n} ds = 0.$$

This effect is common in Eulerian-Lagrangian methods (Russell, 1980; Krishnamachari et al., 1989). Numerical experiments indicate, however, that the mass balance errors are typically not excessive. In a rotating plume problem on $\Omega = (-1, 1) \times (-1, 1)$ and $T = 1$, with $h = 0.02$, the mass balance error varies with the time step k . Table 3 shows values of the *relative mass balance error*,

$$R_M := \frac{|E_M(1)|}{\int_{\Omega} c_h^0 dv},$$

for four choices of k . Since accurate backtracking is necessary to obtain reasonable mass balance, the table also shows the number N_E of Euler steps used to compute the backtrack points \mathbf{x}_m^* in each case.

Table 3: Relative mass balance errors R_M in the ADC-MMOC scheme for a rotating-plume problem on $\Omega = (-1, 1) \times (-1, 1)$, with $h = 0.02$ and $T = 1$. N_E is the number of Euler steps used in the backtracking.

Time step k	N_E	R_M
0.02	10	0.018
0.01	5	0.007
0.005	2	0.027
0.0025	2	0.015

7. Discussion

A variety of extensions are needed to make these numerical methods fully useful in modeling porous-media flows. The most obvious needs are to extend the scheme for

the flow equation to time-dependent, three-dimensional settings and to extend the ADC-MMOC scheme for the contaminant transport equation to three dimensions. These extensions involve modifications that, while conceptually straightforward, require nontrivial changes to the codes and will result in more computationally intensive algorithms. The principles that allow parallelizations should remain intact, however, so the approaches described here should be even more attractive in higher-dimensional applications.

More interesting is the need to extend the methods to problems involving tensor conductivities and tensor hydrodynamic dispersion. It is in the context of tensor conductivities that the two-level iterative scheme for the mixed-method equation has the greatest potential for practical use. Shen (1992), through delicate analysis, shows that one can lump the matrix A in the mixed-method system and preserve global accuracy in the scalar case. Thus one can eliminate the need for the outer iterations used here. However, the analysis does not appear to extend to the case when the conductivity K is a tensor. In this case, the inner-outer iteration scheme still offers reasonable prospects for effective parallelism.

Incorporating tensor hydrodynamic dispersion into the ADC-MMOC formalism most likely will require an iterative formulation, in which one lags off-diagonal entries of D_H by an iteration. The use of iterations in this setting opens the way for simultaneous iterative reduction of the truncation error introduced in the operator splitting used to effect the alternating-direction strategy. The parallelism inherent in the ADC-MMOC approach makes iterations affordable.

The overall approach of combining alternating-direction techniques with the MMOC is by no means restricted to finite-element collocation. Krishnamachari et al. (1989) discuss a related approach for a Galerkin scheme using piecewise bilinear trial functions, and one can easily imagine analogous schemes involving finite differences.

Acknowledgments

The Wyoming Water Research Center supported this work in part through a grant-in-aid. The U.S. National Science Foundation provided support through grant number EHR-910-8774. This work received support from the Applied Mathematical Sciences Program, U.S. Department of Energy Office of Energy Research. The work was performed in part at Sandia National Laboratories for the U.S. DOE under contract number DE-AC04 -76DP00789. The first author expresses gratitude to the College of Engineering and Mathematics at the University of Vermont, which provided valuable computer time and expertise during a sabbatical leave.

References

1. Allen, M.B. and Curran, M.C. (1992), "A multigrid-based solver for mixed finite-element approximations to groundwater flow," in *Computational Methods in Water Resources IX, Vol. I: Numerical Methods in Water Resources*, ed. by T.F. Russell et al., Elsevier Applied Science, London, 579-585.
2. Allen, M.B., Ewing, R.E., and Lu, P. (1992), "Well conditioned iterative schemes for mixed finite-element models of porous-media flow," *SIAM Jour. Sci. Stat. Comp.* 13:3, 794-814.

3. Celia, M.A. (1983), "Collocation on Deformed Finite Elements and Alternating Direction Collocation Methods," Ph.D. dissertation, Princeton University, Princeton, New Jersey.
4. Curran, M.C., and Allen, M.B. (1990), "Parallel computing for solute transport models via alternating-direction collocation," *Adv. Water Resour.* 13:2, 70-75.
5. Douglas, J., Ewing, R.E., and Wheeler, M.F. (1983), "The approximation of the pressure by a mixed method in the simulation of miscible displacement," *R.A.I.R.O. Analyse Numerique* 17, 17-33.
6. Dupont, T. (1973), "Galerkin methods for first-order hyperbolics: An example," *SIAM J. Numer. Anal.* 10, 890-899.
7. S.V. Krishnamachari, L.J. Hayes, and T.F. Russell (1989), "A finite element alternating-direction method combined with a modified method of characteristics for convection-diffusion problems," *SIAM J. Numer. Anal.*, 26:6, 1462-1473.
8. Percell, P., and Wheeler, M.F. (1980), "A C^1 finite element collocation method for elliptic equations," *SIAM J. Numer. Anal.* 17:5, 605-622.
9. Prenter, P.M. (1975), *Splines and Variational Methods*, Wiley, New York.
10. Raviart, P.A., and Thomas, J.M. (1977), "A mixed finite element method for second order elliptic problems," in *Mathematical Aspects of Finite Element Methods*, Lecture Notes in Mathematics vol. 606, ed. by I. Galligani and E. Magenes, pp. 292-315, Springer-Verlag, Berlin.
11. Russell, T.F. (1980), "An Incompletely Iterated Characteristic Finite Element Method for a Miscible Displacement Problem," Ph.D. dissertation, University of Chicago, Chicago, Illinois.
12. Shen, J. (1992), "Mixed Finite Element Methods: Analysis and Computational Aspects," Ph.D. dissertation, University of Wyoming, Laramie, Wyoming.
13. Tuminaro, R.S., and Womble, D.E. (to appear), "Analysis of the multigrid FMV cycle on large-scale parallel machines," *SIAM Jour. Sci. Stat. Comp.*

Myron B. Allen, Department of Mathematics, University of Wyoming, Laramie, WY 82071, USA (allen@corral.uwyo.edu).

Mark C. Curran, Applied and Numerical Mathematics Division, Sandia National Laboratories, Albuquerque, NM 87185, USA (mccurra@cs.sandia.gov).